Universal Networking Language: Advances in Theory and Applications

Research on Computing Science

Research on Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 12**, febrero, 2005. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. 04-2004-062613250000-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor Responsable: *Juan Luis Díaz de León Santiago, DISJ 690604*

***Research on Computing Science** is published by the Centre for Computing Research of IPN. **Volume 12**, February, 2005. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, February, 2005, in the IPN Graphic Workshop – Publication Office.*

Volume 12
Volumen 12

Universal Networking Language: Advances in Theory and Applications

Volume Editors:
Editores del Volumen

Jesús Cardeñosa
Alexander Gelbukh
Edmundo Tovar

Instituto Politécnico Nacional
Centro de Investigación en Computación
México 2005

ISBN: 970-36-0226-6
ISSN: 1665-9899

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.ipn.mx
http://www.cic.ipn.mx

Printing: 500
Impresiones

Printed in Mexico
Impreso en México

Preface

This volume[1] is an attempt to compile and illustrate all the open lines of research within the UNL initiative. The included papers constitute a selection of the most significant papers presented in several international conferences and workshops during the last four years that served as a meeting point for the UNL consortium. In general, papers are not restricted to UNL although they are clearly predominant; they clearly illustrate the wideness and flexibility of this UNL initiative, launched by the United Nations aiming at the elimination of linguistic barriers.

Since the starting of the UNL project in 1996, the participants in the project from initially 15 languages have made substantial progress in technical matters and the organizational aspects involved as well. This book attempts to provide a survey on the approaches and theoretical studies around UNL, since research on UNL is not only devoted to studies on interlinguas, MT or any NLP related issues, the intrinsic properties of UNL make it a firm candidate to support a wide variety of applications ranging from e-learning platforms to management of multilingual document bases. Such a variety of applications, their theoretical basis and subsequent methodological inquiries are at core of this volume.

What is UNL? Its motivation and purpose

The emerging needs and use of Internet for cultural and educational dissemination and commercial expansion of the peoples collide with linguistic diversity, which in principle diminishes the potential of Internet as a vehicle of knowledge for everybody. Aware of this problem, the Institute of Advanced Studies of the University of the United Nations University (UNU/IAS) launched the UNL project in 1996 with the initial participation of 15 languages (German, Arab, Chinese, Spanish, French, Hindi, Indonesian, English, Italian, Japanese, Latvian, Mongol, Portuguese, Russian, Thai). In short, the UNL Programme was initially conceived to support multilingual services in Internet being an alternative to classical machine translation systems.

The UNL system revolves around a unique artificial language (Universal Networking Language) that pretends to capture the meaning of written documents. This language is based on the representation of concepts and its relations. The definition of this language has been possible thanks to the collaboration of more than one hundred people, prestigious researchers, and scientists of all around the world, that worked during the first three years of the project to produce a final version of the UNL specifications[2].

[1] Earlier versions of the papers at pages 10, 109, 117, 125, 145, 215, 230, 254, 268, 276, 309, 347, 359, 370, 380 have been published in the Proceedings of Convergences'03, Alexandria, Egypt. Earlier versions of the papers at pages 3, 10, 27, 38, 101, 261, 326 have been published in the Proceedings of LREC-2002.

[2] UNL Specifications, v.3.1 available at
http://www.undl.org/unlsys/unl/UNL%20Specifications.htm

The UNL organization

The UNL initiative has often been regarded as “hidden organization”. The first years of the project (1996-2000) were devoted to the definition of the interlingua and to the development of the essential components required to undertake the basic process in UNL (mainly dictionaries and language generators). During this period, the organization was closed and limited to a number of participants, because of the need to define the specifications of the language.

By the end of this period, the UNL project reached a significant degree of quality in the development of components, linguistic resources and technical specifications; and the specifications were finally produced. Once the specifications were finished, they were made public and accessible to all the international community, so that collaboration and *participation in this initiative is completely open.*

As a consequence of this degree of development, the Board of the United Nations University, in its fifth meeting in 2000, agreed on the creation of a new institution responsible for the organization and promotion of the UNL in the future under the umbrellas of the United Nations. This new entity was the UNDL Foundation, with headquarters in Geneva[3]. The development of the components of different languages was assigned to the so-called Language Centres, constituted by the initial teams in each country in charge of the development of the essential components of UNL.

The year 2004 represents a turning point in the evolution of UNL for two main reasons. First, it is the year where a new period coordinated and fostered by the Language Centres starts for the debugging, updating and expansion of linguistic resources and developed components of their representative languages, in order to respond to the institutional and marketable challenges at a pre-competitive level in the support of multilingual services. Second, it is the year where the UNL patent has been approved in USA for the UN (US Patent No. 6704700 B1, March 2004). It has been the first software patent of the United Nations.

Open nature and scientific dissemination of UNL

Since 2002, an open annual conference around the convergence of language, culture and knowledge is being held as a meeting point for researchers, politicians, linguists and engineers. The most recent edition of this open conference was Convergences’03, held in Alexandria, Egypt. The most significant papers from this conference have selected and included in this volume. Additionally, an international workshop on UNL and Interlinguas was organized in 2002 (International Workshop on UNL, other Interlinguas and their Applications, held at Las Palmas de Gran Canaria, May, 2002), papers from this workshop are also compiled in this volume. Finally, we include the papers of the current edition of the UNL Workshop, held in Mexico D.F, February, 2005.

These conferences and workshops try to be a forum where all the interested people in this initiative find a vehicle for communication and exchange of knowledge. The

[3] www.undl.org

UNL is a great initiative that could never succeed and advance if the number of participants is limited to the initial ones. The heterogeneity of the authors and languages involved in this selection of papers shows the open nature of UNL.

Research on UNL: Current Trends

Apart from the mere applied studies of UNL, there is a current important trend on theoretical studies of UNL, even though there is a final version of the specifications of the language, dating to July 2003.

The rationale for such theoretical research is the need for standardization and homogenization on the use of the Interlingua both at the applied level and at the theoretical level. The UNL Specifications turned out to be subject to different personal interpretations, thus *creating own UNL dialects.* This is not desirable for an interlingua, that claims to be language independent and that, in fact, turned out to be "person-dependent". For this reason, it is important and desirable to foment theoretical studies on UNL, both from the linguistic point of view and the knowledge point of view.

From a scientific point of view, UNL follows the approach of the concept of Interlingua, as an "artificial" language aiming at the neutral representation of linguistic meaning. In this sense its roots can be sought in the tradition of MT interlinguas and in the tradition of Knowledge Representation formalisms.

When viewed as an interlingua, UNL differs from some of its predecessors and current Interlinguas in the generality of appliance, that is, UNL is not restricted to a number of languages or to a given domain. Thus, its design pretended to show the highest degree of language independence while retaining natural language expressiveness in order to support multilingual generation tasks.

Of course, the staging of UNL is such a general enterprise that requires research and efforts. This process can be divided into several periods:

- Creation of deconversion and enconversion modules, (see Part 3) that is, development of the basic tools to undertake the basic architecture of the UNL system (enconversion and generation), along with dictionaries. Although basic, it is *conditio sine qua non* to have powerful generation systems. This a fruitful trend in the UNL consortium, with three different approaches:

  1. The official one: those using a common engine provided by the UNL Center.
  2. The integrative ones: those that have integrated UNL into pre-existing MT systems, following the transfer-based architecture, showing the flexilibility of UNL with good results.
  3. The new ones: those that have noticed the drawbacks of the *official* components, and have decided to create new architectures for generation

  It should be noticed that emphasis is put on the deconversion process, quantitatively proven by the number of papers devoted to generation. Teams usually develop generation systems, not so much enconversion systems, although the integrative usually includes both processes in UNL.

- Application of UNL in other contexts (see Part 3). Should UNL be considered as an interlingua, it can be applied in fields and tasks other than multilingual generation, being the main one Knowledge representation and Knowledge Management.
- Use of external lexical and ontological resources. It is important as well, and following the spirit of the integrative approaches, the use of external lexical resources such as Wordnet to enhance some of the processes of UNL, especially in the lexicographic part (see Part 3, also). This is also a trend and the philosophy of UNL: integration and complementation of resources is encouraged, rather than confrontation. And this is the spirit of the consortium and of every work in UNL.

From an engineering point of view, research is taken on:

- Creation of methodologies in the workflow.
- Standardization of UNL, integration of UNL into current standards.

Why such studies methodologies and standards? Because of the heterogeneity and diversity of the current consortium, it is needed such a process of standardization and methodologies, since the short and medium term objective of UNL is its staging in the market, where standards and methodologies are required in order to pursue higher productivity and quality. The areas of linguistic engineering together with knowledge engineering are claiming for such methodologies and processes of standardization.

The Future

After some time developing components and systems to support the multilingual services, UNL researchers and new teams have discovered that the UNL could be support of other applications as crosslingual information retrieval, knowledge repositories, automatic building of ontologies from texts once repressented in UNL and much more. UNL could be useful in new possible applications in areas where a common conceptual representation is needed, independent of any particular language. For doing it, new necessities emerge; particularly when putting together semantics and multilingualism. More theoretical studies are needed, along with the tuning up of resources and tools, the proper standardization of the interlingua and processes for enconverting and deconverting, and of course the integration and definition of the lexical component of UNL.

The Structure of the Book

The volume is divided into four parts.

Part 1. Introduction

This fist part is an introduction to the language itself, and its purpose is to set up the reader in the UNL context. These introductory papers posit the general philosophy of the language (paper at page 3) and provide a general introduction to the language itself and to the context of multilingual generation, one of the main and most basic “applications” supported by UNL (paper at page 10).

Part 2. Fundamentals

This part is dedicated to theoretical studies on UNL. As already said, UNL is mainly an interlingua. There are many aspects that have to be taken into account when designing an interlingua, such as its expressiveness, degree of language-independency, accuracy and formality of the language, etc. Most of these issues are covered in this part. Thus, the part opens up with an experiment on the common understandability of UNL by different humans and the admissible degree of indeterminacy and ambiguity in an Interlingua (paper at page 27). Pure theoretical studies on the universality of UNL and its adequacy from a representational and linguistic point of view follow (papers at pages 51 to 101). It has to be pointed out that this part is not exclusively devoted to UNL, but to the field of interlinguas in general (paper at page 38; paper at page 109).

All these papers point at the proper designs of the Interlingua. However, there is another important aspect worth of consideration in any artificial language, namely, the syntactic formalism of the formal language and its adequacy to the declared purpose. These topics are addressed in papers at page 117 and at page 125, where the emphasis is put on the syntactic properties of UNL expressions and its consequences to other issues such as analysis or proper deconversion. Finally, there is a (recurrent) thematic shift; UNL is not viewed as an interlingua to support linguistic tasks, but as a language for knowledge representation (papers at page 138 and at page 145).

These two sides of UNL (an interlingua to support linguistic tasks and a as knowledge representation language) determine the nature of the applications dealt with in Part 3.

Part 3. Applications

The core applications of UNL are those that support the tasks of NL analysis and generation (*enconversion* and *deconversion* in the UNL jargon). When dealing with NLP tasks, the scene is quite heterogeneous: from the use of common generation tools provided by the UNL Center (as shown in papers at pages 215 and 241), to the integration of existing MT translation systems based on the transfer architecture to support an Interlingua architecture (papers at pages 157 and 230). Other languages are supported with new tools, but differs in their configuration and architecture (maybe reflecting language variety, maybe reflecting different ways to support generation and of course, as an advanced over common tools, like Deco). Chinese, Brazilian Portuguese, Arabic or Armenia are example of this, where very different paradigms are illustrated in order to undertake the generation task (papers at pages 167, 175, 195, and 210, respectively).

Papers at pages 254 to 276 illustrate the development of workbenches to support the processes of edition, generation and training and with the creation of multilingual platforms within the UNL framework.

In parallel with the theoretical studies of Part 2, UNL also presents and applied dimension when conceived as a language for knowledge representation (papers at pages 337 and 359). These papers present the use of UNL as an extension (or complementation) to the expressiveness of standard languages such as XML (illustrated in papers at pages 300 and 309), as the communication language among agents, developed in paper at page 326, or as the support of case-based reasoning systems (paper at page 347). It is also remarkable the possibility of complementation and integration with other lexical and ontological resources such as WordNet (papers at pages 370 and 380) to the enhancement of the processes of knowledge acquisition and representation within the UNL context. Finally, paper at page 286 shows how to extend the expressivity of UNL in order to represent and formalize meaning coming for oral sources.

Part 4. Methodologies

Finally, the volume ends up with the methodological work. Methodologies target at the creation of methodologies to support multilingual services (papers at pages 395 and 413) and for the optimization of knowledge intensive tasks (paper at page 430). Needless to say, methodologies conforms an integral part of the UNL R+D activities, as long as productivity, quality and a real consolidation of UNL are pursued both at the scientific and commercial levels.

Mexico D.F, 16th February 2005

Jesús Cardeñosa
Alexander Gelbukh
Edmundo Tovar
Editors

Table of Contents

Índice

INTRODUCTION: Setting up UNL

FOUNDATIONS

APPLICATIONS

METHODOLOGIES

Prologue

UNL is an ongoing worldwide initiative starting in 1996. Almost 10 years have passed a big span of time for a project. We could say that UNL didn't meet its expectations. But let's have a closer look to UNL, the project, its basics and objectives. A closer look at its objective will reveal that this affirmation is gratuitous and unmotivated.

The Problem: Linguistic Diversity

UNL was launched by IAS/UNU to erase linguistic barriers. Linguistic barriers collide with the enhancement of linguistic diversity and the value that native languages as one of the main vehicles to express one's cultural identity. Apart from sociocultural issues, linguistic diversity also knows an economic and political dimension. Institutions like the United Nations or the European Union have to face everyday with the barriers that linguistic diversity imposes. It is well known the enormous amount of documentation that these institutions produce everyday, which have to be produced in all their official languages: 6 for the UN, 25 for the European Union. It is simply unfeasible to rely on human translators for the production of all these amount of documentation.

Aware of this, the IAS/UNU launched the UNL project, aiming at the real access of information in the own native language and not recurrent to dominant languages. UNL is basically an artificial language where contents expressed in natural languages can be converted to and subsequently, contents written in UNL can be generated into any natural language, provided that the adequate tools are built.

MT and Multilinguality

From the technological point of view, multilinguality has been tackled by Machine Translation. In the evolution of the area of MT, there is variety of architectures to undertake the task of translating the *contents* of one text written in a given language into another language. Transfer-based systems could be regarded as the most productive and of better quality. But they are hindered by the exponential growth in the modules to be developed when the number of involved languages increases. A transfer-based system involving N languages need to develop N*(N-1) modules. An astronomic number to create real multilingual platforms.

Further, although there are some very good systems, the quality of these systems seem to be limited, since after years of refinement, the MT system does not surpass a given degree of quality. Besides, the development of transfer based MT systems is usually reduced to the so-called majority languages (English, French, German and even Spanish or Italian), but it is fairly rare to find a good quality and wide coverage MT system covering English and Polish, let's say.

Transfer based MT is not the only option, Interlingua-based systems represents an alternative to transfer systems. Interlingua-based MT does not work on pair of languages, but translation is carries out to and from an artificial language that serves as a pivot for all the natural languages involved in the system. This architecture tries to overcome the exponential growth of transfer-based systems, since the number of modules to develop for N languages is 2*N and the inclusion of new languages into the system does not affect the other language modules. In this way, UNL follows the architecture of Interlingua-based MT systems.

Usually, Interlinguas are abstract formal (or semi formal) languages that captures the meaning of texts in a language independent way. Ideally, the Interlingua should not be close to a given particular language and should not include linguistic devices proper of natural languages. In this way, Interlingua-based systems seem the most plausible (and even the unique) option to tackle massive multilinguality.

But Interlinguas has been often rejected within the scientific community and since their boom in the 80ies, there have no commercial application of Interlinguas and the systems developed under this trend were laboratory products. Why is this so? Let's have a look at the properties of interlinguas.

Problems with Interlinguas

Interlinguas are *semantic languages* designed to represent the meaning of any given text, ideally satisfying the following conditions:

(a) They are language neutral.
(b) They are precise, unambiguous, formal languages

Being so, they usually show the following characteristics:

– Interlinguas are intimately tied up with ideas about the representation of meaning, being meaning the most abstract and deepest level of linguistic analysis (that should be common to all languages, far enough from surface representation of languages).
– An Interlingua is "another language" in the sense that it has autonomy and thus its components need to be defined: vocabulary and "relations" mainly. Besides, and Interlingua is an artificial language that should be as expressive as natural languages.

Here we find the main bottleneck of interlinguas: its proper design and definition. Defining an Interlingua involves the following parameters:

(a) A language whose "atoms" are not dependent on any given natural language so that the ambiguity of natural languages is eliminated.
(b) A language whose "atoms" are not dependent on a given natural language so that the concepts and ideas expressed in different natural languages can be easily and naturally expressed in the Interlingua.
(c) A language that is as expressive as a natural language so that what can be expressed in natural languages can be transposed to the Interlingua, and from the interlingua to other natural languages.

These three conditions make interlinguas hard to design. It is quite difficult to find the equilibrium between language independency, degree of abstraction and expressiveness in a formal device such an Interlingua. Maybe this difficulty in the design of interlinguas is the reason why they have not been successful at least in open domains within massively multilingual environments. The examples of interlingua-based systems are domain dependent and quite limited in the number of languages.

Is UNL a Viable Solution?

The panorama appears quite despairing. While Interlinguas are theoretically biased and difficult to put into practice, transfer based systems have proved to be unattainable when dealing with massive multilinguality. Maybe the concept of Interlingua should be revisited, and re-adapted to real necessities and to real scenarios. This is the spirit of UNL. UNL, by its definition and by its most basic architecture is definitely an Interlingua-based system. Its targets are the support of multilinguality, not restricted to a given domain or to a given family of languages. Thus, the design of a interlingua like UNL encounters all the possible barriers that an Interlingua may encounter (especially to find a real language independent representation).

So why we could considered UNL as different, as a new viable technology if interlinguas were rejected a long time ago? First, let's remember the main objective of UNL:

- to generate and produce contents in any natural language in any domain.
- to support multilingual services.

That is, there is a primacy of generation and coverage of languages and domains, which means that a **very expressive formalism** has to be designed in order to represent such a variety of contents coming from any natural language.

Let's illustrate this fact by have a closer look at the vocabulary of the Interlingua, one of the most difficult and polemic issues of UNL and of any Interlingua. UNL utilizes the so-called Universal Words as the semantic atoms of the Interlingua (no decomposable). They exhibit the following main characteristic:

They are based on English headwords.

From this very simple definition, we can conclude that UNL is language biased (English) and thus:

1. UNL is based on a natural language:
2. It hinders logical relations and inferences (facilitated by primitive based solutions)
3. Its vocabulary is a potential source of ambiguity
4. Its vocabulary fosters lexical and conceptual mismatches among languages.

So is there any advantage in the UW system and in the overall essence of UNL? Well, if theoretical reasons do not support the design of open-domain interlinguas, let's look at the practical or pragmatic ones.

(a) UNL is based on a natural language. At first sight could be a drawback, however, the expressiveness of a natural language is inherited by the Interlingua, thus allowing for the representation of a variety of domains and concepts.

(b) UNL shows an English oriented vocabulary. At this moment, English is the lingua franca, the most accessible to work with for Indo-Europeans, Semitic, Japanese, Chinese, etc. Bilingual dictionaries usually have English as one of their target/source languages, thus the development of lexicographic resources is facilitated by choosing English as the most basic atoms of the language.

Of course, this approach (although supported by pragmatism) is far from perfect. Even at first sight, it can be considered as naïve, since it merely "suggest" well known problems in lexical semantics (like support verbs, compounds expressions, connotational meaning, etc). For this reason, theoretical research on the UNL as a language itself should be fostered within the Consortium, while respecting the basic nature of the language.

That is, UNL should be viewed rather than a perfect Interlingua as **the pillars to support multilingual services**. Its natural language orientation (apparently, its weakest points as an Interlingua) turns the language as a candidate to the support of multilinguality and facilitates converting contents to and from UNL. There are several aspects that support it. First, the creation of generators of medium quality (where post-edition is possible) is rather straightforward. Second, its flexibility and language orientation makes it possible to integrate UNL into other pre-existent MT systems (be it transfer-based be it another architecture) which extends the range of application of UNL and makes possible to alleviate the problem of exponential growth in transfer-based systems. And last, but not least, the processes of enconverting and deconverting are independent so that if generation is taken as a priority, generators are constructed first; the process of enconversion can be done manually, due to the human readability of the language.

At this point in the evolution of UNL, there appears a contradiction, UNL is still not theoretically mature, but from an applied perspective, it is. In the short term there is priority for the UNL Consortium to get feedback from previous experiences in Interlinguas, from Linguistic Theory (semantics, logic, and lexical semantics) in order for UNL to grow and find a place in the scientific community and, why not, in the market as a real approach to support multilinguality, once the applications and utilities are clear and defined within the UNL Programme.

Prospective

So is it worth another attempt? Definitely yes, the real need to overcome linguistic barriers (be it at the institutional level, be it at the social level) claims for a solution to the problem of multilinguality. Transfer based systems simply are out of question *if isolated*. This doesn't mean that they are useless: they are not. An interlingua like UNL is conceived as another autonomous languages, close enough to the superficial form of natural languages, thus integration of the Interlingua into the transfer system is possible and not a *contradiction in terminis*.

After several years of experience, we know that knowledge and language generation do not go *on a par*. Thus the final design have to be done bearing the ultimate purpose of the interlingua (the closer to language semantics is, the better to generate languages) and probably will lead to the success of the interlingua.

A Final Word

I would like to thank the editors of this book for their invitation to write a prologue to this work and to collaborate with them in the selection and revision of the selected papers presented in this volume. Hopefully it will provide a thorough understanding of the UNL Programme, its meaning, its evolution, its shortages and its strengths.

Carolina Gallardo

INTRODUCTION: SETTING UP UNL

A Rationale for Using UNL as an Interlingua and More in Various Domains

Christian Boitet

GETA, CLIPS, IMAG
385, Av. de la Bibliothèque, BP 53
F-38041 Grenoble cedex 9, France
Christian.Boitet@imag.fr

Abstract. The UNL *language* of semantic graphs may be called as a "semantico-linguistic" interlingua. As a successor of the technically and commercially successful ATLAS-II and PIVOT interlinguas, its potential to support various kinds of text MT is certain, even if some improvements would be welcome, as always. It is also a strong candidate to be used in spoken dialogue translation systems when the utterances to be handled are not only task-oriented and of limited variety, but become more free and truly spontaneous. Finally, although it is not a true representation language such as KRL and its frame-based and logic-based successors, and although its associated "knowledge base" is not a true ontology, but rather a kind of immense thesaurus of (interlingual) sets of word senses, it seems particularly well suited to the processing of multilingual information in natural language (information retrieval, abstracting, gisting, etc.).The UNL *format* of multilingual documents aligned at the level of utterances is currenly embedded in html (call it UNL-html), and used by various tools such as the UNL viewer. By using a simple transformation, one obtains the UNL-xml format, and profit from all tools currently developed around XML. In this context, UNL may find another application in the localization of multilingual textual resources of software packages (messages, menu items, help files, and examples of use in multilingual dictionaries.)

1 Introduction

UNL is the name of a project, of a meaning representation language, and of a format for "perfectly aligned" multilingual documents. There is some hefty controversy about the use of the UNL language as an "interlingua", be it for translation or for other applications such as cross-lingual information retrieval. On the other hand, there is almost no discussion on the UNL format, in its current form, embedded in HTML, or some directly derivable form, embedded in XML.

We argue that the UNL language is indeed a good interlingua for automated translation, ranging from fully automatic MT to interactive MT of several kinds through, we believe, spoken translation of non task-oriented dialogues. It is also more than that, due to the associated "knowledge base", and has a great potential in textual information processing applications.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 3–9.

We will first give our view of what the UNL language is, and then develop a "rationale" for using the UNL language UNL along the previous lines. We will then describe some interesting potential uses of the UNL format in an "XML-ized" form.

2 The UNL language

The UNL representation is made of "semantic graphs" where a graph expresses the meaning of some natural language utterance. Nodes contain lexical units and attributes, arcs bear semantic relations. Connex subgraphs may be defined as "scopes", so that a UNL graph may be a hypergraph. Figure 1 illustrates a UNL graph.

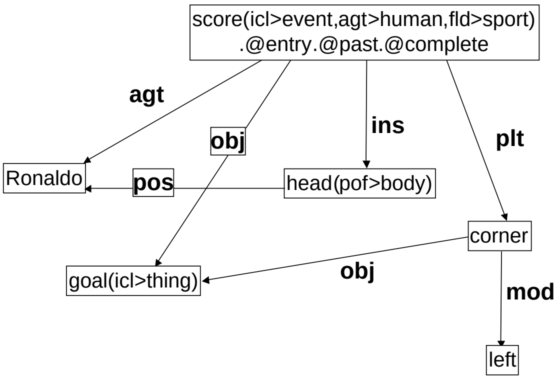


Fig. 1. A possible UNL graph for “Ronaldo has headed the ball into the left corner of the goal”

The lexical units, called Universal Words (in French, not "mot universel" but better "Unité de Vocabulaire Virtuel" or UVV or UW), represent word meanings, something less ambitious than concepts. Their denotations are built to be intuitively understood by developers knowing English, that is, by all developers in NLP. A UW is an English term or pseudo-term possibly completed by semantic restrictions.

A UW such as "process" represents all word meanings of that lemma, seen as citation form (verb or noun here). The UW "process(icl>do, agt>person)" covers the verbal meanings of processing, working on, etc.

The attributes are the (semantic) number, genre, time, aspect, modality, etc.

The 40 or so semantic relations are traditional "deep cases" such as agent, (deep) object, location, goal, time, etc.

One way of looking at a UNL graph corresponding to an utterance U-L in language L is to say that it represents the abstract structure of an equivalent English utterance U-E as "seen from L", meaning that semantic attributes not necessarily expressed in L may be absent (e.g., aspect coming from French, determination or number coming from Japanese, etc.).

3 Some arguments for using the UNL language in various contexts

To show that using UNL is not only a workable but a good or perhaps the best idea at the moment, we can say that

- the "pivot" technique HAS BEEN not only experimented but deployed successfully (ATLAS, PIVOT, ULTRA, KANT).
- in particular, ATLAS-II (Fujitsu) is built on the basis of a pivot from which the UNL representation has evolved. The main designer of UNL, H. Uchida, was also the main designer of ATLAS-II.
- ATLAS-II has been recognized as the best EJ/JE MT system in Japan for over 10 years and has a very large coverage (586,000 words in English and Japanese).
- interlingual representations can not in principle be used (alone) to achieve the highest quality achievable by transfer systems, BUT they can give quite high quality as demonstrated by ATLAS-II.
- due to the precise nature of UNL, it is possible for human non-specialists to improve a UNL representation interactively, a posteriori, from any UNL-related language, and on demand (meaning partially — think of "lazy improvement").
- in many contexts other than translation, an interlingual, semantic-oriented representation like UNL is actually the best solution. For example, all applications related to information processing in multilingual contexts don't need a very precise representation of the FORM of the information, they need a precise ENOUGH representation of the INFORMATION CONTENT of the information.
- applications such as information retrieval and abstracting have already been prototyped successfully with UNL. It is far easier to generate SQL or SQL-like queries and answers from a UNL form than from text in many languages.

4 Applications of the UNL format

The UNL *format* of multilingual documents aligned at the level of utterances is currenly embedded in html (call it UNL-html). A sentence is represented between the [S] and [/S] tags. Its original text is contained between {org:el} (English, here) and {/org}, its UNL graph between {unl} and {/unl}, each French version between {fr} and {/fr}, and analogously for other languages. Atrtibutes such as version, date, location, author, etc. may appear in the tags. Here is a slightly simplified example of a file in UNL-html format.

```
<HTML><HEAD><TITLE>
Example 1  El/UNL
</TITLE></HEAD><BODY>
[D:dn=Mar Example 1, on= UNL French,
mid=First.Author@here.com]
```

```
[P]
[S:1]
{org:el}I ran in the park yesterday.{/org}
{unl}
agt(run(icl>do).@entry.@past,i(icl>person))
plc(run(icl>do).@entry.@past,park(icl>place).@def)
tim(run(icl>do).@entry.@past,yesterday)
{/unl}
{cn dtime=20020130-2030, deco=man}
我昨天在公園裡跑步{/cn}
{de dtime=20020130-2035, deco=man}
Ich lief gestern im Park. {/de}
{es dtime=20020130-2031, deco=UNL-SP}
Yo corri ayer en el parque.{/es}
{fr dtime=20020131-0805, deco=UNL-FR}
J’ai couru dans le parc hier. {/fr}
[/S]
[S:2]
{org:el}My dog barked at me.{/org}
{unl}
agt(bark(icl>do).@entry.@past,dog(icl>animal))
gol(bark(icl>do).@entry.@past,i(icl>person))
pos(dog(icl>animal),i(icl>person))
{/unl}{de dtime=20020130-2036, deco=man}
Mein Hund bellte zu mir.{/de}
{fr dtime=20020131-0806, deco=UNL-FR}
Mon chien aboya pour moi.
[/S] [/P][/D]
</BODY></HTML>
```

The French versions have been produced automatically while the German and Chinese versions have been translated manually.

The output of the UNL viewer for French is:

```
<HTML><HEAD><TITLE>
Example 1  El/UNL
</TITLE></HEAD><BODY>
J’ai couru dans le parc hier.
Mon chien aboya pour moi.
</BODY></HTML>
```

and will probably be displayed by a browser as:

Example 1 El/UNL

J’ai couru dans le parc hier. Mon chien aboya pour moi.

and similarly for all other languages.

The UNL viewer produces on demand as many html files as languages selected and sends them to any available browser.

The UNL-html format predates XML, hence the special tags like [S] and {unl}, but it is easy to derive from it an XML format and to transform the documents into an equivalent "UNL-xml" format. Then, using DOM and javaScript, it is possible to produce various views, including that of a classical viewer, a bilingual or multilingual editable presentation, and a revision interface where not only the text but the UNL graph and possibly other structures may be directly manipulated.

Let us take an example from an experiment performed for the "Forum Barcelona 2004" on documents in Spanish, Italian, Russian, French and Hindi. Hindi and Russian are not shown. The XML form is simplified (see figure 2).

```
<unl:S num: “1”>
<unl:org lg: “en”>
<unl:unl>
<unl:arc>agt(retrieve.@entry.@future, city) </unl:arc>
<unl:arc>tim(retrieve.@entry.@future, after) </unl:arc>
<unl:arc>obj (after. Forum) </unl:arc>
<unl:arc>obj(retrieve.@entry.@future, zone.@indef) </unl:arc>
<unl:arc>mod(zone.@indef, coastal) </unl:arc>
<unl:el> After a Forum, a city will retrieve a coastal zone </unl:el>
<unl:es> Una ciudad recuperará una zona de costa después de Forum
</unl:es>
<unl:fr> Una cité retrouvera une zone côtière après un forum </unl:fr>
<unl:it> Città ricuperarà une zone costiera dopo Forum </unl:it>
```

Fig. 2. Simplified XML form. Correct sentences are produced by the deconverters from correct and complete UNL graphs. Suppose for the sake of illustration that some UNL graph has been produced from a Chinese version, and does not contain definiteness and aspectual information. All results may be wrong wrt articles, and some wrt aspect.

The idea of "coedition" is applicable if there is a UNL graph associated with a segment one wants to modify. The goal is to share the revisions across languages, by reflecting them on the UNL graph, e.g.

- add ".@def" on the nodes containing "city", "Forum".
- replace "retrieve" by "recover" and add ".@complete" on the node containing it.

It is not possible in principle to deduce the modification on the graph from a modification on the text. For example, replacing "un" ("a") by "le" ("the") does not entail that the following noun is determined (.@def), because it can also be generic ("il aime la montagne" = "he likes mountains"). Hence, the technique envisaged is that:

- revision is not done by modifying directly the text, but by using a menu system,
- the menu items have a "language side" and a hidden "UNL side",
- when a menu item is chosen, only the graph is transformed, and the action to be done on the text is stored and shown next to its focus in the "To Do" zone,
- at any time, the new graph may be sent to the L0 deconverter and the result shown. If is is satisfactory, that shows that errors were due to the graph and not to the deconverter, and the graph may be sent to deconverters in other languages. Versions

in some other languages known by the user may be displayed, so that improvement sharing is visible and encouraging.

New versions will be added with appropriate tags and attributes in the original multilingual document in UNL-xml format, or in a DBMS, so that nothing is ever lost, and cooperative working on a document is feasible. UNL may find another application in the localization of multilingual textual resources of software packages (messages, menu items, help files, and examples of use in multilingual dictionaries.)

Apart of the "coedition", there are many other portential applications of UNL, such as:

- crosslingual information retrieval, on which we are currently working,
- abstracting & gisting, which has been prototyped at NecTec and in India,
- localization of software packages: messages in multiple languages could be created from UNL graphs produced from a graphical interface or by enconversion, and then sent to appropriate deconverters.

For this last point, we have found how to represent messages including variables (such as integers, file names etc.), but not yet how to handle messages including morphological or even lexical variants (as "4 goda / 5 let" for "4 years / 5 years" in Russian).

5 Conclusion

The UNL language is an artificial interlingua, embeddable in html or xml formats for multilingual document representation and processing. Because of its both abstract and linguistic nature, the UNL language offers many more interesting potential applications than other types of interlingua such as task and/or domain specific interlingua.

The history of MT shows that UNL will also be usable in the context of high-quality MT, quality being obtained through typology specialization and/or interactive improvement, a priori (interactive disambiguation after all-path robust analysis) and/or a posteriori by coedition of the text in any language and the corresponding UNL graph.

References

Blanc É. & Guillaume P. (1997) *Developing MT lingware through Internet : ARIANE and the CASH interface*. Proc. of Pacific Association for Computational Linguistics 1997 Conference (PACLING'97), Ohme, Japon, 2-5 September 1997, 1/1, pp. 15-22.

Blanchon H. (1994) *Perspectives of DBMT for monolingual authors on the basis of LIDIA-1, an implemented mockup*. Proc. of 15th International Conference on Computational Linguistics, COLING-94, 5-9 Aug. 1994, 1/2, pp. 115—119.

Boitet C., Guillaume P. & Quézel-Ambrunaz M. (1982) *ARIANE-78, an integrated environment for automated translation and human revision*. Proc. of COLING-82, Prague, July 1982, North-Holland, Ling. series 47, pp. 19—27.

Boitet C. (1994) *Dialogue-Based MT and self-explaining documents as an alternative to MAHT and MT of controlled languages*. Proc. of Machine Translation 10 Years On, 11-14 Nov. 1994, Cranfield University Press, pp. 22.21—29.

Boitet C. & Blanchon H. (1994) *Multilingual Dialogue-Based MT for Monolingual Authors: the LIDIA Project and a First Mockup*. Machine Translation, Vol. 9, N° 2, pp. 99—132.

Boitet C. (1997) *GETA's MT methodology and its current development towards personal networking communication and speech translation in the context of the UNL and C-STAR projects.* Proc. of PACLING-97, Ohme, 2-5 September 1997, Meisei University, pp. 23-57.

Boitet C., Réd. (1982) *"DSE-1"— Le point sur ARIANE-78 début 1982*. Contrat ADI/CAP-Sogeti/Champollion (3 vol.), GETA, Grenoble, janvier 1982, 616 p.

Brown R. D. (1989) *Augmentation*. (Machine Translation), Vol., N° 4, pp. 1299-1347.

Ducrot J.-M. (1982) *TITUS IV*. In *Information research in Europe. Proc. of the EURIM 5 conf. (Versailles)*, edited by Taylor P. J., London, ASLIB.

Kay M. (1973) *The MIND system*. In *Courant Computer Science Symposium 8: Natural Language Processing*, edited by Rustin R., New York, Algorithmics Press, Inc., pp. 155-188.

Lafourcade M. (2001) *Lexical sorting and lexical transfer by conceptual vectors*. Proc. of MMA'01, 29-31/1/01, SigMatics & NII, Tokyo, 10 p.

Lafourcade M. & Prince V. (2001) *Synonymies et vecteurs conceptuels*. Proc., 29-31/1/01, SigMatics & NII, Tokyo, 10 p.

Maruyama H., Watanabe H. & Ogino S. (1990) *An Interactive Japanese Parser for Machine Translation*. Proc. of COLING-90, 20-25 août 1990, ACL, 2/3, pp. 257-262.

Melby A. K., Smith M. R. & Peterson J. (1980) *ITS : An Interactive Translation System*. Proc. of COLING-80, Tokyo, 30/9-4/10/80, pp. 424—429.

Moneimne W. (1989) *TAO vers l'arabe. Spécification d'une génération standard de l'arabe. Réalisation d'un prototype anglais-arabe à partir d'un analyseur existant*. Nouvelle thèse, UJF.

Nirenburg S. & al. (1989) *KBMT-89 Project Report.*, Center for Machine Translation, Carnegie Mellon University, Pittsburg, April 1989.

Nyberg E. H. & Mitamura T. (1992) *The KANT system: Fast, Accurate, High-Quality Translation in Practical Domains*. Proc. of COLING-92, 23-28 July 92, ACL, 3/4, pp. 1069—1073.

Sérasset G. & Boitet C. (2000) *On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter*. Proc. of COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL, 7 p.

Slocum J. (1984) *METAL: the LRC Machine Translation system*. In *Machine Translation today: the state of the art (Proc. third Lugano Tutorial, 2–7 April 1984)*, edited by King M., Edinburgh University Press (1987).

Wehrli E. (1992) *The IPS System*. Proc. of COLING-92, 23-28 July 1992, 3/4, pp. 870-874.

Standardization of the Generation Process in a Multilingual Environment

Jesús Cardeñosa, Carolina Gallardo and Edmundo Tovar

Universidad Politécnica de Madrid, 28660 Madrid.
{carde,carolina,edmundo}@opera.dia.fi.upm.es

Abstract. Natural language generation has received less attention within the field of Natural language processing than natural language understanding. One possible reason for this could be the lack of standardization of the inputs to generation systems. This fact makes the systematic planning of the process of developing generation systems to become difficult. The authors propose the use of the UNL (Universal Networking Language) as a possible standard for the normalization of inputs to generation processes.

1 Introduction

In natural language processing (from now on NLP) two areas can be differentiated: analysis and generation. However, one has not received the same attention as the other from the scientific community, that is why generation can be considered as the "poor brother" of the NLP. The reason for this minor development is the different nature of the input to the analysis and generation systems. The input to the analysis systems is always natural language, whose casuistic and phenomenology are known; while in a generation system, the output is always known, but not what it is going to generate from [1].

The input to a generation system varies depending on whether it is monolingual generation (dialogue systems) or a multilingual system (mainly machine translation systems). In dialogue systems it is difficult to establish appropriate characteristics common to all inputs, because "the problem" of generation is usually solved with solutions *ad hoc*, depending on the application and the system language. In machine translation systems, there are also many differences in the inputs to the generation subcomponents, conditioned by the nature of system architecture (transfer, interlingua, etc.), the kind of grammars being used (declaratives vs. procedural) [2], or the number of languages in the system.

This difference in the input to the generators makes a systematic planning of their development process impossible (main cause of the minor development of generation compared to analysis). It is necessary then, that the input to the "generator" can be supported with an appropriate model of contents representation, separated from the format or language that ensures a standard process for the development of generation systems.

In this article we propose the UNL as a possible standard for the generation inputs. To achieve this, in section 2 we will introduce the main generation architectures. Sec-

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 10–24.

tion 3 will describe in detail the UNL system, its qualities and basic architecture for generation. Section 4 will establish the conditions required by any technology in order to be considered a standard and which ones are fulfilled by the UNL. The article will end with the description of a real massively multilingual system (HEREIN) where UNL has been formally studied and proposed as a de-facto standard for generation of contents in natural languages.

2 Generation Architectures

2.1 Dialogue Systems

Dialogue systems represent one of the main applications of natural language generation. This kind of systems have as their most important target "to present information to the users in an easy to understand format" [3] in very specific fields where the user generally interacts with the system in the same language. The user asks the system specific information; once obtained, the system can show it through an answer in natural language. This answer is very frequently obtained (with certain success) through the generation of a "built" language from a series of templates that keep a predefined relationship with the templates that support the questions [4]; this means the generation process takes as input a representation that depends on the way the user makes the question. It could be said that there is not a thorough analysis of the text, nor an abstract representation of the information that should be given to the user. The great dependency of the source language and the domain restrain the construction of multilingual dialogue systems and the reuse of these systems in other domains.

2.2 Machine translation systems

Machine translation systems (from now on MT) are essentially multilingual because their target is the "transformation" of a text written in language A into an equivalent text in language B. In this section main architectures of MT systems will be described, because each architecture sets a series of conditions over the appropriate characteristics of the inputs to the generation process.

2.2.1 Transfer systems

The basic tasks in a transfer system are analysis, transfer and generation. The analysis component produces a syntactic representation (sort of thorough) depending of the source language. This syntactic representation is the input to the transfer module whose task is to transform that representation into a closer structure to the target language. The output of the transfer module shapes the input to the generation system module which finally produces the phrase in the target language. In transfer systems, the components, inputs and outputs are strongly oriented to the source and target languages.

The main problem of the transfer systems is the almost impossibility to reuse the existing resources (transfer modules) and components in order to include new language pairs in the system. In fact, if it were necessary to increase the number of language pairs, a new system would have to be built. Generally, the great orientation of the "transfer" systems towards the target language involves great accuracy in the output, and a considerable difficulty in reusing components to include new languages in the system.

2.2.2 Interlingua based Systems

Interlingua based systems form the second great systems' paradigm of machine translation. Included into the systems based in interlingua are the "traditional" ATLAS-II [5], PIVOT [6] as much as the *knowledge-based* ones such as KANT [7] or Mikrokosmos [8]. Their defining characteristics are:

- **Unique intermediate representation**. The abstract representation, result from the analysis, "feeds" directly the generation module. This intermediate representation is the component named "interlingua".
- **Elimination of the transfer process**. The system carries out two basic tasks: analysis and generation.

The systems based on interlingua are oriented to cover the largest possible number of languages, given that the number of components that requires a system based in interlingua for n languages is 2*n, it is remarkably inferior to n*(n-1) that transfer systems require for the same number of languages.

The basic architecture of the generation in a interlingua system is shown in the next figure.

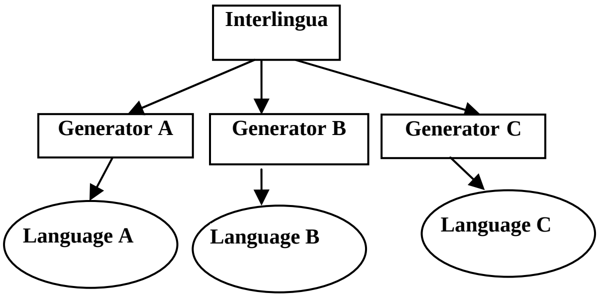


Fig. 1. Generation in interlingua systems

Interlingua based systems offer an important advantage over the transfer ones; the architecture facilitates the inclusion of new languages and are reusable. However, during the conversion process to the interlingua, it is possible that some significant grammar information for the generation may be lost, that is, the interlingua may have less information (grammatical, not conceptual) than a syntactic representation. To sum up, the systems based on interlingua offer a larger number of languages at the expense of lesser precision in the generated texts.

2.2.3 Fusion

Without any doubt, multilingualism is an added value for any generation system. The transfer-interlingua dichotomy seems to imply an opposition between precision vs. number of languages. To take advantage from every one, some transfer systems have "interlingued" their architectures to support a larger number of languages [9] [10]. The common characteristic in these systems is the existence of a deep syntactic representation that has some amount of independence from the source language. The process to combine the interlingua architecture in a transfer system requires the construction of a transfer module between the deep syntactic structure and an interlingua representation [11].

3 The UNL approach

3.1 The UNL system

UNL [12] is an artificial language designed to reproduce the content of texts written in any natural language. The UNL is provided with specifications that formally define the language. A UNL expression is an hyper graph consisting of:

- **Universal words**. They define the vocabulary of the language, i.e., they can be considered the lexical items of UNL. To be able to express any concept occurring in a natural language, the UNL proposes the use of English words modified by a series of semantic restrictions that eliminate the innate ambiguity of the vocabulary in natural languages. In this way, the language gets an expressive richness from the natural languages but without their ambiguity. Take, for example, the English word "construction" meaning "the action of constructing" and the "final product". Thus, the word "construction" will be paired with two different universal words:

  construction$_1$ → construction(icl>action)
construction$_2$ →construction(icl>concrete thing)

  where "icl" is the abbreviation for "included".

- **Relations**. These are a group of 41 relations that define the semantic relations among concepts. They include argumentative (agent, object, goal), circumstantial (purpose, time, place), logic (conjunction, and disjunction) relations, etc. For example, in a sentence like "The boy eats potatoes in the kitchen", there is a main predicate ("eats") and three arguments, two of them are instances of argumentative relations ("boy" is the *agent* of the predicate "*eats*", whereas "potatoes" is the *object*) and one circumstantial relation ("kitchen" is the *place* where the action described in the sentence takes place).

- **Attributes.** They express the semantic information resulting from the morphologic flexion and the functional elements of the phrase (auxiliary verbs, articles, etc.). They are put together with the universal words to complete their meaning when they appear in a specific context. The attributes include information about time or aspect of the event, number, polarity, modality, etc**.** In the previous sentence, attributes are needed to express plurality in the object (“potatoes”), definite reference in both the agent (“boy”) and the place (“kitchen”) and finally and special attribute denoting which UW is the head of the whole expression. (the *entry* node).

Formally, a UNL expression has the form of a semantic net, where the nodes (universal words) are linked by labeled arcs with the UNL concept relations. The graphical representation of the sentence “the boy eats potatoes in the kitchen” in UNL is shown in figure 2.

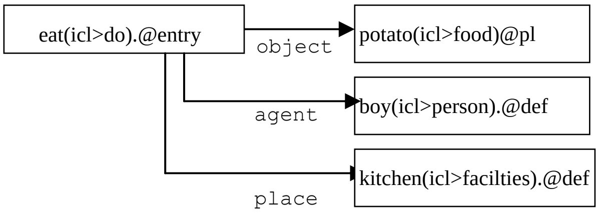


Fig. 2. Representation of a UNL expression.

This sentence is written in UNL in the following manner:

***agt*(eat(icl>do).*@entry*, boy(icl>person).*@def*)**
***obj*(eat(icl>do).*@entry*, potato(icl>food).*@pl*)**
***plc*(eat(icl>do).*@entry*, kitchen(icl>facilities).*@def*)**

3.2 Basic characteristics of UNL

The UNL system represents a generic framework for the massive generation of multilingual contents. Its main goal is the contents’ representation of a document, web page, data base, etc., in a *consensual and normalized structure* that may be transformed into a text in a natural language. The defining characteristics of the UNL system are:

a) It is a system oriented to the generation of multilingual contents. A document written in the UNL has its “own identity” and can be stored in a document data base, etc.

b) The UNL does not involve the use of specific components or tools. The tools and components, as well as the processes that may be defined to accomplish the edition and the generation in the UNL vary from one language to another. The use of the UNL only involves the standardization of the input into a generation system [13].

In spite of the emphasis given to the language generation in the system, the UNL framework includes the editing process of natural language into the UNL, named “enconversion” as well as the generation into natural languages or “deconversion” (see figure 3).

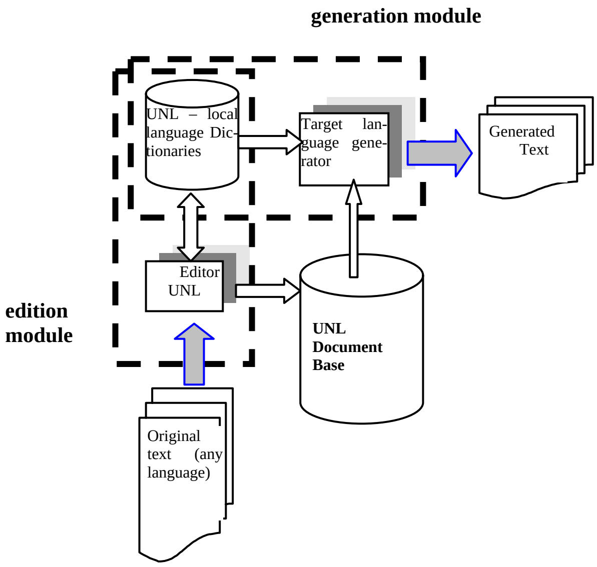

generation module
UNL – local language Dictionaries
Target language generator
Generated Text
Editor UNL
UNL Document Base
edition module
Original text (any language)

Fig. 3. Architecture of the UNL system.

The UNL is an interlingua in essence, that is, an appropriate language for the representation of the meaning in an independent way from the natural languages. The UNL is not restricted to a specific domain (as can be the KANT or Mikrokosmos interlinguas); the fact of not restricting the input in the vocabulary collection of the interlingua guarantees the UNL adaptation for the representation of contents in any language or domain.

3.3 Generation in the UNL framework.

There are several architectures for the generation of natural language from the UNL. Next, the two generation architectures within the UNL framework will be described in detail.

3.3.1 Direct Generation

The UNDL Foundation (http://www.undl.org) supplies a module that carries out the generation process through a unique process. This module is known as DeCo (standing for DeConverter). This module is completely language independent, since all the necessary grammar knowledge for the generation of the target language is included in the dictionary and the rules' set proper of the language.

Given that this module directly transforms the semantic UNL representation into the morphological realization (that is, a sentence in natural language), the dictionary must contain the best detailed information in the following aspects:

- **Grammar category and subcategories**: the more organized by hierarchies the lexical level, the better quality will be expected from the generation.
- **Argument structure** and prepositions required by verbs, nouns, and adjectives.
- **Semantic information** that may be relevant for the syntactic configuration in the target language.

With the help of the information included in the dictionary, the generation rules have, as their main task, to transform the UNL expression into a phrase in the natural language. Basically the following tasks are being carried out:

- **Matching of the UNL relations with the grammar relations of the language**. In the previous sentence, the agent of the predicate in UNL corresponds to the grammatical subject in English or Spanish.
- **"Translation" of the UNL attributes into their appropriate morphologic or syntactic realization**. For example, the attribute "plural" has to be morphologically realized as a plural noun in Spanish. The attribute "definite reference" is translated into Spanish through the insertion of a definite article. Not always there is a direct translation between UNL attributes and morphological/pragmatic information in natural languages. For instance, when dealing with time, UNL only offers three possibilities (past, present and future). It would be "competence" of the generation rules of each natural language to correctly select the tense and verbal moods applicable to the languages that do not have this kind of time system (for instance, Spanish).
- **Generation of pronouns and anaphoric expressions**. The UNL expression is devoid of anaphoric elements, all concepts in UNL should be stated explicitly. It is the task of the generation rules to insert pronouns and other anaphoric elements in the generated texts.
- **Morphologic synthesis**. Finally, generation rules should tackle aspects such as agreement between verb and subject, or between adjectives and nouns, word order or the expression of the correct verb tense.

Figure 4 shows the architecture for direct generation, there it can be seen how the "bilingual" dictionary Natural Language-UNL and the generation rules feeds the DeCo module in order to carry out the generation of UNL text into a natural language text.

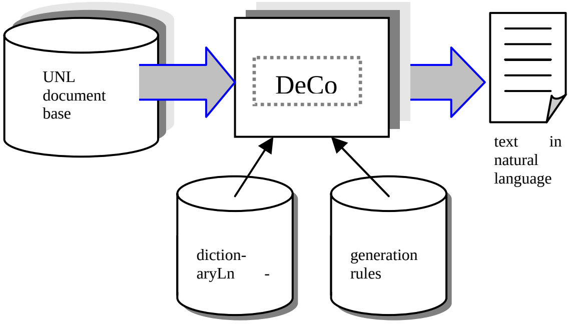


Fig. 4. Architecture of the direct generation in the UNL framework

3.3.2 Combined Generation (reuse of transfer components)

The treatment for Russian and French languages inside the UNL system is the perfect example of the *combined generation* within the UNL framework. Both teams have integrated the UNL system into their transfer systems, ETAP in Russian case [11], and Ariane for the French one [14].

These systems have chosen to reuse the available generators of the target languages and to develop an additional module that allows the conversion of the UNL representation into a friendly format through the generators of their “transfer” systems. An example of combined architecture would be exemplified in figure 5.

The so called “UNL transfer module” is with no doubt a new component to develop. However, the experience in the already mentioned systems has shown that the development costs of this module are cheaper than the costs for developing a new generator that could have the UNL code as its direct input.

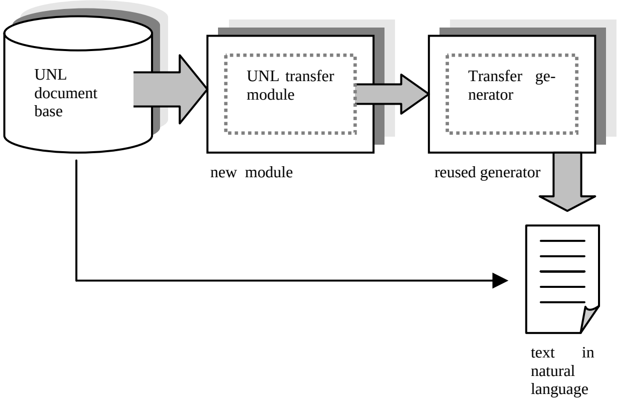


Fig. 5. Combined Generation

4 Can the UNL be a generation standard?

4.1 What is a standard?

If we try to avoid the formal definitions for "standard", it could be said that a standard is a set of rules, criteria and recommendations that allow to build a product or to design and offer a service in a proper way that assures:

- The universalization of the work, that is, a unique way of doing something that at the same time can be independently evaluated, no matter who does it or when.
- The quality. When products or services have been carried out following a standard, there is a certainty that the processes are well implemented and the product quality is not at risk.
- The assessment of the product or service provisions, meaning that it could be determined through a unique way, when a product or a service fulfills the specifications it has been designed and built for.

Many more could be enumerated, but we are focusing in these three that may be the most intuitive. As it has already been mentioned, the lesser development of some products (in this case, language generators) is due to the lack of standards that could assure these characteristics. The diversification and extremely disperse casuistic of the inputs to a generator cause that the output become the only way to assess it to establish a subjective evaluation.

Although there are some researchers that have not neglected this side of the generation [15], this standard has not been yet established, neither formally nor de facto.

4.2 The UNL as a standard

Technically speaking, the lack of uniformity in the inputs to language generators is almost the only reason that restrains a bigger development. Therefore, the support systems to multilingual services see their action limited only to specific languages where translation services may be offered, either automatic or not. However, the language expansion is an unapproachable road with these methods. If the input to language generators is not standardized, this problem will not be solved in a global way. The only standardization would then be the choice of a content support that could express itself in a unique way, with a specific language. Actually, this concept has existed for many years, and it is the Interlingua concept. It is within this context where the UNL can play a role. The UNL has not been conceived as an interlingua, but it can be used as one. The interlinguas had their historic moment when they faced the same problems as the other systems created for machine translation during the 80's. At the beginning of the 90's it was clear that the subject of the languages was much more complex than it seemed during the technological development of the 80's and the exaggerated optimism of the time.

It is not the purpose of this paper to describe the economic advantages of an interlingua over the traditional systems of machine translation regarding many languages (a traditional machine system requires 90 systems to support 10 languages, while one based in interlingua requires only 20). In fact, the crossing point between systems takes place at three languages. For more than three languages interlingua is cheaper.

However, historic matters at the beginning of the 90's buried the interlinguas (mainly those developed in Japan and the USA) because while the interlingua based systems were not well defined, the "transfer" machine translation systems began to offer more positive results. Even so, within the group of language technologies, machine translation became kind of discredited. At the end of the 90's, the United Nations opted for models based on interlingua approximations to define the multilingual support systems for the Internet. The result is the today's named UNL, already described in this chapter. Apparently, it would be the ideal system to solve the problem of the absence of a standard input to language generators. Nevertheless, a standard is something else than a technological solution. It could be summarized like this: a standard is evaluated through the maturity concept that to sum up means that it would be associated to the organized and organizational maturity, that is, there has to be an organization behind the standard that may be able to maintain, modify, allow the study of its acceptance and real use for it, and other factors. Currently, it could be said that the UNL has weak and strong points to formally become a standard [16]:

Weak points:

- Relatively recent technology
- Not too much implemented
- Quality system not implemented

Strong points:

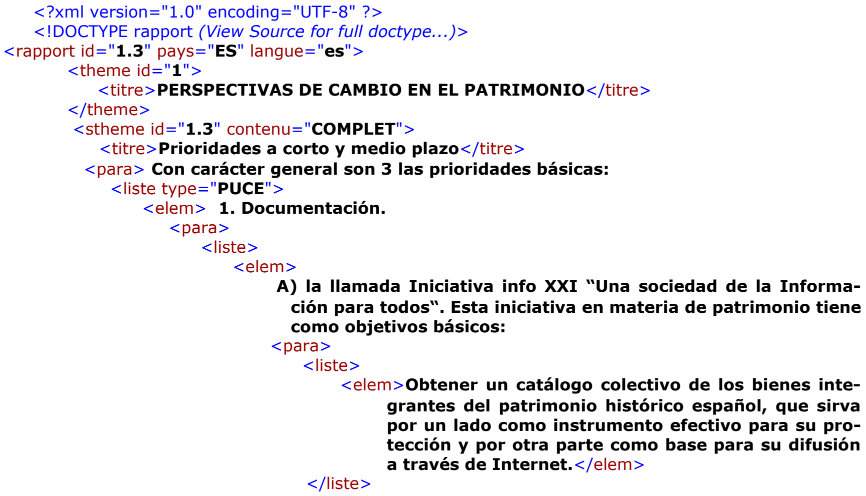
- Worldwide organization behind (dissemination assured)
- Business expectations increased by the incorporation of minority languages
- Quality system defined

However, independently of the global factors, the technological approach is nowadays the only one able to solve the problem of automatic multilingual generation systems. Regarding the business approach, the expansion of multilingual systems in the Internet requires much more than traditional systems of machine translation. This is why the UNL is not just an interlingua, but a language to support knowledge repositories, different ontological approaches, and other matters. Summarizing, the UNL (or something similar to it) is necessary and needed by others.

5 A real experience: HEREIN and UNL

5.1 |Herein and standardization of form and structure.

The Herein system (IST-2000-29355) [17] is a perfect example of a massively multilingual environment. It constitutes an Internet-based facility for improving cultural heritage management methods at the European level. Among the main tasks of the project, participant countries must compose a report providing detailed information about all aspects regarding cultural heritage.



```
     <?xml version="1.0" encoding="UTF-8" ?>
     <!DOCTYPE rapport (View Source for full doctype...)>
 <rapport id="1.3" pays="ES" langue="es">
          <theme id="1">
               <titre>PERSPECTIVAS DE CAMBIO EN EL PATRIMONIO</titre>
          </theme>
           <stheme id="1.3" contenu="COMPLET">
               <titre>Prioridades a corto y medio plazo</titre>
             <para> Con carácter general son 3 las prioridades básicas:
                 <liste type="PUCE">
                      <elem>  1. Documentación.
                          <para>
                               <liste>
                                    <elem>
                                          A) la llamada Iniciativa info XXI "Una sociedad de la Informa-
                                            ción para todos". Esta iniciativa en materia de patrimonio tiene
                                            como objetivos básicos:
                                         <para>
                                              <liste>
                                                   <elem>Obtener un catálogo colectivo de los bienes inte-
                                                          grantes del patrimonio histórico español, que sirva
                                                          por un lado como instrumento efectivo para su pro-
                                                          tección y por otra parte como base para su difusión
                                                          a través de Internet.</elem>
                                               </liste>
```

Fig. 6. Example of Spanish contents in XML structure.

Due to the large number of countries participating in the project (almost 30) and the huge variety of topics that comprise cultural heritage (legislation, preservation, dissemination, etc.), there was an urgent need to standardize both the format and the structure of the contents that each country should provide. A definite structure was established and every country involved in Herein had to integrate its particular contents into such structure. Eventually, this structure turned out to be a de-facto standard for the description of the cultural heritage issues of a country since it met the two basic conditions for a de-facto standard, which are:

a) it has been actually used in a real and working environment

b) its application has been universalized: it proved valid for almost 30 countries.

Furthermore, a de-facto standard requires a support for its physical representation. In Herein's case, XML was the chosen support. The standardization of format/structure is twofold in Herein: structure has been normalized as a de-facto standard, whereas format has been normalized with a canonical standard (XML). Figure 6 shows the appearance of a typical report in the Herein project.

However, the contents and their structuring are just one side of the problem in the HEREIN system, the other side is the verbalization of such contents: that is, the linguistic aspect in Herein.

If the contents' side has been solved, the linguistic aspect has not. Although there are almost 30 countries involved in the project, the Herein web site and produced resources are far form being truly multilingual: only three languages are official (English, French, and Spanish), therefore all documents and resources created in Herein can only be accessed in these languages.

The reasons for such a dramatic reduction of languages are simple and straightforward:

a) Translation costs: If HEREIN were a really multilingual environment, one document of a given country will require around 24 translation into the other involved languages. For all documents of all languages, the number of required translations will be 25*24, that is, 600 translation works. Providing that only the Spanish national report counts with 10.000 words, costs for translating the Spanish report into the other languages ascends to the translation costs of 240.000 words.

b) The availability of translators in all pairs is not the same. Obviously, availability of translators for the pair English – French is higher than availability of translators for Dutch – Croatian, which can be really difficult to find.

Both reasons are enough for desisting from human translation in massively multilingual environments.

There is only one alternative to this approach, and it is the use of an interlingua. Previously we have briefly described an interlingua as "a common intermediate representation" between languages, and have postulated two conditions that interlinguas should meet, namely:

- Independence from any natural language.
- Same semantic expressiveness as a natural language.

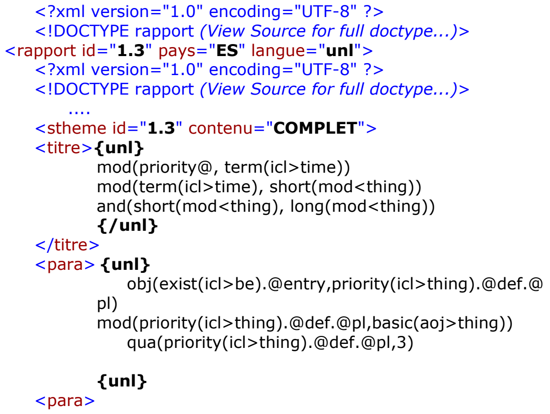
The UNL takes these two conditions as its defining characteristics. The elements that compose the UNL are all based on semantic notions, detached from any residue of morpho syntactic categories found in natural languages. These elements and the way to compose them in order to form valid and meaningful UNL expressions are completely defined and formalized in the UNL specifications [12]. But the main potential of the UNL for achieving the same expressiveness of a natural language lies in its vocabulary (the universal words). The UNL profits from the richness of natural language vocabulary (universal words are based on English lexical items) while devising a system of semantic restrictions that eliminate the ambiguity and vagueness inherent to lexical units of natural languages. In this case, the UNL perfectly fits in the definition of an interlingua or a "pivotal language".

Alongside with its adequacy for being used as an interlingua, the UNL also satisfies the conditions for its qualification as a potential standard for generation. These two characteristics have been already exploited in the Herein project, as it will be shown in the next section.

5.2 The UNL approach in HEREIN

As an initiative of the Ministry of Education and Culture of the Spanish government, representative institution of the Herein contents in the Spanish language, and in collaboration with the Spanish Language Center (representative and responsible of the Spanish language in the UNL program), the complete report of the Spanish cultural contents was codified into the UNL.

This UNL code has been capable of being embedded into the XML structure common to all reports, as if the UNL were another "natural language"(see figure 7).

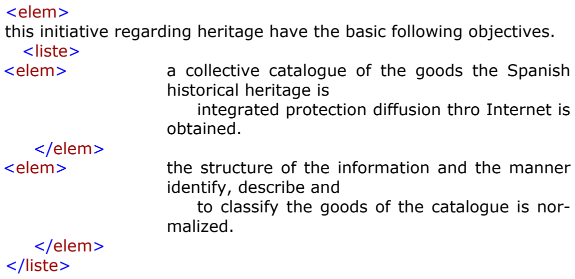


```
    <?xml version="1.0" encoding="UTF-8" ?>
    <!DOCTYPE rapport (View Source for full doctype...)>
<rapport id="1.3" pays="ES" langue="unl">
    <?xml version="1.0" encoding="UTF-8" ?>
    <!DOCTYPE rapport (View Source for full doctype...)>
        ....
    <stheme id="1.3" contenu="COMPLET">
    <titre>{unl}
            mod(priority@, term(icl>time))
            mod(term(icl>time), short(mod<thing))
            and(short(mod<thing), long(mod<thing))
            {/unl}
    </titre>
    <para> {unl}
                obj(exist(icl>be).@entry,priority(icl>thing).@def.@
            pl)
            mod(priority(icl>thing).@def.@pl,basic(aoj>thing))
                qua(priority(icl>thing).@def.@pl,3)

            {unl}
    <para>
```

Fig. 7. UNL text embedded in a XML document

The difference lies in the fact that the aforementioned contents can be “captured” by the generators of any language. After generation of the UNL, the corresponding contents (now in the “form” of a natural language) will be reinserted in the XML structure of the document. The result, at the internal level, is visualized as shown in figure 8 and 9 for the English and Russian language.

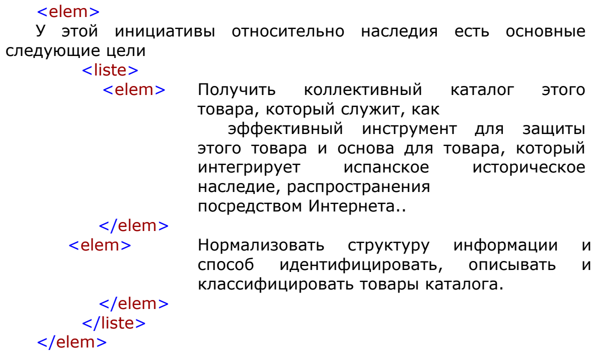


```
<elem>
this initiative regarding heritage have the basic following objectives.
   <liste>
<elem>               a collective catalogue of the goods the Spanish
                     historical heritage is
                          integrated protection diffusion thro Internet is
                     obtained.
     </elem>
<elem>               the structure of the information and the manner
                     identify, describe and
                          to classify the goods of the catalogue is nor-
                     malized.
     </elem>
</liste>
```

Fig. 8. Output of English generator

6 Conclusions

It seems clear that the architecture of an “interlingua” system (based on a format of a unique input to all generators) supports the idea of formally defining the input to develop the “generator” component, according to some precise specifications (not existing until now). This precise specification would be the base of a standard generators development, creating an environment that may allow carrying out tests of reliability for this component, essential in the generation of multilingual contents.



```
   <elem>
   У этой инициативы относительно наследия есть основные
следующие цели
          <liste>
             <elem>     Получить коллективный каталог этого
                        товара, который служит, как
                           эффективный инструмент для защиты
                        этого товара и основа для товара, который
                        интегрирует испанское историческое
                        наследие, распространения
                        посредством Интернета..
             </elem>
        <elem>          Нормализовать структуру информации и
                        способ идентифицировать, описывать и
                        классифицировать товары каталога.
             </elem>
          </liste>
   </elem>
```

Fig 9: Output of Russian Generator

Given the characteristics of the UNL (independence from the natural languages and adaptation to express any content of a natural language) and the possible integration of the UNL system with any other existing generation system, it is appropriate to propose the UNL as a standard for the normalization of the inputs to natural language generation systems.

A standard must be supported by an organization that can assure its stability and maintenance. In this case, there is an organization that fulfills these requirements: the UNDL Foundation under the protection of the United Nations. Finally, it is important to mention that the UNL has been recently qualified as the first software patent of the United Nations.

References

1. Dale R., Di Eugenio B, and Scott D. (1998). “Introduction to the special issue on Natural Language Generation”. *Computational Linguistics*, Volume 24, number 3.
2. Whitelock, P. (1995): “Linguistics and computational techniques in machine translation system design”. UCL Press Limited. London.
3. Reiter E, y Dale R. (1995). “Building applied natural language systems”. Cambridge University Press.
4. Ballim, A and Payota V. (2001). “Weighted Semantic Parsing: A robust approach to interpretation of Natural Language Queries”. *Flexible Query Answering Systems*. ed Larsen H., et al. Physica Verlag.
5. Uchida, H. (1989). “ATLAS-II: A machine translation system using conceptual structure as an interlingua”. *Proceedings of the Second Machine Translation Summit*. Tokyo, Japan.
6. Muraki, K. (1989). “PIVOT: Two-phase machine translation system”. *Proceedings of the Second Machine Translation Summit*. Tokyo, Japan.
7. Nyberg E and Mitamura T. (1992). “The KANT System: Fast, Accurate, High-Quality Translation in Practical Domains”. *Proceedings of COLING-92*: 15th International Conference on Computational Linguistics. Nantes, France.
8. Beale, S., S. Nirenburg and K. Mahesh. (1995). “Semantic Analysis in the Mikrokosmos Machine Translation Project”. *Proceedings of the 2nd Symposium on Natural Language Processing*. Bangkok, Thailand.

9. Aikawa T, Melero M, Schwartz L and Wu A. (2001). “Multilingual Natural Language Generation”. *Proceedings of MT Summit VIII*. Santiago de Compostela, Spain.
10. Boguslavsky I, Frid N, Iomdin L, et al. (2000). “Creating a Universal Networking Language Module within an Advanced NLP System”. *Proceedings of COLING 2000: 18th International Conference on Computational Linguistics*. Saarbrucken, Germany.
11. Lavoie B, Kittredge R, Korelsky T; Rambow O. (2000). “A Framework for MT and Multilingual NLG Systems Based on Uniform Lexico-Structural Processing”. *Proceedings of 6th Applied Natural Language Processing Conference*. ACL, 2000 Seattle, USA.
12. Uchida, H. (2002). “The Universal Networking Language. Specifications”. http://www.undl.org.
13. Boitet C and Sérasset, G. (2000). “On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter”. *Proceedings of COLING 2000: 18th International Conference on Computational Linguistics*. Saarbrucken, Germany.
14. Boitet C and Sérasset G. (1999). “UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction”. *Machine Translation Summit 99*. Singapore.
15. Hovy E. ; Reeder F. eds. (2000). *Proceedings of the International Workshop on Machine translation Evaluation*. AMTA 2000. Cuernavaca, Mexico.
16. Cardeñosa J. and Tovar E.; "A descriptive structure to assess the maturity of a standard. Application to the UNL System". *2nd IEEE conference of standardization and innovation in information technology*. Boulder. Colorado. USA. International Centre for standard research. ISBN: 0-7803-9817-3
17. HEREIN Project (IST-2000-29355). Final Report. European Commission.

FOUNDATIONS

The UNL Distinctive Features: Inferences from a NL-UNL Enconverting Task

Ronaldo Teixeira Martins,[1] Lúcia Helena Machado Rino,[2]
Maria das Graças Volpe Nunes,[3] Osvaldo Novais Oliveira Jr.[4]

[1]Núcleo Interinstitucional de Lingüística Computacional - NILC
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
ronaldo@nilc.icmc.sc.usp.br

[2]Departamento de Computação - Centro de Ciências Exatas e de Tecnologia - UFSCar
Rod. Washington Luiz, km 235 - Monjolinho - 13565-905 - São Carlos, SP, Brazil
lucia@dc.ufscar.br

[3]Instituto de Ciências Matemáticas e da Computação (ICMC) - Universidade de São Paulo
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
mdgvnune@icmc.sc.usp.br

[4]Instituto de Física de São Carlos (IFSC) - Universidade de São Paulo
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
chu@ifsc.sc.usp.br

Abstract. This paper reports on the distinctive features of the Universal Networking Language (UNL). We claim that although UNL expressions are supposed to be unambiguous, UNL itself is able to convey vagueness and indeterminacy, as it allows for flexibility in enconverting. The use of UNL as a pivot language in interlingua-based MT systems is also addressed.

1 Introduction

Machine Translation (MT) is one of the most controversial subjects in the field of natural language processing. Researchers and developers are often at odds on issues concerning MT systems approaches, methods, strategies, scope, and their potentialities. Dissent has not hindered, however, the establishment of tacit protocols and core beliefs in the area. It has often been claimed that:[1] (1) fully automatic high-quality translation of arbitrary texts is not a realistic goal for the near future; (2) the need of some human intervention in pre-edition of the input text or in post-edition of the output text is mandatory; (3) source language should be rather a sublanguage, and the input text should be domain- and genre-bounded, so that the MT system could cope with natural language ambiguity; (4) the transfer approach is more feasible than the interlingual one, since the latter, albeit more robust and economic, is committed to the

[1] Most of these assumptions can be extracted from the Survey on the State of the Art in Human Language Technology (Cole et al., 1995). Of special interest are the articles concerning multilinguality by Martin Kay (8.1, 8.2) and Christian Boitet (8.3, 8.4).

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 27–37.

somewhat insurmountable task of designing a perfect (universal) language, comprising any other one; (5) common sense and general knowledge on both the source and the target cultures are as important as linguistic information, like in Knowledge-Based Machine Translation Systems (Nirenburg et al., 1992); 6) existing human translations can be used as a prime source of information for the production of new ones, similarly to the Example-Based Machine Translation Systems (Furuse and Iida, 1992); 7) existing MT systems are not appropriate to monolingual users, although they can be used to facilitate, speed up or reduce the costs of human translation, or to produce quick and cheap rough translations that may help the users to get a very broad idea of the general subject of the text.

Many authors obviously do not endorse all the listed statements, specially the fourth one. Hozumi Tanaka (1993), for example, argues in favor of the interlingua-based approach, and so do the research and development groups involved in interlingua-based systems, such as ULTRA (Farwell and Wilks, 1993), KANT (Mitamura et al., 1993), or PIVOT (Okumura et al., 1993). These works, however, rather confirm the very general observation that commercially available MT systems (e.g., SYSTRAN, VERBMOBIL, DUET (Sharp), ATLAS I (Fujitsu), LMT (IBM), METAL (Siemens)) are primarily transfer-based.

The most serious arguments against the interlingua approach concerns its alleged universality and excessive abstractness (Hutchins and Somers 1992). In order to cope with multilinguality, the interlingua should put aside language-dependent structures (such as the phonological, morphological, syntactical and lexical ones) and work at the logical level, which is supposed to be shared by human beings. Even at such uppermost level, however, there seems to be cultural differences. Eco (1994) reports, for instance, the case for Aymara, a South-American Indian language which would have three truth values, instead of the two "normal" ones. Furthermore, it has been said that, even if one comes to find this kind of perfect language, it would be so abstract that it would not be cost-effective, since the tools for departing from natural language and arriving at the logical representation would be excessively complex.

In what follows, we present some extra evidence towards the feasibility of interlingua-based MT. The Universal Networking Language (hereafter, UNL), developed by Uchida et al. (1999), brings some distinctive features that may lead to overcome some of the bottlenecks frequently associated to the interlingua approach. Although UNL was not designed as an interlingua, and MT is only one of the possible uses for UNL, it has been claimed that multilingual MT systems can use UNL as a pivot language. In this paper, some of the distinctive features of UNL are analyzed. We build upon the experience in developing the Brazilian Portuguese (hereafter, BP) UNL Server, a bilingual MT system for translating Portuguese into UNL and vice-versa.

This paper is organized as follows. Section 2 provides a brief introduction to the UNL approach and some of its premises. In Section 3 we describe an experiment in which human subjects were asked to enconvert sentences from Portuguese into UNL. Section 4 brings the general results of the experiment. One of them is specially addressed in Section 5. Some issues arising from the results are presented in Section 6. Conclusions are stated in Section 7. The reader is supposed to have previous information on the UNL Project and knowledge on UNL Specification (at http://wwww.unl.ias.unu.edu) is considered mandatory.

2 The Universal Networking Language

The Universal Networking Language (UNL) is "an electronic language for computers to express and exchange every kind of information" (Uchida et. al., 1999, p. 13). According to the UNL authors, information conveyed by each natural language (NL) sentence can be represented as a hyper-graph whose nodes represent concepts and whose arcs represent relations between concepts. These concepts (called Universal Words or simply UWs) can also be annotated by attributes to provide further information on the circumstances under which they are used.

In this context, UNL is not different from the other formal languages devised to represent NL sentence meaning. Its structure is said to suffice to express any of the many possible meanings conveyed by any sentence written in any NL. This does not mean, however, that it is able to represent, at the same time, all the possible meanings conveyed by the very same NL sentence. Instead, UNL is able to represent each of them independently, and it is by no means able to provide a single structure coping with all of them. In this sense, there will never be a single UNL expression that completely suffices the meaning correspondence to a NL sentence. Or else: no UNL expression will be ever completely equivalent to a NL sentence, since the latter, but not the former, will allow for ambiguity.

In the following section, we report on results of a BP-UNL enconverting task that has been carried out by BP native speakers. In this experiment, we observe evidences that BP sentences must be disambiguated in order to be represented as UNL expressions.

3 The Experiment

In August 2001, we carried out an experiment on BP-UNL enconverting that involved 31 BP native speakers, all of them graduate and postgraduate students. Most of them (over 95%) were Computer Sciences students, aging 21 to 42 years old (90% of them were under 30 years old).

The experiment was split into training (steps 1-4) and test sessions (step 5), as follows: 1) a very general description of the UNL structure; 2) a general presentation of the definitions provided for five relation labels by the UNL Specification (1999), namely, ‘agt’ (agent), ‘cag’ (co-agent), ‘obj’ (affected thing), ‘cob’ (affected co-thing), and ‘ptn’ (partner); 3) an individual exercise on the use of the presented relation labels, in which subjects were asked to identify 50 different relations appearing in different BP sentences, indicating the corresponding UNL relation labels; 4) a public discussion on the exercise results; and 5) a final individual test in which subjects were asked again to identify 30 different relations appearing in different BP sentences, through their correspondence with the very same set of UNL relation labels. In Step 3 and 5, the subjects had also the option of pinpointing the impossibility of identifying either a relationship or its corresponding relation label, by choosing a “catch all” alternative (see option (a) in Figure 1). This exercise aimed at providing the means for the subjects to understand and explore BP-UNL enconverting, concerning the relation labels identification. This was then reinforced in Step 4, which was supervised by a

UNL specialist. As it can be observed, these steps aimed at Step 5, the actual BP-UNL assignment, focusing on specific relation labels. In this step, some of the BP sentences presented to the subjects in Step 3 have been replicated.

Altogether, this experiment has taken 1 hour and 40 minutes, considering a 20-minute interval between the training and test sessions. Steps 1 and 2 have last 20 minutes, and so has Step 3 alone. Step 4, the longest one, has taken 40 minutes. Step 5, the actual test, has taken another 20 minutes. The interval between training and test aimed at allowing for the subjects settling on UNL specification, since test has been totally unsupervised. This also justifies our replication of some of the BP sentences used in training.

An English version of the task proposed in Step 3 is presented in Figure 1 below.

Considering the information presented in the first part of this experiment, identify the following:

1) If the relation depicted between the words signaled in each of the sentences below belongs to the five-relation set discussed previously; and

2) If so, which relation label would most suitably describe the involved relationship.

Use, for reference, the following code:

a) if NO label describes the relationship between the signaled words;
b) if the label AGT (agent) is the most suitable one;
c) if the label CAG (co-agent) is the most suitable one;
d) if the label COB (affected co-thing) is the most suitable one;
e) if the label OBJ (affected thing) is the most suitable one;
f) if the label PTN (partner) is the most suitable one.

Fig. 1. Instructions for identifying and classifying relations.

The 30-sentence set used in the test session, along with its corresponding English translation, is shown in Figure 2.

SENTENCES

1. A crise quebrou o empresário >> ???(quebrou, crise)
The crisis broke the business man. >> ???(broke, crisis)

2. A crise quebrou o empresário >> ???(quebrou, empresário)
The crisis broke the business man. >> ???(broke, business man)

3. A farsa acabou. >> ???(acabou, farsa)
The farce is over. >> ???(is over, farce)

4. A neve caía lentamente. >> ???(caiu, neve)
Snow felt slowly. >> ???(felt, snow)

5. Alugam-se casas. >> ???(alugar, casa)
Houses are rented (also: *Someone rents houses*) >> ???(are rented, houses)

6. Choveu canivete ontem. >> ???(choveu, canivete)
It rained knives yesterday >> ???(rained, knives) (Brazilian Idiom)

7. João jogou o vaso com Maria contra Pedro. >> ???(jogou, Maria)
John threw the bowl with Mary against Peter. >> ???(threw, Mary)

8. João jogou o vaso com Maria contra Pedro. >> ???(jogou, Pedro)
John threw the bowl with Mary against Peter. >> ???(threw, Peter)

9. João lutou com Maria para vencer a doença. >> ???(lutou,Maria)
John fought with Mary to win the disease. >> ???(fought, Mary)

10. João não teve filhos com Maria. >> ???(ter, João)
John did not have children with Mary. >> ???(have, John)

11. Maria esqueceu o dia do aniversário da filha. >> ???(esquecer, dia)
Mary forgot her daughter's birthday. >> ???(forgot, birthday)

12. Maria foi despedida. >> ???(despedir, Maria)
Mary was fired. >> ???(fire, Mary)

13. Maria lembrou Pedro do horário. >> ???(lembrou, horário)
Mary remembered Peter about the schedule. >> ???(remembered, schedule)

14. Maria morreu com a falta de oxigênio.. >> ???(morreu, falta)
Mary died with the lack of oxygen. >> ???(died, lack)

15. Maria namorou Pedro. >> ???(namorou, Maria)
Mary flirted (with) Peter. >> ???(flirted, Mary)

16. Maria não foi ao cinema com a vizinha. >> ???(foi, vizinha)
Mary did not go to the cinema with her neighbor. >> ???(go, neighbor)

17. Maria não quis matar Pedro! >> ???(matar, Maria)
Mary did not intend to kill Peter. >> ???(kill, Mary)

18. Maria não se sentiu bem. >> ???(sentir, Maria)
Mary did not feel well. >> ???(feel, Mary)

19. Maria nunca conquistou Pedro. >> ???(conquistou, Pedro)
Mary never conquered Peter. >> ???(conquered, Peter)

20. Maria parece cansada. >> ???(parece, Maria)
Mary looks tired. >> ???(looks, Mary)

21. Maria se esqueceu de João. >> ???(esquecer, João)
Mary forgot John. >> ??(forgot, John)

22. Maria se matou. >> ???(matou, Maria)
Mary killed herself. >> ???(kill, Mary)

23. O filme deu origem a muitas controvérsias. >> ???(deu, filme)
The movie raised many controversies >> ???(raised, movie)

24. O frio congelou o pássaro. >> ???(congelar, frio)
The cold froze the bird. >> ???(froze, cold)

25. O medo da morte provoca insônia. >> ???(provoca, medo)
Fear of death causes insomnia. >> ???(causes, fear)

26. O pai com os filhos matou a mãe. >> ???(matou, filhos)
The father with the children killed the mother. >> ???(killed, children)

27. O pássaro congelou com o frio. >> ???(congelar, frio)
The bird froze (i.e., was frozen) with the cold. >> ???(froze, cold)

28. Os carros se chocaram na estrada. >> ???(chocaram, carros)
The cars crashed each other on the road. >> ???(crashed, cars)

29. Pedro se parece com a mãe. >> ???(parece, mãe)
Peter looks like his mother. >> ???(looks, mother)

30. Precisa-se de funcionários. >> ???(precisar, funcionários)
Employees are needed. (also: *Someone needs employees*) >> ???(need, employees)

* Students were presented only to the original Brazilian Portuguese sentence. In the translation from Portuguese into English we tried to preserve the Portuguese syntactic structure as often as possible, even when the resulting English sentence sounds agrammatical.

Fig. 2. Test corpus

4 Results

The results of the experiment as summarized in figure 3.

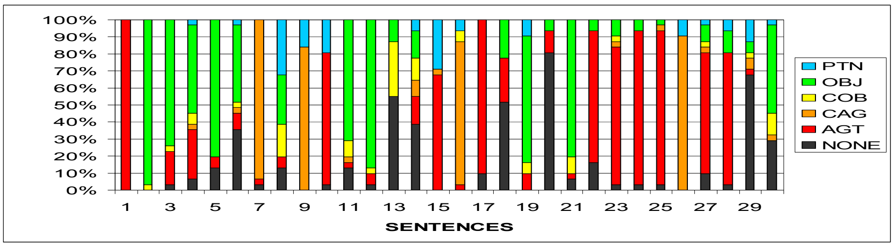


Fig. 2. Distribution of BP-UNL enconvertings by subjects, with respect to the 5-relation labels set

Figure 4 below groups the results according to the agreement among enconverters.

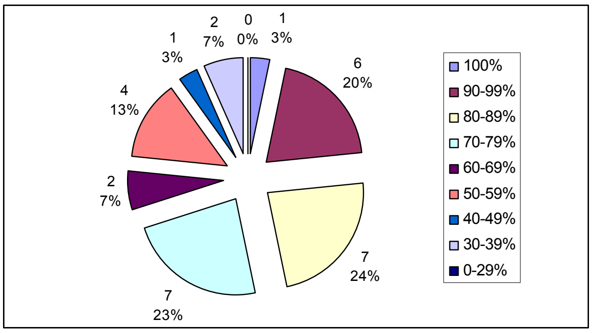


Fig. 3. Agreement among enconverters

A single relation (between "crise" (*crisis*) and "quebrou" (*to break*) in sentence 1: *"A crise quebrou o empresário" (= The crisis broke the business man*) led to an agreement of 100% among enconverters: they all used the 'agt' label in this case. There was an agreement between 90% to 99% on labeling relations in 6 sentences. Enconverters also agreed between 80% to 89% in assigning labels in 7 sentences. Other 7 sentences involved 70% to 79% agreement. In the remaining 9 sentences, agreement among enconverters was lower than 70%.

5 Case Study: Sentence 14

Sentence 14 ("Maria morreu com a falta de oxigênio." (literally: "Mary died with the lack of oxygen.") can be taken as a typical example of those involving considerable disagreement among enconverters. The relation between the verb "morreu" (to die) and the noun "falta" (lack) was encoded in varied ways, as follows: a) as an agent one (16%); b) as an object one (16%); c) as a co-object one (13%); d) as a co-agent one (10%); e) as a partner one (6%); and f) as none of the previous five relations (39%).

The unavoidable issue that follows from the above is why UNL labels were used in such apparently fuzzy way. Several reasons could be pinpointed here: a) the lack of expertise (or even of attention) of human enconverters', for they could not have had enough knowledge of language, or motivation, to carry on the experiment (although they are BP native speakers and seemed to be willingly helpful and interested in participating); b) the lack of clarity of the UNL Specification itself, even though there had been considerable discussion in the training session, for the problems posed by the enconverters to be tackled; c) the structure of the experiment itself, which was indeed too brief and too shallow to properly evaluate the human enconverters' performance; and, finally, d) the ambiguity of test sentences.

The analysis of the enconverters' choices certifies that disagreements are due to the latter point. Although it is unlikely for a BP speaker to say that 14 above, out of context, could have many different colliding meanings, the experiment has proved that apparently unambiguous sentences are unambiguous only apparently. Although eventually invisible, NL vagueness and indeterminacy would be pervasive in ordinary language,

Actually, none of the labels assigned to the relation between "morreu" (to die) and "falta" (lack) in sentence 14 could be considered wrong. The lack of oxygen could be understood in many distinct ways, such as:

a) an agent ("agt"), or the "initiator of the action" of "Mary dying" (or "killing Mary");

b) a co-agent ("cag"), or a "non-focused initiator of an implicit event that is done in parallel", in the sense it was not the lack of oxygen that killed Mary but either b.1) the situation (or the person) that has provoked the suppression of Mary's air supply or, in a more precise way, b.2) the reaction provoked (mainly in the brain) by the lack of oxygen;

c) an object ("obj") for the event described by "dying", since it is somehow "directly affected" by it, as the conclusion that the oxygen was lacking might be said to come directly from the fact that Mary died, otherwise no one would perceive that oxygen was lacking;

d) an affect co-thing ("cob"), or as being "directly affected by an implicit event done in parallel", if the observation that the oxygen was lacking were said not to come directly from the fact that Mary died, but from the fact that her lungs stopped working, which caused her to die;

e) a partner ("ptn"), for it could be somewhat "an indispensable non-focused initiator" of the action of "Mary dying", as if the main responsible for Mary's death was Mary herself (or someone else) that turned the oxygen suply off.

Besides such illustrations, many other relations can be said to hold between ‘lack of oxygen’ and ‘die’, namely, "met" (method), "man" (manner), "ins" (instrument), and "rsn" (reason), all easily applicable to such a case.

Such a variety proves that sentence 14 was indeed vague. The syntactic relation between the BP verb and its adjunct can convey many different semantic cases. Nevertheless, the UNL expression – whatever it may be – will have, in turn, a single interpretation, because relation labels are not supposed to overlap. The relations agt(die,lack), cag(die,lack), cob(die,lack), obj(die,lack), ptn(die,lack), although applicable to that very same NL sentence, are expected to label different (albeit related) phenomena. Indeed, to say agt(die,lack) is not the same as to say cag(die,lack) or ptn(die,lack). No intersection between these relations is envisaged in the UNL Specification, since they are meant to be exclusive[2].

This makes clear that the UNL specification forces filtering possible interpretations for NL sentences, in the sense a UNL expression must provide a completely unambiguous representation for the source sentence. As a matter of fact, although UNL is intended to be as expressive as any NL, UNL expressions cannot convey, at least at the relation level, NL vagueness and indeterminacy. Like any other formal language, UNL is committed to disambiguate NL sentences and, hence, to impoverish their semantic power.

Nevertheless, in no one of the above situations it is possible to say that a relation label is wrong, or that is completely inappropriate, although some of them may seem really unlikely to hold, depending on the context. The point is that the meaning of the sentence "Mary died with the lack of oxygen." is not encapsulated in the sentence itself but it is built out from the reading (and hence from the analysis) made by human enconverters. Since different enconverters have different underlying assumptions during their readings, the same BP phenomena can naturally imply different interpretations, which in turn lead to distinct UNL labeling. To conclude, it seems impossible to prevent subjectivity (or context-sensitiveness, or else, enconverter-sensitiveness) at that extent, no matter how univocal NL sentences seem to be.

6 Consequences

From the above it is possible to state that UNL should not seek for a straightforward correspondence between UNL expressions and NL sentences. It would be useless. As meaning is not encrypted in NL sentences but build through the analysis process, different enconverters will unavoidably propose different UNL expressions for the very same NL sentence and many of these different expressions are legitimate.

Due to structure of UNL, UNL expressions cannot replicate NL sentence vagueness and indeterminacy. Enconverters are obliged therefore to choice a single interpretation among many different possible ones. This choice will be inevitably affected by the enconverters' context, which will be unreplicable itself by other enconverters. Once all these enconvertings will be valid, in the sense they are context-motivated,

[2] Accordingly, it is worthy to observe that the individuality of relations seems to be less strong when we consider other UNL relation labels set, e.g., that comprising "qua" (quantity), "nam" (name) and "pos" (possessor), which seems to be, to some extent and context, replaceable by "mod" (modification), implying that the latter can quite feasibly be at an uppermost level in a relation hierarchy. The same could be said of "met" (method) and "ins" (instrument), which seem to be under the scope of "man" (manner). Conversely, this does not mean that "mod" comprises any of "qua", "nam", or "pos", or that "man" embeds "met" and "ins". Instead, it does mean that both "mod" and "man" seem to share a comprehensive set of features with the relations that they replace. This is not the case of "agt", "cag", "cob", "obj", and "ptn", which seem to be in a more outstanding opposition.

there will never be a one-to-one mapping between NL sentences and UNL expressions.

Accordingly, correctness, in UNL, instead of representing a (impossible) single possibility of enconverting, should rather be considered as fidelity to enconverters' intentions. UNL should clearly state that it would be up to the (human and machine) enconverter to decide what should the UNL representation be for a NL sentence. That is to say, the object of the UNL representation should be considered not exactly the meaning conveyed by the NL sentence but the *interpretation inferred by the enconverter from the use of that NL sentence in the enconverter's specific context.*

The fact that there could be more than a single (and adequate) UNL expression for the same NL sentence implies that UNL allows for flexibility in the enconverting process, although the UNL expression itself is not supposed to be flexible. It is up to the enconverter, and not the UNL specification itself, to decide which of the many possible interpretations is to be represented by a UNL expression. This is a significant UNL distinctive feature. Most formalisms do not allow for such variability and postulate that there should be a biunivocal relation between NL and its artificial representation. Otherwise, the formal representation would keep mirroring NL vagueness and indeterminacy, resulting useless.

The problem here is how to assure that enconverting flexibility will not prevent UNL from being a machine tractable language. As far as UNL expressions are dependent on the enconverter, there could be uncontrolled variations, which could blow out UNL into many different (and maybe mutually unintelligible) dialects.

This problem can be divided into two parts: 1) how to be sure that the UNL expression represents indeed what is intended by the enconverter; and 2) how to be able to generate, from such varied UNL expressions, NL grammatical sentences.

The first question is somewhat an educational problem. There are obviously misunderstandings and misuses of many relations. To say that it is up to the enconverter to decide which label should be used is not to say that the enconverter can do whatever he/she/it wants. The UNL Specification and other guidelines are to be followed. The relation "agt" must be applied to "a thing that initiates an action", and "ptn" should stand for "an indispensable non-focused initiator of an action". The relation "agt" cannot be used in a different sense: it would be wrong. Flexibility in encoding should not be mistaken for permissiveness. There are many correct UNL expressions for the same NL sentence, but there are also wrong UNL expressions.

The solution to such a problem cannot be, however, to state a rigid (a culture-, language-, context- and even enconverter-independent) relationship between a NL and UNL, otherwise UNL will not suffice to cope with inevitable varying enconvertings. The fact that meaning is build through the enconverting process and its main consequence, the fact that different enconverters will propose different expressions for the same NL sentence, should be both considered starting points, instead of something that one can or should avoid.

The best solution is, thus, to trust the enconverter (and maybe to certify enconverters), and to be conscious that, as in any other translation activity, there are good and bad translations, and bad translations do not prove that translating is not possible or that it does not work. Only time and enconverters' expertise can make UNL expressions better.

Nevertheless, to trust enconverters may imply making deconverting extremely difficult and costly. The more UNL allows flexibility in enconverting, the more costly will be UNL-NL deconverting, since the UNL expression may contain unexpected relations.

This is, however, a false problem. Deconverters are not committed to generate back the source sentence enconverted into UNL. Instead, they should be supposed to generate a NL sentence corresponding to the UNL expression. The original source sentence is definitely lost as it has been enconverted into UNL; only one of its possible interpretations (the one carried out by the enconverter) is preserved. Deconverters should take then UNL expression as the new source sentence, instead of using it just as an intermediate expression.

Furthermore, deconverting seems to be easier than enconverting, since much of the eventual meaning gaps may be inferred from the context by a human being (which is supposed to be the final user), instead of a machine. There is a very fragile break-even-point, from which generation results become excessively degraded, but the extent to which this happens will depend on the architecture of the UNL System.

7 Conclusion

The main conclusion to be extracted from the previous section seems to be a paradox: in multilingual MT Systems, in order to be a pivot language, UNL should not be treated as an interlingua, but as a source and a target language, at the same level as any other NL. Flexibility in enconverting brings UNL to be just like any other NL, in the sense it would allow UNL for coping with NL vagueness and indeterminacy, without sacrificing, however, the explicitness and clarity of UNL expressions, which would continue to be univocal and machine-tractable.

Acknowledgments The authors acknowledge Mr. Tadao Takahashi, for his management of the Brazilian branch of the UNL Project, and to CNPq (Brazil) for the partial financial support. This work has also been partially supported by the UNU/IAS and the UNDL Foundation. The opinions expressed in the paper, however, do not necessarily represent the view of the UNL mentors.

References

Cole, R.A.; Mariani, J.; Uszkoreit, H.; Zaenen, A.; Zue, V. (Eds.) (1995). *Survey of the State of the Art in Human Language Technology*. NSF/CEC/CSLU. Oregon Graduate Institute. November.
(http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html)

Eco, U. (1994). *La recherche de la langue parfaite dans la culture européene*. Paris, France: Editions du Seuil.

Farwell, D. and Wilks, Y. (1993). ULTRA: A Multilingual Machine Translator. In S. Nirenburg (Ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.

Furuse, O. and Iida, H. (1992). Cooperation between transfer and analysis in example-based framework. In *Proceedings of the 14th International Conference on Computational Linguistics*. Nantes, France.

Hutchins, W. J. & Somers, H. L. (1992). *An Introduction to Machine Translation.* San Diego, CA: Academic Press.

Mitamura, T., Nyberg, E. and Carbonell, J. (1993). In S. Nirenburg (Ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.

Nirenburg, S., Carbonell, J, Tomita, M. and Goodman, K. (1992). *Machine Translation: A Knowledge-Based Approach*. San Mateo, CA: Morgan Kaufman.

Okumura, A., Muraki, K and Akamine, S. (1993). Multi-lingual Sentence Generation from PIVOT Interlingua. In S. Nirenburg (ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.

Tanaka, H. (1993). Multilingual Machine Translation Systems in the Future. In S. Nirenburg (ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.

Uchida, H., Zhu, M. and Della Senta, T. (1999). *Universal Networking Language: A gift for a millennium.* Tokyo, Japan: The United Nations University.

Issues in Generating Text from Interlingua Representations

Stephan Busemann

DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
busemann@dfki.de

Abstract. Multi-lingual generation starts from non-linguistic content representations for generating texts in different languages that are equivalent in meaning. In contrast, cross-lingual generation is based on a language-neutral content representation which is the result of a linguistic analysis process. Non-linguistic representations do not reflect the structure of the text. Quite differently, language-neutral representations express functor-argument relationships and other semantic properties found by the underlying analysis process. These differences imply diverse generation tasks. In this contribution, we relate multi-lingual to cross-lingual generation and discuss emergent problems for the definition of an interlingua.

1 Introduction

In this contribution, we relate multi-lingual to cross-lingual generation and discuss emerging problems for the definition of an interlingua. Multi-lingual generation starts from non-linguistic content representations for generating texts in different languages that are equivalent in meaning. The generation of weather forecasts or environmental reports are typical examples. In contrast, cross-lingual generation is based on a language-neutral content representation which is the result of a linguistic analysis process. Generation for machine translation is a most prominent example.

Non-linguistic representations do not specify linguistic semantics nor do they reflect the structure of the text to be generated. In contrast, language-neutral representations express functor-argument relationships and other semantic properties found by the underlying analysis process. These differences imply diverse generation tasks.

However, there are also commonalities. In both cases, generation is the mapping of some semantic representation onto linguistic strings. We may assume a single generation process that uses different separately defined language specific knowledge sources. In both cases, we may view the underlying representation as an interlingua, since it attempts to cross the language barrier by providing content descriptions independently of the target language.

An instance of each type of tasks has been implemented using the generation system TG/2 (Busemann, 1996), quickly overviewed in Section 2. The usage of the same framework allows us to relate the tasks to each other (Section 3) and to gain insights

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 38–50.

relevant to a coherent definition of interlinguas, generation tasks, and generation knowledge (Section 4.).

2 TG/2 in a Nutshell

TG/2 is a flexible production system that provides a generic interpreter to a set of user-defined condition-action rules representing the generation grammar. The generic task is to map an input structure onto a chain of terminal elements as prescribed by the rule set. The rules have a context-free categorial backbone used for standard top-down derivation, which is guided by the input representation. The rules specify conditions on input ("tests") determining their applicability and allow navigation within the input structure ("access functions").

The right-hand side of a rule can consist of any mixture of terminal elements (canned text) or other categories associated with an access function. The presence of canned text is useful if the input does not express explicitly everything that should be generated. With very detailed input, the terminal elements of the grammar will usually be words. Given a category C and some (piece of) input structure I, production rules are applied through the standard three step processing cycle:

1. Identify the applicable rules;
2. Select a rule on the basis of some (freely programmable) conflict resolution mechanism; and
3. Apply that rule.

A rule is applicable if its left-hand side category is C and its tests hold on I. A rule is applied by processing its righthand side elements from left to right. Canned text is output right away, and non-terminal elements induce a new cycle with the new category and the return value of the access function. Processing terminates when all right-hand side elements have been realized successfully. In the case of a failure, processing backtracks to step 2. If no more rules are applicable, a global failure occurs. For details see (Busemann, 1996).

3 Relating Two Distinct Generation Tasks

TG/2 has been used in a variety of NLG tasks. We look at multi-lingual report generation and cross-lingual summarization. We then locate the tasks on a scale ranging from shallow to in-depth generation, and discuss advantages and drawbacks of these locations.

3.1 Task 1: Generating air quality reports from measurement data

Reports about air quality in a German-French border region (Busemann and Horacek, 1998) are currently produced in six languages (a web demo is available at http://www.dfki.de/service/nlg-demo). The reports are based on real measurement data taken from a database and on the user's parameters determining the type of the report (time series, average or maximum value description, threshold passing description). A report consists of up to six statements most of which are verbalized by TG/2. The initial text organization stage retrieves the relevant data, decides about the content of the statements and defines their order. For each statement to be verbalized by TG/2 it produces a domain-oriented non-linguistic intermediate feature structure serving as input to TG/2 (cf. Figure 1 for an example). Input expressions for TG/2 may specify e.g. the pollutant, the actual measurements, and their date and location. Moreover, further information is specified according to the user's choice of parameters. It should be noted that some input is just carried forward from the original system input (in Figure 1, this is `LANGUAGE, TIME, POLLUTANT, SITE, THRESHOLD-TYPE`), whereas other information originates from the DB query and text organization stage (`COOP` and `EXCEEDS` in Figure 1).

The text organization stage is entirely content-oriented, and the intermediate feature structures do not exhibit linguistic properties. The 'language' feature causes the selection of the rule set for the language requested. The determination of linguistic structure for each input expression is achieved by the TG/2 grammar rules. Since implicit information is associated with some parts of input expressions, canned text is used to make it explicit at the surface. An example in Figure 1 is the added notion of "at the measuring station at" in the case of `(SITE "Saarbrücken- City")`, which is verbalized through the rule in Figure 2. The grammars comprise about 100-120 rules for each language and are specifically designed for this application. The development of a grammar for another language takes between one and three weeks depending on skills.

```
[    (COOP THRESHOLD-PASSING)
     (LANGUAGE ENGLISH)
     (TIME [(PRED SEASON)
     (NAME [(SEASON WINTER)
     (YEAR 2001)])])
     (POLLUTANT SO2)
     (SITE "Saarbruecken-City")
     (SOURCE [(THRESHOLD-TYPE MIK-WERT)])
     (EXCEEDS [(STATUS YES) (TIMES 1)])]
```

Fig. 1. A Non-Linguistic Input Expression for Report Generation: "In Winter 2001 at the measuring station at Saarbrücken-City, the MIK value for sulfur dioxide was exceeded once."

3.2 Task 2: Generating medical scientific text for summaries

This generation task occurred in the context of the cross-lingual text summarization system MUSI (Lenci et al., 2002). MUSI involves a combination of analysis and generation similar to machine translation. An interlingua approach was chosen to represent selected English and Italian medical scientific sentences in a language-neutral way.

```
(defproduction site "S01"
     (:PRECOND
            (:CAT SITE-E
             :TEST ((always-true)))
      :ACTIONS
            (:TEMPLATE
                "at the measuring station at "
                (:RULE SITE-NAME-E (self)))))
```

Fig. 2. Making Implicit Meaning Explicit: A TG/2 grammar rule. The rule is "unconditioned" and uses the current piece of input structure to access the site name.

The sentences can be complex and quite long (50 words are no exception). Interlingua expressions were fed to sentence generation components producing the elements of a French or German summary.

The generation of German sentences (Busemann, 2002) starts from so-called IRep4 interlingua expressions. A sample IRep4 expression is shown in Figure 3. IRep4 expressions are hierarchical predicate-argument structures complemented by a rich variety of features and modifiers. The basic elements are atomic and predicative concepts, forming an ontology shared across the MUSI system. In particular, predicative frames are based on the SIMPLE formal specifications (Lenci et al., 2000). IRep4 expressions are composed of `PROP` and `ITEM` elements used to represent propositions and terms, respectively. Although IRep4 is in principle a semantic representation language, its expressions also keep track of some syntactic properties of the source language elements. For instance, number and determiner information is specified for NPs as well as categorial information for propositions (`CAT`). This information can be very useful in guiding text generators. IRep4 is suitable for representing the semantics of very complex sentences, but at the same time, it leaves room for various degrees of specification. In fact, co-reference resolution, attachment ambiguities and the incorrect identification of arguments and modifiers are common sentence analysis problems that may lead to incomplete output. To cope with these problems, IRep4 has been designed to integrate possibly underspecified or fragmentary representations. This feature greatly enhances the robustness of the system and can guarantee a better interface with the text analysis component.

A direct interpretation of IRep4 by TG/2 would require choosing the lexemes and the syntactic realizations. This could have been achieved within the TG/2 grammar through complicated tests. These choices partly depend on each other, which would have caused massive backtracking. Moreover, testing the presence of a concept in IRep4 would have been triggered by rules expanding the syntactic category of the lexemes (part of speech), e.g. the rule `Noun  "acetylcholin"` would have been associated with a test whether the current concept was `C acetylcholine`. As there would have been hundreds of these, concerns of processing efficiency were in order. Finally, a pre-existing grammar should be reused that was not previously adapted to IRep4.

```
PROP{ Value = P_ARG1_cause_ARG2;
       Time_Rep = [PRESENT, PRES_USUAL];
       Cat = V_SEN;
       Arg1 = PROP{ Value = P_antagonism_with_ARG1;
                    Cat = NP; Det = INDEF;
                    Arg1 = ITEM{ Value = C_acetylcholine;
                                 Mod1 = [LOC, ITEM{
                                 Value = C_level;
                                 Det = DEF;
                                 Mod1 = [RESTR, ITEM{
                                 Value = C_sight;
                                 Number = PLUR; Det = DEF;
                                 Mod1 = [RESTR,
                                            C_muscarinic];
                                 Mod2 = [RESTR, ITEM{
                                               Value =
                                               C_substance;
                                         Number = PLUR;
                                         Det = DEMONST1;}];
                                      }]; }]; };
                  Mod1 = [RESTR, C_competitive]; };
     Arg2 = ITEM{ Value = C_effect;
                  Det = DEF; Number = PLUR; }; }
```

Fig. 3. IRep4 Expression for “Die Wirkungen werden durch einen kompetitiven Antagonismus zu Acetylcholin auf dem Niveau der muskarinischen Bindungsstellen dieser Substanzen verursacht.” [The effects are caused by a competitive antagonism with acetylcholine on the level of the muscarinic sights of these substances.].

For these reasons it appeared more convenient to introduce an initial sentence planning stage. The resulting representation – see Figure 4 for an example corresponding to Figure 3 – forms the input to TG/2. It can be viewed as a syntactically enriched, language-specific paraphrase of the underlying IRep4 expression. It represents explicitly the linguistic structure of the sentence. The TG/2 grammar is responsible for word order and inflection. Very much like in a classical sentence realization system, no canned text parts are used. If a phrase like “at the measuring station at” had to be generated here, an underlying interlingual semantic expression would be mandatory.

A pre-existing TG/2 grammar for German syntax was reused and adapted to the needs of MUSI (Busemann, 2002; Lenci et al., 2002). Its final version comprises over 950 rules.

3.3 Shallow and in-depth generation

The notion of shallow generation, as opposed to indepth generation, has been coined by (Busemann and Horacek, 1998) to describe a distinction corresponding to that of shallow and deep analysis. In language understanding deep analysis attempts to “un-

```
[(SENTENCE DECL)
 (VC [     (VOICE PASSIV)
           (MOOD IND)
           (TENSE PRAESENS)
           (SBP S2)
           (STEM "verursach")])
(DEEP-SUBJ [(TOP Y)
            (TY GENERIC-NP)
            (NUMBER SG)
            (DET INDEF)
            (NR V2)
            (GENDER MAS)
            (STEM "antagonismus")
            (PP-ATR [(LOCATIVE ...)
                      (GENDER NTR)
                      (STEM "Acetylcholin")
                      (DET WITHOUT)
                      (NUMBER SG)
                      (TY GENERIC-NP)
                      (PREP MIT)])
            (ADJ [(STEM "kompetitiv")
                   (POS ADJECTIVE)
                   (DEG POS)])])
           (DEEP-AKK-OBJ [(TY GENERIC-NP)
                          (NUMBER PLUR)
                          (DET DEF)
                          (STEM "wirkung")
                          (GENDER FEM)])]
```

Fig. 4. TG/2 Input Expression Partly Corresponding to Figure 3. The material for "on the level of the muscarinic sights of these substances" would appear under DEEP-SUBJ.PP-ATR.LOCATIVE, but has been omitted for reasons of space. The representation contains content word stems and names for syntactic structures (SBP, NR features). Determiners and prepositions are also provided.

derstand" every part of the input, while shallow analysis tries to identify only parts of interest for a particular application, omitting others. In-depth generation is inherently knowledge-based and theoretically motivated, whereas shallow generation quite opportunistically models only the parts of interest for the application in hand. Often such models will turn out to be extremely shallow and simple, but in other cases much more detail is required. Thus, techniques such as those developed within TG/2 for varying modeling granularity according to the requirements posed by the application are a prerequisite for reusing NLG systems.

Obviously a shallow NLG system is, in general, based on representations that carry implicit meaning. We call this shallow input. Additional text has to be "invented" by the generator (in TG/2, this is usually achieved using canned text in the grammar)[1].

[1] Of course, these texts are defined by the application, viz. the customer, as all other output..

This leads to domain-dependent, shallow grammars that cannot be reused easily for another task. The in-depth models assume a very fine-grained grammar describing all the linguistic distinctions covered by the interlingua. Such a grammar corresponds closely to familiar generic linguistic resources. The report generation task described was solved by a typical shallow approach, whereas the MUSI generation task required an in-depth model.

The tension between shallow and in-depth generation has been discussed further in the literature. According to Reiter and Mellish, shallow techniques (which they call "intermediate") are appropriate as long as corresponding indepth approaches are poorly understood, less efficient, or more costly to develop (Reiter and Mellish, 1993). Bateman and Henschel describe ways of compiling specialized grammars out of general resources (Bateman and Henschel, 1999). A platform for generating, storing and reusing representations is described in (Calder et al., 1999), showing that such reuse can be seen as a shallow methodology to text generation. A major conclusion seems that there is no dichotomy between both approaches, but that shallow systems can indeed be based on theoretically sound in-depth models.

In practice though, NLG tasks turn out to be highly diverse, and no NLG system could be reused for a new application off the shelf. The necessary effort for adaptation and extension of large existing in-depth resources such as KPML (Bateman, 1997) or FUF/Surge (Elhadad and Robin, 1996) is often considered high. In fact, the development from scratch of a shallow grammar for a small NLG application on the basis of a simple framework like TG/2 can be more cost-effective. Shallow and in-depth generation tasks can be related with help of TG/2. As the amount of domain-specific canned text in the TG/2 grammars correlates to the shallowness of the input, the generation tasks described can be located on a scale that ranges from shallow to in-depth domain and input models. There are trivial systems at one end that just produce canned text according to triggers (e.g. system error reports). A bit further on the scale we find template-style systems, like the air quality report generator, which use canned text to make knowledge implicit in the input explicit. In-depth realizers with sophisticated grammars that do not use domain-specific canned text at all are located at the other end of the scale, such as the MUSI generator. Why are shallow and in-depth interlinguas both viable? One obvious reason lies in the origin of the interlingua representations. Shallow representations usually originate from non-linguistic processing, such as accessing a database or interpreting some user interaction, whereas indepth representations generally have a linguistic origin, e.g. from an NL parsing component.

More interestingly, the type of domain and application determines the depth of modeling. Air quality reports form a small and closed domain. Implicit knowledge is easy to make explicit. A shallow model, being inherently simple, is perfectly adequate. A complex functor-argument representation would mean a dramatic overshot for this type of application. The same holds for many generation applications, such as reporting about stock exchange (Kukich, 1983) or weather forecasts (Boubeau et al., 1990). Medical scientific texts, on the other hand, form a very large domain, requiring broad-coverage linguistic knowledge. A shallow model would not even be able to capture the most frequent semantic relations. General means of expressing semantic relationships are mandatory.

What are the advantages and drawbacks of either approach? Shallow interlinguas allow for a straightforward multi-lingual generation. All linguistic processing can be concentrated in the module consuming the interlingua expression, e.g. TG/2. A drawback consists in domain dependent grammars, which are hardly reusable for other applications. Still it is worthwhile, as the effort to create a grammar for another language is low. With in-depth language-neutral representations, the issue of reusing existing linguistically motivated grammars arises, simply because of the tremendous effort for developing them from scratch. Technically an existing grammar may be reused if a well-defined interface is available. In TG/2, the interface to the input representations consists of the tests and access functions called from within the grammar rules. Depending on the different organization of information within input languages, this interface must be modified. If the same types of information required by the grammar can be produced by the new input language, the way is paved for a successful reuse. If the new input language offers different types of information, the adaptation problem described above arises.

4 On the Definition of Interlinguas

We now address issues on the semantics and pragmatics of interlinguas from a generation perspective by discussing three types of problems generators may encounter with in-depth interlinguas, using experiences with IRep4 as our source of examples[2].

4.1 Extrinsic problems

In MUSI, a variety of problems with interlinguas known from machine translation were experienced, showing that this interlingua, as so many others, is not language-neutral in a strict sense. The problems were related to the fact that languages encode information differently and the interlingua cannot sufficiently abstract away from this. More precisely, although IRep4 does not contain elements specific to any of the four languages involved, the analysis results reflected some grouping and nesting of phrases and clauses of the source language.

For instance, Italian (and English) uses post-nominal adjectival clauses that correspond to a post-nominal relative clause or pre-nominal adjectival modifiers in German (cf. Figure 5a). German does not have the possibility to linearize or nest several adjectival or participial clauses after the head noun. Moreover, large phrases in pre-nominal position are difficult to understand since the head noun is uttered only afterwards.

In IRep4, these clauses are typically represented as restrictive modifiers (`RESTR`), accompanied, in the case of a predicative concept, by the source-language specification `CAT = ADJP`. The generator follows the heuristic strategy of assigning small adjectival phrases to the pre-nominal adjective position and large ones to the post-nominal relative clause position. In the latter case, the `CAT` specification will be ignored, as a full sentence with a copula must be generated. A further requirement con-

[2] By critically reviewing IRep4, we necessarily omit mentioning many excellent features that made it very useful for the challenging task of representing scientific text

a) [[In the clinical case described,] [the symptoms] [were] [caused] [by ingestion [of anticolinergic substances
[probably contained [in the leaves [of plants [consumed a few hours before]]]]]]].

b) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch [Verzehr [von anticholinergen
Substanzen, [[die] [die Bl¨atter [der Pflanze], [die vor ein paar Stunden genossen wurden,] m¨oglicherweise enthielten,]]]]]
[verursacht]].
In the described clinical case were the symptoms by ingestion of anticolinergic substances, that-were in-the
leaves of-the plants, that-were a few hours before consumed, possibly contained.

c) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch Verzehr [von anticholinergen
Substanzen]] [verursacht], [[die] [die Bl¨atter [der Pflanze]] m¨oglicherweise enthielten, [die vor ein paar Stunden
genossen wurden]]].

d) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch Verzehr [von anticholinergen
Substanzen]] [verursacht], [[die] [die [vor ein paar Stunden genossenen] Blätter [der Pflanze]] m¨oglicherweise
enthielten]].

Fig. 5. Stylistic Variations in Translation. Brackets indicate some syntactic structure. a) English original sentence; b) Corresponding sentence in German with APs realized as relative clauses, with inter-linear translation; c) Extraposition of the relative clauses beyond the respective verbs; d) Realization of the innermost clause as a prenominal AP

sists of the need for one argument of the adjective to be realizable as the relative pronoun.

The result is not satisfactory, as it can lead to recursive center-embedding causing bad readability (cf. Figure 5b). The sentence in Figure 5c is stylistically much better; it has fewer closing brackets in a sequence, which means less deep embedding and improved readability. Linguistically, it shows two extrapositions, i.e. the innermost relative clause (not bracketed further) occupies the post-field[3] of the embedding one, which in turn occupies the post-field of the main clause. The stylistically preferred solution would be to realize the innermost clause as a prenominal AP, while extraposing the larger clause as a relative clause, as in Figure 5d.

Another striking example of language differences experienced with IRep4 is the use of determiners. English text does not use always definite articles when they are mandatory in German. For instance, “features of malnutrition” should be translated into “Merkmale der Mangelern ¨ahrung” (definite article included), whereas “features

[3] The post-field follows the infinite verb complex in a German declarative sentence. This position can be occupied by one constituent.

of chronic malnutrition” corresponds to “Merkmale chronischer Mangelern¨ahrung” (no article).

IRep4 does, of course, not represent definite articles when there are no such determiners in the source-language text. The generator uses as a general rule that “naked” generalized possessives – i.e. the head of a RESTRictive modifier that corresponds to a noun and does not have a determiner or a modifier – are automatically accompanied by a definite article, covering the above examples. English “Treatment consisted in...” should translate to “Die Behandlung bestand aus...”, using a definite article. In these cases, a decision within the generator on whether or not to use a definite article would rely on lexical semantic information about both the source and target language lexemes.

The obvious solution to the extrinsic problems is to complement the level of interlingua with a set of transfer rules specific for every pair of source and target language. This complicates the situation, but would, in MUSI, have led to considerable stylistic improvements of the generated sentences.

For shallow models, this problem simply does not exist.

4.2 Intrinsic problems

IRep4 also has a few intrinsic properties that affected generation. Most prominently, it does not represent scope and thematic, or constituent, order information. The scope of negation would be important for the proper placement of the negation particle. Moreover, the scope of modifiers is not represented. With the current, inherently flat representation, i.e. multiple modifiers at the same level of embedding, generation cannot decide between e.g. “the following clinical case” and “the clinical following case”. Modifiers should be nested to express this information.

Deciding about word order in generation is relevant to represent the argumentative structure in complex sentences and ensure coherence. The order of constituents in the source language text is not marked in IRep4, which may cause a deviating target-language order in German. This can lead to a lack of textual coherence, if e.g. a modifier that starts the sentence appears at the end. Consider “upon objective investigation, the woman‘s face was red and congested”, which was translated into “das Gesicht der Frau war rot und geschwollen bei objektiver Untersuchung”, generating the introductory PP at the end. A possible subsequent anaphoric reference would be less felicitous than in the original text. In the absence of a super-ordinated text planning stage, interlingua expressions should specify thematic order, or constituent order, in the source language text.

German generation assumes a standard word order for active voice, unless other information is given. The standard word order does not take into consideration the complexity, or the “weight”, of a constituent. A heavy-weight subject preceding a short object in a transitive sentence is often considered bad style. Based on heuristics about a constituent’s “weight”, passive voice could have been chosen within the generator, causing the short constituent to precede the complex one, which generally leads to more fluent text (cf. the example in Figure 3). An interlingua should include hooks to provide this information. IRep4 might indirectly allow a good estimate by counting concepts, arguments and modifiers; further investigation is needed to identify a reliable formula.

For shallow interlinguas, intrinsic problems of this kind do not exist, as they are entirely dealt with in the grammar.

4.3 Pragmatic problems

In this section, we sketch some issues that can take a lot of effort to create a shared understanding among the researchers looking at interlingua expressions from different perspectives.

A grammatically correct input sentence is a legitimate input to a parser. Few systems can deal with incorrect sentences in an error-tolerant way. For generation, in-depth interlingua expressions should be correct in a similar sense. A formal specification of the interlingua is required to define its syntax and, very importantly, its semantics. Generation requirements should be formally specified as well and should be part of the "pragmatics" of the interlingua. For instance,

- the omission of information about tense, aspect, determination and number may mean that a default applies;
- a personal pronoun must either refer to an antecedent, or be accompanied by information about gender, person and number;
- an expression realized as a relative clause must contain exactly one constituent with a plain coreference specification; this constituent will become the relative pronoun;
- etc.

During the development of IRep4, this effort was not spent due to shortage of resources[4]. While from an analysis viewpoint, some decent output looks more or less satisfactory, it is the details that make generation feasible or cause its failure. Most importantly, the interpretation of interlingua expressions in NLG should be functional. Different surface representations corresponding to the same interlingua expression should be considered as equivalent in meaning. If this fundamental principle is not maintained, translation is not guaranteed to be meaning-preserving. An interlingua can support this principle by making meaning representation explicit. IRep4 unfortunately has a fairly abstract representation for PP adjuncts and modifiers. The scheme is "`Mod = [<name>, <Irep4- expression>]`", where `<name>` is taken from a finite set of strings that more or less denote the semantics of the modifier. These names can be interpreted unambiguously by generation, but analysis may encounter difficulties in relating prepositions and head nouns to them, if only little lexical semantic knowledge is available. In Figure 3, the same name `RESTR` is realized differently, depending on the part of speech used for the embedded concept. If it is a noun, the semantics is that of a generalized possessive, which is realized in post-nominal position in German. If it is an adjective, a prenominal adjectival modifier is usually generated. Other uses of `RESTR` were mentioned above. If two or more

[4] It is debatable though whether the resulting difficulties have been resolved with less effort.

meanings are connected to one name, it may appear psychologically difficult to refrain from using this name as a waste-basket.

Pragmatic problems exist for shallow models as well, as shallow input expressions are partly produced by external systems. In the air quality report generator, measuring values are received as input from a database. Time series are occasionally shortened by aggregating information (“from 9.00 to 11.00: 6,7 g/m”). During the development, we have not been aware of the systematic omission of certain half hour values in the database, which occasionally leads to awkward results: “at 9.00: 6,7 g/m; at 9.30: 0 g/m; at 10.00: 6,7 g/m; at 10.30: 0 g/m; at 11.00: 6,7 g/m”. We easily could have implemented another aggregation rule that leads to output like “from 9.00 to 11.00: 6,7 g/m, with every half hour value at 0”.

5 Conclusion

In this contribution, we have related multi-lingual to cross-lingual generation and discussed emerging problems for the definition of an interlingua. This discussion was based on experience gained from implementing NLG components for a multi-lingual report generator and a crosslingual summarization system within the same framework, TG/2. Shallow interlinguas originate from non-linguistic processing. They usually carry implicit meaning that must be made explicit in the generation process. For relatively small-coverage, closed domains, such as air quality reports, weather reports, or stock market reports, it is adequate to write specialized grammars using domain-specific canned text for this purpose. In-depth interlinguas usually originate from linguistic analysis, as in machine translation. The nature of the interlingua is closely tied to the sophistication of the generation task in hand.

While well-modularized generation systems can be easily adapted to shallow interlinguas, an in-depth interlingua is much more complex to work with, as so many distinctions need to be addressed. In this paper we have identified some NLG requirements on in-depth interlinguas. From the experience with the MUSI application, we have learned that it is worthwhile to formally specify NLG requirements on the interlingua at the outset. For a new application involving multi-lingual or crosslingual generation, the interlingua should be chosen, adapted or designed according to the kind of linguistic processing involved and in view of the depth of modeling envisaged. On the shallow/in-depth scale, it should be as shallow as possible.

References

John Bateman and Renate Henschel. 1999. From full generation to ’near-templates’ without loosing generality. In (Becker and Busemann, 1999), pages 13–18. Also available at `http://www.dfki.de/service/NLG/KI99.html`.

John Bateman. 1997. KPML delvelopment environment: multilingual linguistic resource development and sentence generation. Report, German National Center for Information Technology (GMD), Institute for integrated publication and information systems (IPSI), Darmstadt, Germany, January. Release 1.1.

Tilman Becker and Stephan Busemann, editors. 1999. May I Speak Freely? Between Templates and Free Choice in Natural Language Generation. Workshop at the 23rd German Annual Conference for Artifi- cial Intelligence (KI ’99). Proceedings, Document D- 99-01. Also available at http://www.dfki.de/ service/NLG/KI99.html.

L. Boubeau, D. Carcagno, E. Goldberg, Richard Kittredge, and A. Polgu´ere. 1990. Bilingual generation of weather forecasts in an operations environment. In Proceedings of the 13 International Conference on Computational Linguistics (COLING-90), Volume 1, pages 90–92, Helsinki.

Stephan Busemann and Helmut Horacek. 1998. A flexible shallow approach to text generation. In Eduard Hovy, editor, Nineth International Natural Language Generation Workshop. Proceedings, pages 238– 247, Niagara-on-the-Lake, Canada. Also available at http://xxx.lanl.gov/abs/cs.CL/9812018.

Stephan Busemann. 1996. Best-first surface realization. In Donia Scott, editor, *Eighth International Natural Language Generation Workshop. Proceedings*, pages 101–110, Herstmonceux, Univ. of Brighton, England. Also available at the Computation and Language Archive at http://xxx.lanl.gov/abs/cmplg/9605010.

Stephan Busemann. 2002. Language generation for crosslingual document summarisation. In Huanye Sheng, editor, International Workshop on Innovative Language Technology and Chinese Information Processing (ILT&CIP-2001), April 6-7, 2001, Shanghai, China, Beijing, China, May. Science Press, Chinese Academy of Sciences.

Jo Calder, Roger Evans, Chris Mellish, and Mike Reape. 1999. ”free choice” and templates: how to geth both at the same time. In (Becker and Busemann, 1999), pages 19–24. Also available at http:// www.dfki.de/service/NLG/KI99.html.

Michael Elhadad and Jacques Robin. 1996. An overview of SURGE: a reusable comprehensive syntactic realization component. In Donia Scott, editor, Eighth International Natural Language Generation Workshop. Demonstrations and Posters, pages 1–4, Herstmonceux, Univ. of Brighton, England.

Karen Kukich. 1983. Design and implementation of a knowledge-based report generator. In Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics, pages 145–150, Cambridge, MA.

Alessandro Lenci, Nuria Bel, F. Busa, Nicoletta Calzolari, E. Gola, M. Monachini, Alexandre. Ogonowsky, I. Peters, W. Peters, N. Ruimy, M. Villegas, and Antonio Zampolli. 2000. SIMPLE: a general framework for the development of multilingual lexicons. International Journal of Lexicography, 13(4):249–263.

Alessandro Lenci, Ana Agua, Roberto Bartolini, Stephan Busemann, Nicoletta Calzolari, Emmanuel Cartier, Karine Chevreau, and Jos´e Coch. 2002. Multilingual summarization by integrating linguistic resources in the MLIS-MUSI project. In Procs. Third International Conference on Language Resources and Evaluation (LREC), Las Palmas, Canary Islands, Spain, May.

Ehud Reiter and Chris Mellish. 1993. Optimizing the costs and benefits of natural language generation. In Proc. 13th International Joint Conference on Artificial Intelligence, pages 1164–1169, Chambery, France.

On the Aboutness of UNL

Ronaldo Teixeira Martins,[1,2] Maria das Graças Volpe Nunes.[2,3]

[1]Faculdade de Filosofia, Letras e Educação – Universidade Presbiteriana Mackenzie
Rua da Consolação, 930 – 01302-907 – São Paulo – SP – Brazil

[2]Núcleo Interinstitucional de Lingüística Computacional (NILC)
Av. Trabalhador São-Carlense, 400 – 13560-970 – São Carlos – SP – Brazil

[3]Instituto de Ciências Matemáticas e da Computação – Universidade de São Paulo
Av. Trabalhador São-Carlense, 400 – 13560-970 – São Carlos – SP – Brazil

ronaldomartins@mackenzie.com.br, gracan@icmc.usp.br

Abstract. This paper addresses the current status, the structure and role of the UNL Knowledge Base (UNLKB) in the UNL System. It is claimed that the UNLKB, understood as the repository where Universal Words (UWs) are named and defined, demands a thorough revision, in order to accomplish the self-consistency requirement of the Universal Networking Language (UNL). In order to emulate human cognition and constitute the "aboutness" of the UNL, the UNLKB should be decentralized, distributed and reorganized as a network of networks, allowing for multicultural information and dynamic data.

1 Introduction

The Universal Networking Language (UNL) is an "electronic language for computers to express and exchange every kind of information" [Uchida, Zhu & Della Senta, 1999]. It can be defined as a knowledge-representation formalism expected to figure either as a pivot language in multilingual machine translation (MT) systems or as a representation scheme in information retrieval (IR) applications. It has been developed since 1996, first by the Institute of Advanced Studies of the United Nations University, in Tokyo, Japan, and more recently by the UNDL Foundation, in Geneva, Switzerland, along with a large community of researchers—the so-called UNL Society—representing more than 15 different languages all over the world.

Formally, the UNL is a semantic network believed to be logically precise, humanly readable and computationally tractable. In the UNL approach, information conveyed by natural language utterances is represented, sentence by sentence, as a hyper-graph composed of a set of directed binary labeled links (referred to as "relations") between nodes or hyper-nodes (the "Universal Words", or simply "UW"), which stand for concepts. UWs can also be annotated with attributes representing context-dependent information.

As a matter of example, the English sentence 'Peter kissed Mary?!' could be represented in UNL as (1) below:

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 51–63.

(1)
```
[S]
{unl}
agt(kiss(agt>person,obj>person).@entry.@past.@interrogative.@exclamative,
Peter(iof>person))
obj(kiss(agt>person,obj>person).@entry.@past.@interrogative.@exclamative,
Mary(iof>person))
{/unl}
[/S]
```

In (1), 'agt' (agent) and 'obj' (object) are relations; 'Peter(iof>person)', 'Mary(iof>person)' and 'kiss(agt>person,obj>person)' are UWs; and '@entry', '@past', '@interrogative' and '@exclamative' are attributes.

Differently from other semantic networks (such as conceptual graphs [Sowa, 1984, 2000] and the RDF [Lassila & Swick, 1999]), UNL relations and attributes are predefined in the formalism. As of the 3.2 version of the UNL Specification (UNL Center, July, 2003), the set of relations, which is supposed to be closed and fixed, consists of 44 elements that conveys information on ontological relations (such as hyponym and synonym), on logical relations (such as conjunction and condition), and on semantic case or thematic role (such as agent, object, instrument, etc.) between UWs. The set of attributes, which is subject to increase, currently consists of 72 elements, and cope with speaker's focus (topic, emphasis, etc.), attitudes (interrogative, imperative, polite, etc.) and points-of-view (need, will, expectation, etc.) towards the event. This feature brings UNL to represent not only denotative but also connotative, non-literal, information. The set of UWs, which is open, can be extended by the user, but any UW should be registered and defined in the UNL Knowledge-Base in order to be used in UNL documents.

Under the UNL Program, natural language analysis and understanding is referred to as a process of "enconverting" from natural language (NL) into UNL. This enconverting process, which has been carried out in a somewhat computer-aided human basis, is said to be not only a mere encoding (i.e., to rephrase the original sentence using different symbols), but truly a translation from the source sentence into a new target language - the UNL -, which is thought to be as autonomous and self-consistent as any NL, and whose graphs are expected to be language-independent and semantically self-governing. As it targets the information conveyed by the source text rather than its syntactic or even its semantic structure, the UNL is assumed to be different from other interlingua-based approaches, and to be more akin to the knowledge representation paradigm than to the machine translation techniques. As a matter of fact, and at least for the time being, UNL has been mainly used for multilingual document generation, through a process referred to as "deconverting", which consists in automatically providing NL outputs that can be said to be functionally (yet not formally) equivalent to the information conveyed by UNL graphs.

In this paper, we address the current structure of the UNL Knowledge Base and some of the problems we have been facing during the process of creation and definition of UWs inside EPT-WEB, an English-to-Portuguese UNL-based MT project. The paper is organized as follows: Section 2 brings some additional information on UWs and their internal structure; Section 3 analyzes the concept of Master Definition (MD) and the structure of the UNL KB; in Section 4, we explore some problems and shortcomings of the current version of the UNL KB; finally, in Section 5, we suggest some changes and enhancements in the UNL KB structure.

The authors should acknowledge that some of the opinions and definitions indicated below do not necessarily represent the official perspective on the UNL and may not coincide with those supported by the UNL Center.

2 Universal Words (UWs)

Universal Words, the words of UNL, are composed of a root (usually referred to, in UNL Specifications, as "headword") and a suffix ("the constraint list"). The latter comes between parentheses and is used mainly when the root is believed to be ambiguous. Examples of UWs are presented below:

(2a) 'Universal Word'
(2b) 'UW(equ>Universal Word)'
(2c) 'Peter(iof>person)'
(2d) 'apple(icl>fruit)'
(2e) 'kiss(agt>person,obj>person)'
(2f) 'explain(icl>express(agt>thing,gol>person,obj>thing))'
(2g) 'Manyoshu(icl>Japanese poem)'

Both root and suffix are normally made out of English words, but it is also possible to find "extra UWs" (2g above), when no simple (single) English word can be suited to convey the intended meaning, as in the case of culture-dependent concepts, such as special sorts of clothing, dish, furniture, etc. In those cases, foreign (transliterated, if necessary) words may be used to label the root of a UW.

The fact that the vocabulary of UNL is mainly derived from English may introduce an undesired natural language bias which can be said to be not only ethnocentric (in the sense all foreign concepts would be reduced to the ones carved up by the English language) but mainly counter-effective, as it would lead UNL to be a mere sort of controlled English. However, it is claimed that UWs, as labels, do not have meaning themselves. They would be just unique strings of characters that are used to refer to concepts. In this sense, the root (the headword), as well as the suffix (the constraint list), do not play any role other than disambiguating and ensuring uniqueness to the UW. The obvious resemblance between UWs and English words would be rather accidental, in order to cope with the commitment that UNL, as a semantic network, should be, to some extent, humanly readable. The use of English words would make UWs to be mnemonic and would facilitate the use of UNL by humans, but it would be completely useless and ineffective from the machine-tractability point-of-view. Yet it may seem to convey some meaning, the machine would consider 'apple(icl>fruit)' as meaningful as any arbitrarily assigned memory address.

In order to UNL to be really self-consistent and language-independent, the meaning of a UW, i.e., its value, should be entirely derived from a set of relations assigned in the UNL itself. The meaning of 'apple(icl>fruit)' should not come from a human comprehension or an external language that would never be replicable by the machine, but, instead, should be stated in a purely intensional (non-mental) dimension, a sort of electronic (possible) world, which would represent the sense and the reference for UNL words and expressions. This digital (and artificial) world, and not the human analogical one(s), would be the "aboutness" of UNL, as it would comprise the truth-condition requirements for UNL expressions to be "meaningful". Inside the UNL System, such synthetic world has been referred to as the UNL Knowledge-Base (or simply the UNLKB), a huge network where nodes (concepts) would be interconnected as to emulate the structure of human cognition.

As a matter of example, the meaning of 'apple(icl>fruit)' should be defined by a set of binary relations such as those indicated by (3) below:

(3a) icl(apple(icl>fruit),fruit(pof>plant))=1;
(3b) obj(eat(agt>thing,obj>thing), apple(icl>fruit))=1;
(3c) aoj(round(aoj>thing), apple(icl>fruit))=1;
(3d) pof(apple(icl>fruit), apple tree(icl>tree))=1;

This means that "apple(icl>fruit)" would be the concept that concomitantly a) assigns an 'icl' (a-kind-of) relation to the concept labeled by the UW 'fruit(pof>plant)'; b) receives an 'obj' (object) relation from the concept labeled by the UW 'eat(agt>thing,obj>thing)'; c) receives an 'aoj' (attribute of thing) relation from the concept labeled by the UW 'round(aoj>thing)`; and finally d) assigns a 'pof' (part-of) relation to the concept labeled by the UW 'apple tree(icl>tree)'. The value of 'apple(icl>fruit)' would be the sum (and nothing but the sum) of all relations in which it takes part in the UNLKB. This is a rather negative definition, given that it does not state positively the meaning of "apple(icl>fruit)", but only the relations that it may take. The set of UWs would be therefore a sort of sign system where the value of a given sign would solely derive from its position in the network. This is to say that, at least at the lexical level, UNL would consist of "un système où tout se tient" (Meillet, 1901; Saussure, 1916), following hence the structuralist approach that "every language is a system, all parts of which organically cohere and interact [... where] no component can be absent or even different, without transforming the whole" (Gabelentz, 1901). This would be a *sine qua non* condition for the autonomy and self-consistency of the UNL.

It should be stressed that this negative (relational) definition does not necessary coincide with the positive, contentful one, normally ascribed by a human. In the example above, for instance, nothing has been said about the relation that the UW 'apple(icl>fruit)' takes with other UWs such as 'red(icl>color)' or 'apple pie(icl>pie)'. This means that, in UNL, at least in the given situation, such features do not participate in the definition of 'apple(icl>fruit)', which would be therefore incomplete from a human point-of-view. But given that complete definitions are not to be easily achieved, because they can be self-contradictory (as apples can be red or green, for instance) and dynamic (different users, or even the same user at different times, may have different experiences or reactions towards apples), the UNLKB is not expected to define, in an exhaustively way, all the meaning intended by any concept.

Actually, in the UNL Program, there seems to be at least two different representational levels for defining UWs. The first would be related to the UNLKB itself and would target the (alleged) systematic part of the meaning, in a sense very close to the one intended by the concept of "semantic markers" (Katz & Fodor, 1963). On the other hand, the unsystematic part of meaning (the "distinguishers") would be treated in the UNL Encyclopaedia, which is a huge UNL document base, also organized as a network, where idiosyncrasies and additional information on UWs are expected to be stored. Here we will focus only on the UNLKB structure and on the boundaries between relations that should be necessarily included in a UW definition.

3 The UNL Knowledge-Base (UNLKB)

The UNLKB is a semantic network whose entries have the structure exemplified in (3) above. They comprise a binary directed relation (extracted from the UNL relation-set) between two UWs, along with a degree of certainty, which can range from 0 (completely false) to 255 (completely true). Any UNL-relation can hold between UWs in the UNLKB, and a single UW may receive and assign many different relations from and to other UWs. However, in order to assure replacebility, inference and cross-reference (inheritance) inside the network, any UW should be linked by at least one of the three ontological relations, namely "icl" (a-kind-of), "iof" (an-instance-of) or "equ" (equal-to). The relation "pof" (part-of), formerly used, has no longer been adopted, as it does not allow for direct inheritance.

One can say that linking a UW to any other by means of "icl", "iof" or "equ" is to compose a UW Thesaurus, or the UNL Ontology, but it should be stressed that such network is only a part of the UNLKB. Inside the UNL System, this subnetwork has been referred to as the "UW System", and it constitutes a lattice structure, given that a child-node may have many different parent-nodes. This hierarchical network also comprises an inheritance mechanism, so that all information assigned to a given parent-node could be directly inherited by its children-nodes. In this sense, if (4) below had been stated in the UNLKB, there would be no need for (3b), provided that it could be easily inferred from (3a):

(4) obj(eat(agt>thing,obj>thing), fruit(pof>plant))=1;
(3a) icl(apple(icl>fruit),fruit(pof>plant))=1;
(3b) obj(eat(agt>thing,obj>thing), apple(icl>fruit))=1;

The need for the UNLKB has been subject to criticism inside the UNL Project, but it should be observed that knowledge-based MT systems have proved to provide better results than those that are only language-based (Nirenburg, Raskin et al., 1986). Inside the UNL System, the UNLKB is intended to assure robustness and precision both to the NL-UNL enconverting and to the UNL-NL deconverting. In the former case, the UNLKB would be used as a sort of word sense disambiguation device; in the latter, the UNLKB, through the replacebility operations, would allow for the deconversion of UWs not predicted by the target language dictionaries. Additionally, the power of the UNLKB for intelligent searching and semantic inference and reasoning should not be underestimated.

In order to discipline and organize the creation of UWs, the UNL Center has proposed a particular technique for both naming (labeling) and defining a UW at a single movement: this is the Master Definition (MD), introduced in 2000. The MD for naming the UW "apple(icl>fruit)" and defining it in the UNLKB (through an "icl" relation to the UW "fruit(pof>plant)") is presented in (5) below:

(5) apple(icl>fruit{pof>plant})

The MD is said to facilitate (and regulate) the labeling of a UW, which would derive its suffix (the constraint list) from its definition in the UNLKB. The name of the UW would be the same as the MD without the strings included inside the curly braces. This would motivate the UW name and keep it as mnemonic as possible for human use (although, as said before, the name of a UW has nothing to do with its "mechanical" meaning). However, it should be noticed that the concept of MD brings itself many shortcomings, mainly the facts that: 1) due to the simplification of syntax, the MD is not capable of conveying any degree of certainty other than 1; and 2) MDs can only be used to define the UW by means of 'icl', 'equ' or 'iof'; any richer definition would require longer strings and more expensive strategies that may fall out the MD scope. Nevertheless, and at least for the time being, the UNLKB has been entirely defined as a hierarchy of MDs.

4 Problems and Limitations of the Current Version of the UNL KB

The current version of the UNLKB has been developed and constantly updated by a single person, the director of the UNL Center, and can be downloaded from the UNDL web site [http://www.undl.org]. The comments below refer to the version as of November 28, 2004. Among the problems, three have been specially selected and are going to be addressed in this paper, all of them related to our experience in creating UWs in an ongoing UNL-based MT Project carried out by NILC. The problems are the syntactic bias of the UNLKB; the synonymy of UWs; and the lack of criteria for categorization.

The syntactic bias of the UNLKB can be mainly ascribed to the third and the fourth uppermost levels of the UW System. According to the current version of the UNLKB, the topmost level of the UW System is the UW 'Universal Word' itself, followed by 'UW(equ>Universal Word)' and, next, by four different UWs, each of which has their own children, leading to the basic structure depicted in Figure 1 below.

From Figure 1, it is possible to see that the UNLKB structures the lexical semantics of UNL in a rather syntactically-biased way. Every UW should be linked – either directly or indirectly – to the four grammatical nodes: "nominal concept", "verbal concept", "adjective concept" and "adverbial concept", taken as semantic primitives. Even though these concepts may lead indeed to semantic notions, they normally refer to syntactic and morphological markers, as indicated in the case of the English word "reading" in the examples below, adapted from the WordNet [http://www.cogsci.princeton.edu/]:

(6a) She is **reading** a book.
(6b) She enjoys **reading** books.
(6c) She disapproved his **reading** of Shakespeare.
(6d) She bought some **reading** material.

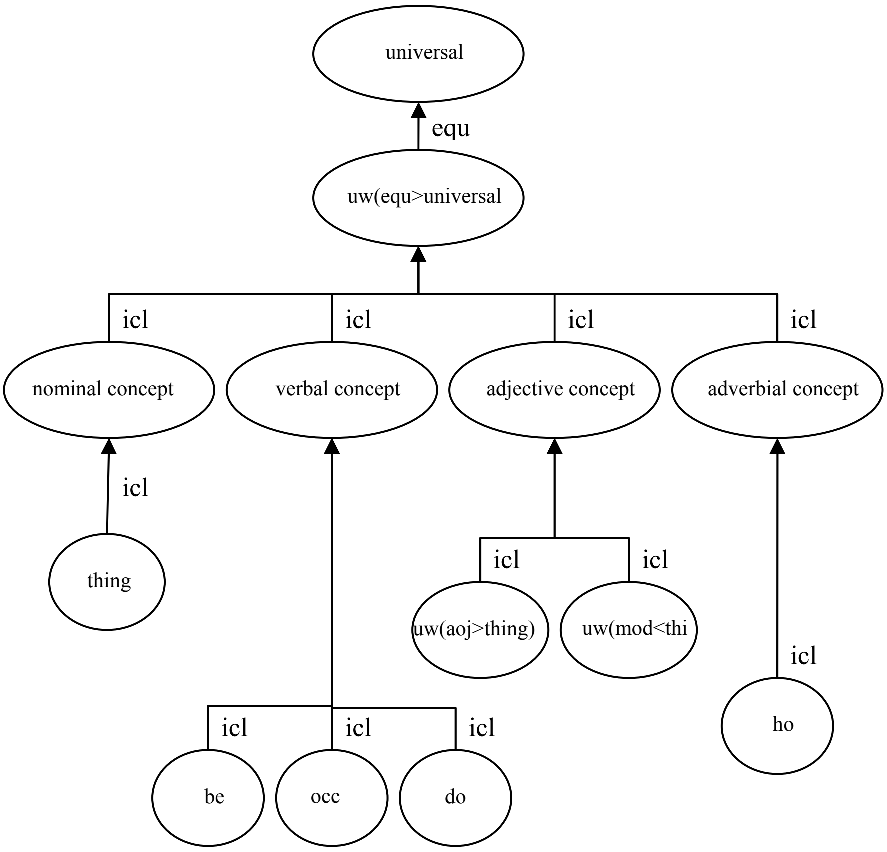


Figure 1: the topmost levels of the UW System

In the UNL representation for those sentences, it is likely to find the following:

(6a) read(agt>person,obj>information)
(6b) reading(icl>action)
(6c) reading(icl>information)
(6d) reading(aoj>thing)

Although UNL is fine-grained enough and capable of disambiguating between different uses of “reading”, it should be primarily attached to the semantic content conveyed by each occurrence. It should be observed, however, that those semantic values may have very little to do with the part-of-speech information, which is actually only a feature of English, rather than a universal character. In Portuguese, for instance, “reading” in (6b) would be normally translated by a verb, instead of a noun, as indicated in (6b’) below:

(6b’) Ela gosta de **ler** livros.

A similar problem occurs in (6c) and (6d), which would be indirectly linked to “nominal concept” and “adjective concept”, respectively, because in English they would play the roles of a noun and an adjective. In Portuguese, however, “reading” would be normally translated as a noun in (6c) and as prepositional phrase in (6d), according to (6c’) and (6d’) below:

(6c’) Ela desaprovou sua **leitura** de Shakespeare.
(6d’) Ele comprou algum material **para ler** (**de leitura**).

Both cases illustrate that, at least from the perspective of Portuguese, the definition of “reading” in (6a) to (6d) would be different from the one achieved if English is to be taken as the source language. This proves a language-dependent feature that may interfere in the definition of UWs, assigning a semantic value to operations that are rather syntactic (such as nominalization, deverbalization, adjectivation, etc). If UNL is really expected to represent the ideational content of utterances, rather than its syntactic or semantic structure, it should consider such sort of cross-language mismatches, in order to be as language-independent as possible.

Two additional illustrations of the syntactic bias of the UNLKB can be reached if we consider the representation of UWs conveying information on places and adjectives. In both cases, we have to consider the realm of ‘adjective concepts’ and ‘adverbial concepts’, which, differently from ‘nominal concepts’ and ‘verbal concepts’, do not correspond to a real taxonomy, but to a flat list where there is no internal hierarchy among the elements.

As to places, for instance, the UNLKB comprises two different UWs corresponding to the English word “here”: a) simply ‘here’, without any suffix, under ‘adverbial concept’; and b) ‘here(icl>place)’, under ‘nominal concept’. It is possible to say that they cover the meanings intended respectively by (7a) and (7b) below (extracted once again from the WordNet):

(7a) in or at this place; where the speaker or writer is; "I work here"
(7b) the present location; this place; "where do we go from here?"

However, it should be stressed that, in both cases, ‘here’ is essentially a place, regardless of its grammatical role (i.e., its part-of-speech) in the sentence. Although this information may be kept in ‘here(icl>place)’, it is definitely lost in the basic UW ‘here’, which is directly located under ‘how’ (and hence under ‘adverbial concept’). This is especially unproductive because the “adverbial role” of “here”, if any, could be alternatively represented by means of the relation “plc” (place) or even “man” (manner).

A third illustration may come from the classification of adjectives. In UNL, adjectives - which also correspond to a plain list instead of a hierarchy - are said to be either predicative, or attributive, or both. This is the case for “good”, that can be found either as ‘good(mod<thing)’ or ‘good(aoj>thing)’, in order to cope with (8a) and (8b), respectively:

(8a) A good boy
(8b) The boy is good.

It is under dispute, however, if the opposition between predicative and attributive, that maybe is relevant for English, really holds in every language. Is this a real general semantic phenomenon or simply a language-dependent syntactic event? If semantic, should not it be represented by relations or attributes instead of UWs? Is it really useful to register, in the UW dictionary, both 'good(aoj>thing)' and 'good(mod<thing)', given that they mean the same?

As a matter of fact, the repertoire of UWs seems to indicate that the lexicalization of UWs is exaggeratedly based on the surface structure of English sentences and on the lexical items of English. This can be further attested because of the presence of variants, antonyms and synonyms in the UW dictionary.

For instance, one will find, in the UNL KB current version, both 'behavior(icl>action)' and 'behaviour(icl>action)'. The difference between them is not semantic, but strictly orthographic. There is no reason for cataloging such kind of spelling difference in a semantic database.

The same should apply for pairs of antonyms such as give/receive, borrow/lend, etc. These verbs are supposed to convey the same meaning in a reversed subcategorization frame: give(x,y) = receive(y,x). Once "give" and "borrow" are there, would there be any reason for including "receive" and "lend" as well?

(9a) give(agt>thing,gol>person,obj>thing)
(9b) receive(agt>thing,obj>thing,src>thing)
(10a) borrow(agt>thing,obj>thing)
(10b) lend(agt>thing,gol>person,obj>thing)

This sort of overlapping among UWs does not affect only antonyms and can be found all over the UNLKB. Let us consider two last examples: is there any real need for registering, in the same knowledge base, all the words appearing in (11) and (12) below? Are the semantic differences between them really relevant? Are they going to be preserved in languages other than English?

(11a) begin(agt>thing,obj>thing)
(11b) commence(icl>begin(agt>thing,obj>thing))
(11c) start(icl>begin(agt>thing,obj>thing))
(12a) nurse(icl>medical assistant)
(12b) nurse({icl>person>human,}icl>occupation{>work})

The examples referred to above prove that economy has not been an asset of the UNLKB. Obviously, one can claim that synonyms and variants are to be represented, because there is no perfect synonymy and UNL is supposed to be as comprehensive and fine-grained as any natural language. But again, and provided that there is no perfect lexical matching between languages, would UNL be wide enough to comprehend the vocabulary of every existing natural language? How to prevent combinatorial explosion inside the UNLKB? How to prevent that the proliferation of UWs will not affect the maintenance of UNL resources, will not introduce different dialects to UNL, and will not cause the entropy of the whole system?

In addition to the syntactic bias and the synonymy, there are many other problems that could be pointed out inside the UNLKB, but most of them are far much easier to handle. Due to the discrepancy on the use of braces, for instance, there are many duplicated entries inside the network, such as (13) below:

(13a) eau de cologne{(icl>perfume>functional thing)}
(13b) eau de cologne(icl>perfume{>functional thing})

And there are also some class inconsistencies. Tigers and panthers, for instance, are normally defined as belonging to the species of felines, but, in the UNLKB, they have been categorized directly under 'mammal(icl>animal)', differently from 'cat(icl>feline)':

(14a) tiger(icl>mammal{>animal})
(14b) panther(icl>mammal{>animal})
(14c) cat(icl>feline{>mammal})

In the same way, specific languages and types of languages have been categorized at the same level, as indicated in (15) below:

(15a) spoken language{(icl>language>system)}
(15b) Russian(icl>language{>system})
(15c) inflectional language{(icl>language>system)}
Circularity may also be found, as in (16) and (17):
(16) thing{(icl>nominal concept)}
abstract thing{(icl>thing)}
event(icl>abstract thing{>thing})
thing(icl>event{>abstract thing})
(17) figure(icl>figure{>attribute})

The main problem, however, concerns the lack of (uniform) criteria for categorizing concepts. In (18) below, for instance, the concept conveyed by the English words "film" and "movie" is said to be linked to the concept of "abstract thing". Why that? Why not "concrete thing"? Or why not "functional thing" instead? What about instances of films, such as "Gone with the wind"? Would they also be considered a sort of "abstract thing"?

(18) abstract thing{(icl>thing)}
art(icl>abstract thing)
cinema(icl>art{>abstract thing})
film(icl>cinema{>art})
movie(icl>cinema{>art})

Such categorization turns to be even more astonishing if we consider the case for "book", which is also located under the "abstract thing" branch of the UNLKB, as indicated in (19):

(19) abstract thing{(icl>thing)}
information{(icl>abstract thing)}
document(icl>information)
book(icl>document{>information})
book of general works{(icl>book>document)}
manuscript{(icl>book of general works)}
rare book{(icl>book of general works)}
book of geography{(icl>book>document)}

On the other hand, both "landscape" and "scenery", and even "beauty spot", are categorized under "concrete thing", as seen in (20):

```
(20) concrete thing{(icl>thing,icl>place>thing)}
    natural world{(icl>concrete thing,icl>place>thing)}
      landscape(icl>natural world)
          scenery(icl>landscape{>natural world})
                    beauty spot(icl>scenery{>landscape})
                    scene(icl>scenery{>landscape})
```

The absence of categorization guidelines and protocols cause the UNLKB to be excessively impressionist, in the sense it contains, to a considerable extent, subjective and personal ideas towards the world and the structure of events. Although some of those decisions may sound quite reasonable from a given perspective, it is clear that they cannot be taken for granted. They are rather culture- and even individual-dependent and will be subject to an everlasting dispute. As a matter of fact, this is said to be the main reason why knowledge-based approaches have been discarded as a feasible strategy for language processing and, inside the UNL Program, this is probably the reason why there is so much resistance on adopting a more fine-grained level of lexical description.

In fact, outside the UNL Center, it has been observed a relatively flat use of the suffixes of UWs, as if their only role was to assign some part-of-speech information to the roots and to disambiguate between nouns, verbs, adjectives and adverbs. Even though UWs as simple as 'book(icl>thing)' or 'book(icl>do)' can be really uncomplicated and effortless, they are not trouble-free, as they may not totally disambiguate English words and assure precision and robustness to both enconverting and deconverting. In the WordNet, for instance, the English word 'book' (presented in Table 1, below), as a noun, may take ten different senses, some of which may not be not translated, in Portuguese, by the same single word. In those circumstances, a low-level use of suffixes would not only be insufficient, but mostly misleading. To reduce all senses of "book" to 'book(icl>thing)' would be no better than declaring that "book" is a sort of "abstract thing".

Consequently, the best solution for the limitations pointed out above ought not to be to extinguish the UNLKB, and cause the UNL System to be a strictly language-based representation formalism (which would turn UNL into a mere metalingua), or to deprive the UNLKB, restricting its power and the granularity of its representation. Actually, the answer is to keep improving the UNLKB, but in a rather different perspective, as suggested in the next section.

5 On the ideal structure of the UNLKB

The UNLKB urges to be decentralized. The development of the UNLKB cannot be a single-man activity, regardless of how good this man can be. In order to avoid one-sided decisions on categorization and in order to push for classification standards and protocols, the UNLKB has to be conceived as a really multilateral and multicultural endeavor. In this sense, the UNLKB has to be plural rather than singular. This is to say that there should be allowed many different UNLKBs, or that the UNLKB should

Table 1. English-to-Portuguese correspondence for the noun “book”

English	Definition	Portuguese
1. **book**	a written work or composition that has been published printed on pages bound together; "I am reading a good book on economics"	livro
2. **book**, volume	physical objects consisting of a number of pages bound together; "he used a large book as a doorstop"	brochura
3. ledger, leger, account book, book of account, **book**	a record in which commercial accounts are recorded; "they got a subpoena to examine our books"	registro
4. **book**	a number of sheets ticket or stamps etc. bound together on one edge; "he bought a book of stamps"	álbum
5. record, record book, **book**	a compilation of the known facts regarding something or someone; "Al Smith used to say, `Let's look at the record'"; "his name is in all the recordbooks"	registro
6. **book**	a major division of a long written composition; "the book of Isaiah"	livro
7. script, **book**, playscript	a written version of a play or other dramatic composition; used in preparing for a performance	livro
8. **book**, rule book	a collection of rules or prescribed standards on the basis of which decisions are made; "they run things by the book around here"	livro
9. Koran, Quran, al-Qur'an, Book	the sacred writings of Islam revealed by God to the prophet Muhammad during his life at Mecca and Medina	Livro
10. Bible, Christian Bible, Book, Good Book, Holy Scripture, Holy Writ, Scripture, Word of God, Word	the sacred writings of the Christian religions; "he went to carry the Word to the heathen"	Livro

behave as a distributed network, where many different (even contradictory) repositories of information are somehow interrelated, according to some specific topology that should be addressed by the UNL Center. Ultimately, the UNLKB has to be turned into a network of networks, or the UNL World Wide Web itself.

The fact is that the UNL Center’s KB can no longer be considered the sole parameter for verifying the correctness and adequacy of UNL expressions, but should be taken as a reference to be pursued by any Language Center involved in the UNL Program. The UNL Center’s KB should play the role of a Core KB, which, at least for time being, would be responsible for regulating and indexing any other satellite KBs to be available in the UNL Web. However, in other to be taken as such a reference, the UNL Center`s KB should undergo some general categorization protocols that are to be discussed and obeyed. These protocols must take into consideration the fact that the UNLKB is a semantic network where UWs are interconnected to emulate human perception and categorization, and that human cognition may vary a lot, between different cultures and even among different subjects. Such considerations ought to govern the whole process in order to avoid excessively naive approaches on ontology and should benefit from the extensive use and study of knowledge representation strategies that have been carried out inside the Artificial Intelligence.

Whatever the case may be, it should be stressed that the answers to the questions presented in the last section are not exactly as simple as they may seem. Actually, they involve the whole philosophy behind UNL and what UNL is supposed to represent. Banishing synonyms, antonyms, variants and other alleged excesses from the UNLKB may obviously impoverish and weaken the representation power of the UNL and will bring consequences that should be considered in the UNL environment. From the human speaker`s perspective, “John gave a book to Mary” and “Mary received a book from John”, although quite related, may convey different meanings. The same would hold for the difference between “behavior” and “behaviour”, which may be used to attest a dialect. In both cases, however, UNL would be far much closer to the semantic and syntactic surface structure of natural language sentences than it would be advisable. Maybe UNL should focus, at least in its very beginning, on a sort of deeper information structure that could be more easily extracted from natural language utterances, so that it would be possible to represent a part, yet infinitesimal, of its alleged meaning. This would be not only more straightforward and faster, but it would also allow for extending the knowledge on natural language syntax and semantics so to provide better results somewhere in the future.

References

Katz, J. and Fodor, J. The structure of a semantic theory, Language, 39, pp. 170-210, 1963.

Lassila, O. and Swick, R. R. Resource Description Framework (RDF): model and syntax specification. W3C Recommendation, 1999.

Nirenburg, S, Raskin, V et al. “On knowledge-based machine translation', In Proceedings of the 11th International Conference on Computational Linguistics, Bonn, 1986,

Sowa, J. F., Conceptual Structures: Information Processing in Mind and Machine, Addison-Wesley, Reading, MA, 1984.

Sowa, J. F., Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

Uchida, H. and Zhu, M. UNL annotation. Version 1.0. UNL Centre/UNDL Foundation, Geneva, 2003.

Uchida, H., Zhu, M. and Della Senta, T. A gift for a millennium, IAS/UNU, Tokyo, 1999.

UNL Centre. Enconverter specifications. Version 3.3. UNL Centre/UNDL Foundation, Geneva, 2002.

NL Centre. UNL Specification. Version 3.2. UNL Centre/UNDL Foundation, Geneva, 2003

A Comparative Evaluation of UNL Participant Relations using a Five-Language Parallel Corpus

Brian Murphy and Carl Vogel

Brian.Murphy@cs.tcd.ie*, Vogel@cs.tcd.ie
Department of Computer Science, University of Dublin, Trinity College

Abstract. In this paper we describe a manual case study in interlingual translation among five languages. Taking the UN Declaration of Human Rights in Chinese, English, German, Irish and Spanish, we annotated the five texts with a common interlingual logical form. We then studied four inventories of semantic roles (developed for both theoretical andNLP applications), including a subset of UNL's relations, and evaluated their suitability to describe the predicate-argument relationships found in the annotation. As a result, we make some suggestions for possible additions to the UNL relations, and propose that some of the existing relations be conflated or redefined.

1 Introduction

The work described here is part of a feasibility study on the use of semantic roles in interlingua-based machine translation. Our objective was to see if any set of semantic roles could give a description of verb-predicate relationships across a range of languages that would form an adequate basis for automatic generation.

The languages chosen were those that the authors have some working knowledge of (English, Chinese, German, Irish and Spanish), and include widespread and minority languages, both well and less-studied. The corpus used is theUN Declaration of Human Rights [1], a short text covering a broad range of topics in many languages (see Sect. 2).

From the literature on roles we selected four inventories (of whichUNL's relations is one) that we considered to be well-enough developed for the annotation of unrestricted text. These inventories ([2,3,4,5] detailed in Sect. 4) were also chosen to be representative both theoretically and in terms of application to tasks such as machine translation and information retrieval.

After aligning the five language versions of the corpus, we manually annotated each article of the text with a language-neutral logical form (effectively a prototype interlingua) following the guidelines described in Sect. 3.1. The main part of the work then involved applying each of the role inventories in turn to the logical form and determining whether they satisfied three key criteria: coverage, differentiation and lack of ambiguity (Sect. 5). In other words, one should

* Supported by the TCD Senior Lecturer's Broad Curriculum Fellowship and Enterprise Ireland

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 64–76.

be able to annotate every predicate argument with a role, that role should be unique with respect to its predicate, and the assignment of that role should be unequivocal. During this process we also gathered some impressions on the relative strengths and weaknesses of each of the inventories studied.

We had some problems interpreting UNL's documentation on some relations (based on the publicly available specifications and manual [5,6]), and we make suggestions on where this can be improved. In particular we suggest some redefinition of the causal/affected relations AGT, OBJ and AOJ (Sect. 5.1) and propose a more radical rationalisation of the locational relations PLC, PLF/PLT, SRC/GOL and FRM/TO (Sect. 5.2). Finally we consider whether further dedicated relations should be added for arguments that do not contribute as much to causality or directionality, such as possessors/possessions and the peripheral participants of beneficiary and recipient (Sect. 5.3).

2 The Corpus

For our research we were interested in a source of parallel texts that, besides three major Western European languages (German, English and Spanish), included both a minority language (Irish Gaelic) and a major non-European language (Standard Chinese). This rules out most collections from international organisations like the UN or the EU. However the UN Declaration of Human Rights [1], though short (approx. 1500 words) is freely available from the web, and professionally translated to more than 300 languages. While the register is restricted, it covers a wide range of topics, including education, politics, religion, law, the family, asylum, ownership, employment, leisure, culture and health. It offers complex sentence structures (such as deeply nested clauses) and widespread inter-sentence relationships, such as anaphora and mutual conditions between propositions (for example the dependencies between predicates in (2f)), but is simple to align to a sentence level, due to its organisation into articles and sub-articles.

The five languages included cover several branches of the Indo-European family of languages (Celtic, Romance and Germanic) together with a Sino-Tibetan language, and are varied in terms of argument structure. Compared to the fixed *subject-verb-object* structure found in English, German differs in using case and in allowing object fronting, while Spanish allows both object fronting and subject omission (pro-drop). Irish has a *verb–subject–object* word order, while Chinese argument realisation is very flexible, in principle allowing any argument to be moved or dropped. German clause structure differs from the others in grouping non-finite verbs at the end of a clause (e.g. 'gemacht werden' in (1a)), while subordinate clause ordering in Chinese is radically different with modifiers generally preceding heads ('type' and 'right' in (1b)).[1] The copula ('be') has multiple realisations in both Spanish ('ser'/'estar') and Irish ('bí'/'is'), and Irish also has widespread use of prepositional and adverbial forms for representing events – e.g. in (1c) an abstract possession is expressed as being 'at' the owner.[2]

[1] DE is a modifier particle.

[2] Examples from the Declaration indicate the source article.

(1) a. Berufsschulunterricht müssen allgemein verfügbar gemacht werden …
vocation-lesson must general available made to-be …
'Professional education shall be made generally available …' [Art. 26.1]

b. 父母对其子女所应受的教育的种类,有优先选择的权利
fùmŭ duì qí zĭnŭ suŏ yīng shòu de jiàoyù de zhŏnglèi, yŏu yōuxiān xuănzé de quánlì
parent to its children that should receive DE education DE type, has priority select DE right
'Parents have a prior right to choose the kind of education that shall be given to their children.' [Art. 26.3]

c. Is ionann na cearta atá acu …
is same the rights that-are at-them …
'They are entitled to equal rights' [Art. 16]

3 Interlingual Annotation

We manually aligned all 49 articles and sub-articles of theUN Declaration across the five languages, before adding English glosses (i.e. word-for-word translations, as seen in the previous examples) for all the non-English texts. The logical annotation of each article then proceeded on the basis of the English original and the four glosses, yielding over 500 predicates with almost 900 arguments. The aim was to arrive at a single, cross-linguistic logical form that, to the extent possible, adequately represented an article's meaning as expressed in all five versions. Although the result does not follow any of the five surface forms exactly, we aimed to abstract away from them only to the extent necessary to find a common representation.

To our knowledge there are no generally accepted guidelines for the manual annotation of unrestricted text with logical forms, as they are often theory or application specific. However, two sources proved useful. The Penn Propbank (a semantically annotated corpus) guidelines [7] have useful suggestions that we adopted for the treatment of phrasal verbs, support verbs and nominalizations. From cognitive science, Kintsch [8] gives an brief overview of annotation conventions for the 'microstructure' (roughly intra-sentence structure) of propositions, as used in comprehension modelling. We have broadly followed his treatment of negatives, modals, adjectives, adverbs and the status of propositions as arguments themselves.

3.1 Guidelines Developed

Negatives, modal verbs, adjectives and adverbs are expressed as one-place predicates with an event or object argument. As the focus of our studies is valency patterns, the quantification of objects was not annotated and noun phrases are rarely decomposed. Thus "all the rights and freedoms [Art. 2]" would be rendered as the atomic object *AllTheRightsAndFreedoms* as opposed to a form like $[\forall x.[\mathrm{right}(x) \vee \mathrm{freedom}(x)]]$. Tense and aspect are not encoded.

Passive sentences are expressed actively with an undefined logical subject (annotated *U*). Complex sentences are decomposed into component predicates, and nominalizations are given predicate translations where possible (e.g. “interference in privacy” becomes *interfere(U,Privacy)*). Repeated objects and events are given numbered *O* and *E* variables to indicate identity:[3]

(2) a. Everyone charged with a penal offence has the right to be presumed innocent until proved guilty . . . [Art. 11.1]
b. 凡受刑事控告者,在. . . 证实有罪以前,有权被视为无罪. . .
fán shòu xíngshì kònggào zhě, zài . . . zhèngshí yǒu zuì yǐqián, yǒu quán bèi shìwéi wúzuì
every receive criminal charge person, at . . . confirm has guilt before, has right BEI regard innocent
c. Jeder, der wegen einer strafbaren Handlung beschuldigt wird, hat das Recht, als unschuldig zu gelten, solange seine Schuld nicht . . . nachgewiesen ist . . .
everyone, who because-of a criminal act charged be, has the right, as innocent to count, while his/her guilt not proved is
d. Gach duine a cúiseofar i gcion inphíonois is tuigthe é a bheith neamhchiontach go dtí go gcruthaítear ciontach é . . .
every person that charged in offence punishable be understood him that be innocent until that prove guilty him
e. Toda persona acusada de delito tiene derecho a que se presuma su inocencia mientras no se pruebe su culpabilidad . . .
every person accused of crime has right to that one presumes his/her innocence while not one proves his/her guilt
f. E1:charge(U1,O1:Anyone,E2:penally(offend(O1)))
depend(E3,not(E4)) E3:entitled(O1,presume(U3,innocent(O1)))
E4:prove(U2,guilty(O1,E2))

Support verb constructions (e.g. ‘give education’, ‘subject to limitations’ etc.) are reduced to their nominal object as predicate. Thus the meaning of “enjoy . . . protection” is expressed with the predicate *protect()*:

(3) a. All children . . . shall enjoy the same social protection. [Art. 25.2]
b. 一切儿童. . . 都应享受同样的社会保护
yīqiē értóng . . . dōu yìng xiǎngshòu tóngyàngde shèhùi bǎohù
all child . . . all should enjoy same society protect
c. Alle Kinder . . . genießen den gleichen sozialen Schutz
all children . . . enjoy the same social protection
d. Bhéarfar an chaomhaint shóisialach chéanna don uile leanbh . . .
given the protection social same to all children . . .
e. Todos los niños . . . tienen derecho a igual protección social
all the children . . . have right to equal protection social
f. shall(equally(protect(U1,AllChildren)))

[3] In all examples from the corpus languages are listed in the following order: English, Chinese, German, Irish, Spanish. BEI is an agentive marker.

Many of the conflicts between annotations suggested by individual language glosses are superficial, (e.g. near synonyms such as 'fair' (English) versus 'córa' (Irish: 'just') and 'equitativo' (Spanish: 'equitable')), in which case one of the lexicalisations is arbitrarily chosen. However, when there is a conflict in meaning we use two criteria to decide on a common predicate structure. Majority rule is one – for example in (2), the predicate 'presumed' won out, as it is used in both Spanish and English, and we judged it semantically close to 'regarded' (Chinese) and 'understand' (Irish), but significantly different from the German 'count'. Secondly, subject to majority rule, the most componential logical form available is used, as what is lexicalized in one language as a single verb may be a verb-argument complex in another. Hence, in the example below, the form *expel(U,Person,Country)* as suggested by the German version is preferred over *exile(U,Person)*.

(4) a. No one shall be subjected to ... exile [Art. 9]
b. 任何人不得加以...放逐
rènhé rén bùdé jiāyǐ ... fàngzhú
any person must-not be-made ... exile
c. Niemand darf ... des Landes verwiesen werden
no-one may ... the country expelled be
d. Ní déanfar ... aon duine ... a chur ar deoraíocht
not make ... single person ... that put in exile
e. Nadie podrá ser ... desterrado
no-one will-be-able to-be ... exiled
f. shall(not(expel(U,O1:Anyone,O2:Country))) belong(O1,O2)

We have not yet settled on semantic model of the formal language we use, but it resembles a higher-order logic, or a first-order logic with named Skolem functions.

4 Models of Semantic Roles

Semantic roles were first posited by linguists to describe the nature of meaning relationships among arguments and verbs in sentences. They correspond to a subset of UNL's relations. In this work we concentrate on so-called participant relations (see Table 1) as opposed to the more oblique circumstantial roles such as *manner, purpose* or *condition*, which are less commonly included in role inventories.

The earliest role inventories [9,10] were causally based and mirrored the grammar of argument structure quite closely (consider Fillmore's*agentive, dative,* and *objective* cases). Jackendoff went on to introduce a localist hypothesis [11] (or "thematic hypothesis") based on the extension of verbs (e.g. 'stay', 'go') and prepositions (e.g. 'from', 'to', 'at') of location and movement to more abstract situations (5). For example, information is viewed as*theme* ('story' in (5d) and by extension 'what' in (5g)) and holders can be viewed as*location* ('student' in (5d) and by extension 'document' in (5e) and 'mine' in (5f)).

Table 1. Typical participant roles

Agent	the (typically animate) volitional initiator of an action
Effector	the non-volitional initiator of an action
Patient	the affected party, or undergoer of the action
Theme	the entity whose state, movement or location is described
Experiencer	the entity that perceives the situation
Percept	the entity that is perceived
Recipient	the entity to which another entity is passed
Beneficiary	the entity to whose advantage the action is performed
Instrument	the entity with which the action is performed
Goal	the location towards which an entity moves
Source	the location away from which an entity moves

(5) a. Ciara$_{theme}$ stayed [at work]$_{location}$/[angry]$_{location}$
b. Saoirse$_{theme}$ went [from A$_{source}$ to B$_{goal}$]/[from happy$_{source}$ to sad$_{goal}$]
c. The meeting$_{theme}$ will be at [the main office]$_{location}$/[6pm]$_{location}$
d. The teacher$_{source}$ [gave]/[told] a story$_{theme}$ to her student$_{goal}$
e. Your ideas$_{theme}$ were not included in the document$_{location}$
f. The tricycle$_{theme}$ is mine$_{location}$!
g. They$_{location}$ know what$_{theme}$ they're talking about

There are obvious problems with both the purely causal or localist approaches. It is unclear how a localist scheme would tag an *instrumental* role, and with verbs of perception (e.g. 'hear', 'look') is the *experiencer* the *goal* or the *source*? Nor is it obvious how a purely causal scheme would distinguish between spatial *source* and *goal* (e.g. (5b)).

4.1 Hybrid Models of Roles

Because of these difficulties Jackendoff developed a hybrid, two-tier scheme [2] as part of his semantic representation (Lexical Conceptual Structure, orLCS) with his localist roles on a 'thematic' tier, and causal roles in an orthogonal 'action' tier:

(6) Pete$_{source\&agent}$ kicked the ball$_{theme\&patient}$ down the field$_{goal}$

Saeed [12] suggests completing the Jackendoffian scheme as such, and it is this version that we use here (*actor* is equivalent to *effector*):

Thematic Tier theme, goal, source, location
Action Tier actor, agent, experiencer, patient, beneficiary, instrument

Dorr [3] took Jackendoff's work as a departure point when designing a semantic representation for the lexicon of her interlingual machine translation system UNITRAN, also seeking to "...strike a balance between the causal and motion/location dimensions ...". Her inventory differs in being on a single tier and by incorporating situation-specific roles such as *information* and *percept* (Table 2). She has made an extensive verb lexicon available [13], where each of the 11 thousand entries is annotated with argument syntax and role structure, using verb frames based on Levin's [14] semantic classes.

Table 2. Dorr's LCS roles

AG	agent	**TH**	theme
EXP	experiencer	**INFO**	information
SRC	source	**GOAL**	goal
PERC	perceived item	**PRED**	identificational predicate
LOC	locational predicate	**POSS**	possessional predicate
BEN	benefactive modifier	**ISNTR**	instrument modifier
PROP	event or state	**PURP**	purpose modifier or reason
MANNER	manner	**TIME**	time modifier

Sowa [4] has developed a model of roles for knowledge representation (see Table 3) based on Dick's [15] work in information retrieval, and Somers' Case Grid [16]. Sowa replaces the locational column labels (*Source, Path, Goal, Local*) of Somers and Dick with the four causes from Aristotle's *Metaphysics* (*Initiator, Resource, Goal, Essence*) and introduces six intuitive verb classes, which combined with several additional distinguishing features (such as animacy for differentiating *agent* and *effector*) correspond to more conventional roles.

Table 3. Sowa Roles

	Initiator	**Resource**	**Goal**	**Essence**
Action	Agent, Effector	Instrument	Result, Recipient	Patient, Theme
Process	Agent, Origin	Matter	Result, Recipient	Patient, Theme
Transfer	Agent, Origin	Instrument, Medium	Experiencer, Recipient	Theme
Spatial	Origin	Path	Destination	Location
Temporal	Start	Duration	Completion	PointInTime
Ambient	Origin	Instrument, Matter	Result	Theme

The model of relations used by UNL is more extensive, including logical operators such as AND and OR, and other novel roles such as BAS (*basis for expressing degree*) and SEQ (*sequence*). In particular it gives us a comprehensive treatment of the commitative roles CAG, COB and CAO (*co-agent*, *affected co-thing* and *co-*

thing with attribute) not offered by any of the other schemes examined. They allow us to express the difference in focus between (7a) and (7b):

(7) a. [Fergal and Fergus]$_{OBJ}$ bumped into each other on the street
b. Fergus$_{OBJ}$ bumped into Fergal$_{COB}$ on the street

5 Comparative Evaluation of UNL Relations

The following evaluation is essentially critical, in that we draw attention only to shortcomings of UNL relations or their documentation. The treatment given here of the other inventories will not be comprehensive – rather we will mention them only where they seem to provide a superior solution to UNL. To make a fair comparison, the assignment of roles was carried out on the interlingual form described in Sect. 3, rather than in the context of the semantic representation intended for each inventory (i.e. UNL Expressions, Sowa's Conceptual Graphs, or LCS for the Jackendoff and Dorr schemes). The criteria we used for evaluating role assignments were as follows:

1. Coverage: must be able to assign a role to every argument of every predicate, e.g. in (5b) we saw how a purely causal scheme would fail to express spatial start and end points
2. Differentiation: must be able to assign a unique role to every argument with respect to its predicate, e.g. a scheme without commitative roles would lack differentiation between the syntactic subject and object in (7b)
3. Lack of Ambiguity: must be able to assign a single role unequivocally to each argument - an argument should not fit multiple roles or fall between roles, e.g. a single tier scheme might be unclear on whether 'ball' in (6) is a*theme* or *patient*

Generally, all four inventories performed well on coverage and differentiation, though Jackendoff's small number of roles sometimes presented problems of duplicate assignments to single predicates. Most problems we encountered were with ambiguity. In the following discussion, we make suggestions for alterations to the UNL relations according to the principle that they should adequately and efficiently express generalisations either in semantics (e.g. inferences that can be drawn) or in syntax (e.g. structures that are licensed). We now examine some problematic aspects of the UNL relations in turn, based on the definitions and prototypical examples given in [5,6].

5.1 Causal Relations: AGT, OBJ, AOJ, INS

AGT agent: thing that initiates an action, e.g. "John$_{AGT}$ broke the window"
OBJ affected thing: thing in focus which is directly affected by an event or state, e.g. "write a novel$_{OBJ}$"
AOJ thing with attribute: thing which is in a state or has an attribute, e.g. "This flower$_{AOJ}$ is beautiful"

INS instrument: instrument to carry out an event, e.g. “cut with scissors$_{INS}$”

The OBJ relation (i.e. *patient* role) is used for both clearly affected patients (e.g. the *Anyone* argument of *expel()* in (4f)) and for less affected participants such as the complements of psychological verbs (e.g. the *innocent()* argument of *presume()* in (2f)) and communication verbs (e.g. ‘story’ in (5d)). While this in itself may not be a problem, it may be missing significant syntactic generalisations. In several languages the tendency of a syntactic object to be promoted to a more prominent position, such as subject, seems in part determined by its affectedness. In the examples below the passive (8b) and ‘ba’/‘bei’ (9b, c) variants of *enjoy(I, TheArts)*, all of which promote the object, are anomalous.[4]

(8) a. I$_{AGT}$ enjoy the arts$_{OBJ}$ [variation on Art. 27.1]
b. * the arts$_{OBJ}$ get enjoyed by me$_{AGT}$
(9) a. 我享受艺术
wǒ$_{AGT}$ xiǎngshòu yìshù$_{OBJ}$ [Chinese]
me enjoy art
b. * 艺术被我享受
* yìshù$_{OBJ}$ bèi wǒ$_{AGT}$ xiǎngshòu
art BEI me enjoy
c. * 我把艺术享受
* wǒ$_{AGT}$ bǎ yìshù$_{OBJ}$ xiǎngshòu
me BA art enjoy

Both [2] and [4] give a directional interpretation of these verbs, where the *enjoyer* above is a *goal* and ‘the arts’ a *source*. However examples from our corpus show that using the localist hypothesis (see Sect. 4) with these verbs does not generalise across languages. As we see below (10), in German our enjoyment is ‘in’ the arts, while in Irish almost the reverse is true – the enjoyment is ‘at’ us. As a result we suggest that a simple alternative is to use AOJ (roughly equivalent to *theme*) for non-affected syntactic objects. A more significant reworking would be to add the new roles of PRC (*percept*) and INF (*information*) following the practise of [3].

(10) a. Everyone$_{AGT}$ … to enjoy the arts$_{OBJ}$ … [Art. 27.1]
b. … sich$_{AGT}$ an den Künsten$_{OBJ}$ zu erfreuen … [German]
… self at the arts to enjoy …
c. … áineas na n-ealaíon$_{OBJ}$ a bheith aige$_{AGT}$ … [Irish]
… pleasure of-the arts that be at-him …
d. enjoy(Everyone,TheArts)

[4] A ‘got’ passive is used here as it cannot be mistaken for a non-passive adverbial sentence such as “he was unimpressed by the play”. The star ‘*’ indicates an idiosyncratic or ungrammatical form. The relation annotations shown follow UNL as it stands, rather than our proposals. BEI is an agentive marker and BA is an affectedness marker, both of which promote the object to a preverbal position.

Similarly there is a tendency for non-volitional or inanimate subjects (such as “I$_{AGT}$ think”, “someone$_{AGT}$ is sleeping” and “a process$_{AGT}$ makes something”) to resist being demoted by passivisation or other processes. We suggest thatINS could be used for inanimate initiators such as ‘a process’, or a newEFT (*effector*) relation could be introduced (see [2,4]). The subjects of psychological verbs (e.g. ‘think’ and ‘sleep’ above) could take the AOJ relation, or a newly coined EXP (*experiencer*) relation. However, then we would lose the distinction between the volitional and non-volitional subjects of perception verbs such as ‘listen’/‘hear’ and ‘watch’/‘see’ – the relative merits are debatable.

5.2 Locational Relations: PLF/PLT, SRC/GOL, FRM/TO, PLC

PLF initial place: the place an event begins or a state becomes true, e.g. “come from home$_{PLF}$”
PLT final place: the place an event ends or a state becomes false, e.g. “leave for India$_{PLT}$”
SRC initial state: initial state of object or the thing initially associated with object of an event, e.g. “the light changed from red$_{SRC}$”
GOL final state: final state of an object or the thing finally associated with an object of an event, e.g. “getting better$_{GOL}$”
FRM origin: origin of a thing, e.g. “a letter from him$_{FRM}$”
TO destination: destination of a thing, e.g. “a train to Edinburgh$_{TO}$”
PLC place: place an event occurs or a state is true or a thing exists, e.g. “stay at home$_{PLC}$”

FRM/TO are problematic as they are used for two rather different purposes: describing the concrete path a *Thing* takes, as in the *Country* argument of the *expel()* predicate in (4f); and for the origin of a *Thing*, as seen in *belong()* of the same example. These two functions are treated quite differently in three of the languages examined. Consider possible translations for the constructed examples “the man from London” and “the train from London” respectively:

(11) a. 伦敦的人/ 伦敦来的火车 [Chinese]
lúndūn de rén / lúndūn lái de huǒchē
london DE person / london come DE train
b. an fear as London / an traen ó London [Irish]
the man out-of london / the train from london
c. el hombre de Londres / el tren desde Londres [Spanish]
the man of london / the train from london

While “lúndūn de huǒchē” and “el tren de Londres” are both possible, they can mean several things, including the train both going to or coming from London, much as “the London train” can in English. As predicates exist for some other prepositions, for example *against()*, we suggest using a new predicate called *origin(*AOJ,PLC*)* for describing the provenance of a thing.

For concrete path uses of FRM and TO, we suggest that these relations be conflated with PLF/PLT. None of the other role inventories examined have an event/entity distinction when it comes to locational roles, and the UNL relation PLC can be applied to both *Things* and *Events* (e.g. “a town$_{Thing}$ in Bavaria$_{PLC}$” and “She is$_{Event}$ in Bavaria$_{PLC}$”). In addition, it seems strange that the English prepositions ‘from’ and ‘to’ receive such special treatment, while the similarly common ‘in’ and ‘of’ do not.

Initially, the opposition of PLF/PLT for locations (e.g. (12) “return to his country$_{PLT}$”) with SRC/GOL for states (e.g. (1a) “make education available$_{GOL}$”) seems well justified.

(12) a. Everyone has the right … to return to his country [Art. 13.2]
b. 人人有权…返回他的国家
rénrén yǒu quán … fǎnhuí tā de guójiā
everyone has right return s/he DE country
c. Jeder hat das Recht … in sein Land zurückzukehren
everyone has the right in his/her land to-return
d. Tá ag gach uile dhuine an ceart chun … filleadh ar a thír féin
is at each every person the right to return to his country own
e. Toda persona tiene derecho … a regresar a su país
every person has right to return to his/her country
f. entitled(O1:Everyone,return(O1,O2:Country)) belong(O1,O2)

However, some of the examples given in the documentation blur the distinction, in particular “go to Brussels$_{GOL}$” and “withdraw from the stove$_{SRC}$”. It is not clear to us what basis there is for differentiating between ‘his country’ above as the final state of the entity ‘Everyone’ (GOL) or the final place of the event ‘return’ (PLT) – in both cases the ending of the event and the arrival of the agent happens in the same place at the same time. As a result, we suggest restricting SRC/GOL to non-spatial states only.

A more radical alternative would be to eliminate the SRC/PLF and GOL/PLT distinction altogether. We do not make a similar distinction for static locations (stative “famous in his field” and spatial “live here” both use PLC), and this is supported by [2,3] where spatial and stative end-points are conflated in *source/goal.*

5.3 Miscellaneous: POS, BEN

POS possessor: possessor of a thing, e.g. “the company’s$_{POS}$ building”
BEN beneficiary: not directly related beneficiary or victim of an event or state, e.g. “be fortunate for you$_{BEN}$”

Possession is treated differently in UNL, depending on whether a genitive form (“that is my car$_{POS}$”) or a possessional predicate (“I$_{AGT}$ have a pen$_{OBJ}$”) is used. As with FRM/PLF and TO/PLT this seems like an unnecessary complication that none of the other inventories require. We also have to ask how agentive the subjects of verbs like ‘have’ and ‘own’ are – e.g. in what sense is the subject of “I have no money” an *agent*? Again we see that sentences of this type resist passivisation in English (13b) and the ‘ba’/‘bei’ constructions in Chinese (14b, c). We suggest that possession be annotated as *possess(*POS,AOJ*)* following the practise of [3].

(13) a. … [people$_{AGT}$] own property$_{OBJ}$ … [variation on Art. 17.1]
b. * property$_{OBJ}$ gets owned by people$_{AGT}$
c. own(People,Property)

(14) a. 人所有财产
rén$_{AGT}$ suǒyǒu cáichǎn$_{OBJ}$
people own property
b. * 财产被人所有
* cáichǎn$_{OBJ}$ bèi rén$_{AGT}$ suǒyǒu
property BEI people own
c. * 人把财产所有
* rén$_{AGT}$ bǎ cáichǎn$_{OBJ}$ suǒyǒu
people BA property own

The beneficiary relation BEN works well for adjuncts in English (e.g. "do something for you$_{BEN}$"), but we suggest it be extended to beneficiary syntactic objects. These are currently assigned the GOL relation, even though "make someone$_{GOL}$ a cup of tea" is equivalent to "make a cup of tea for someone$_{BEN}$".

In our opinion a *recipient* relation REC is also needed [3,4]. Note how in English *recipient* arguments ('Anja' in constructed example (15a)) can be syntactic objects, while inanimate arguments that would take a GOL or PLF relation (e.g. 'Munich') cannot – rather an adjunct is necessary, as in "I sent a present to Munich". In German different prepositions and case are used to express these two roles (accusative 'an' for *recipients* and dative 'nach' for *goals*).

(15) a. I sent Anja/*Munich a present
b. Ich habe ein Geschenk an Anja/nach München geschickt
I have a present to Anja/to Munich sent
c. send(I,Present,Anja) / send(I,Present,Munich)

6 Conclusion

In this work a prototype interlingua was manually applied to a five-language parallel corpus to reveal predicate valency patterns. Then several inventories of semantic roles, including a subset of UNL's relations, were assigned to the resulting logical forms. In the subsequent evaluation UNL performed well in terms of coverage and differentiation, but we encountered some problems of ambiguity in the assignment of locational and causal relations. As a result we have some opinions on how parts of the UNL relations might be reformed, based on semantic and syntactic generalisations in the languages examined (English, Chinese, German, Irish and Spanish) and particular structures we encountered in the corpus.

Firstly, we propose that the FRM/TO relations be folded into PLF/PLT, and that the distinction between spatial end-points PLF/PLC and stative end-points SRC/GOL be firmed up. We also propose redrawing the lines between causal relations – specifically non-affected *objects* (e.g. the syntactic objects of communication verbs) should be assigned AOJ rather than OBJ, and non-volitional *agents* should be assigned INS rather than AGT. We propose extending the usage of BEN from adjuncts to also cover syntactic objects, and using POS for verbal as well as nominal structures that express possession. Finally we suggest several new situation specific roles (*recipient, effector, experiencer*) and explain how they might be of use in future versions of UNL.

References

1. United Nations General Assembly: Universal declaration of human rights. http://www.unhchr.ch/udhr/navigate/alpha.htm (1948) [Viewed December 2004].
2. Jackendoff, R.: Semantic Structures. MIT Press, Cambridge (1990)
3. Dorr, B.J.: Machine Translation: A View from the Lexicon. MIT Press, Cambridge (1993)
4. Sowa, J.F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks/Cole, London (2000)
5. UNL Centre: The universal networking language specifications 3.2. http://www.undl.org/unlsys/unl/UNL Specifications.htm (2003) [Viewed December 2004].
6. UNL Centre: UNL manual. http://www.undl.org/unlsys/unlman/index.html (2001) [Viewed December 2004].
7. Kingsbury, P.: Propbank annotation guidelines. http://www.cis.upenn.edu/~ ace/propbank-guidelines-feb02.pdf (2002) [Viewed December 2004].
8. Kintsch, W.: Comprehension: A Paradigm for Cognition. Cambridge University Press, Cambridge (1998)
9. Gruber, J.S.: Studies in Lexical Relations. Indiana University Linguistics Club, Bloomington (1965) Reprint of PhD Thesis.
10. Fillmore, C.J.: The case for case. In Bach, E., Harms, R.T., eds.: Universals in Linguistic Theory. Holt, Rinehart and Winston, New York (1968) 1–92
11. Jackendoff, R.: Semantic Interpretation in Generative Grammar. MIT Press, Cambridge (1972)
12. Saeed, J.I.: Semantics. Blackwell, Oxford (1997)
13. Dorr, B.J.: LCS database documentation. http://www.umiacs.umd.edu/~ bonnie/LCS_Database_Documentation.html (2001) [Viewed December 2004].
14. Levin, B.: English Verb Classes and Alternations. University of Chicago Press, Chicago (1993)
15. Dick, J.: A Conceptual, Case-Relation Representation of Text for Intelligent Retrieval. PhD thesis, Department of Computer Science, University of Toronto (1991)
16. Somers, H.: Valency and Case in Computational Linguistics. Edinburgh University Press, Edinburgh (1987)

Some Controversial Issues of UNL: Linguistic Aspects

Igor Boguslavsky

Universidad Politécnica de Madrid (Spain)/IITP RAS (Russia)
igor@opera.dia.fi.upm.es

Abstract. We discuss several linguistic aspects of the Universal Networking Language (UNL); in particular, those connected with Universal Words (UWs), UNL relations, and hypernodes. On the one hand, the language should be rich enough and provide sufficient means to express the knowledge that might be required in the applications it is intended for. On the other hand, it should be simple enough to allow uniform and consistent use across languages and by all encoders. The major expressive device of UNL used for overcoming lexical divergence between languages is so-called restrictions. They have three functions, which are relatively independent of each other: the ontological function, the semantic function, and the argument frame function. We discuss various types of restrictions and propose new expressive means for describing UWs. Sample dictionary entries are given which incorporate our proposals. We propose several new UNL relations and discuss when and how hypernodes should be introduced.

1 Background

Among many problems that developers and users of a meaning representation language are facing, two somewhat conflicting requirements are standing out. On the one hand, the language should be rich enough and provide sufficient means to express the knowledge that might be required in the applications it is intended for. The more complex and knowledge-demanding the application, the more complex the design of the meaning representation language becomes. On the other hand, it should be simple enough to allow uniform and consistent use across languages and by all encoders. In the case of UNL, the latter problem is particularly serious, since the encoders work in different countries, belong to different linguistic schools, and have different linguistic traditions. Therefore, uniform understanding and use of UNL by all partners is difficult to achieve.

Since the start of the project in 1996, a large number of UNL-encoded documents have been accumulated that were produced by the project participants from 16 language groups each working on its native language. The analysis of these documents clearly shows two things: *UNL is still lacking means to express meaning adequately*, and *there is not enough uniformity in the UNL use among the partners*. To some extent, UNL has developed its own dialects. Despite the existence of the UNL Specifications, divergences between the dialects tend to grow. This tendency clearly manifests itself in the fact that all deconverters (=generators) are doing much better when dealing with the UNL documents produced by the authors of the deconverter than

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 77–100.

with those provided by other teams. If it goes on this way, the dialects will soon become hardly understandable by the deconverters and we will need special modules to translate from one UNL dialect to another.

These problems were raised at several discussions at the UNL workshops and working sessions. Of particular importance was the “Forum Barcelona 2004” project carried out in 2001 by the UNL groups from France, India, Italy, Russia and Spain. During this work a number of texts were encoded to UNL by project participants and each text was extensively debated. Participants of the discussion have been: Ramon Armada, Pushpak Bhattacharyya, Etienne Blanc, Igor Boguslavsky, Carolina Gallardo, Luis Iraola, and Irina Prodanoff. The results of this debate were presented in [1] and at the UNL conference in Suzhou, [2]. In this paper, I will summarize the understanding of UNL that took shape in the course of discussions and put forward some proposals on the linguistic aspects of UNL.

The paper is organized as follows. In section 2, some general remarks will be made concerning the requirements imposed on UNL representations. Section 3 will be devoted to Universal Words. In section 4, I will give some comments on the semantic categories of UWs which constitute upper levels of the UNL Knowledge Base. Aside from that, there will be no special discussion of the problems connected with the UNL Knowledge Base and Master Dictionary. Issues of UNL relations will be discussed in section 5. Finally, in section 6 I will speak about hypernodes (scopes).

I will not give any introduction to UNL. It can be found, for example, in [3], [4], [5]. It is expected that the reader have some preliminary knowledge of UNL, at least as far as the UNL Specifications are concerned [6].

2 General remarks

UNL representations (UNLR) can be evaluated from the points of view: correctness and adequacy. A UNLR is correct if it conforms to UNL specifications. To be adequate, the UNLR should contain enough information and be convenient for the applications it is intended to serve. UNL is conceived as a meaning representation language applicable in a wide range of applications – multilingual generation, machine translation, information retrieval, text summarization, question answering. I will discuss it mainly from the perspective of one of them – multilingual generation of UNL documents for the dissemination of information in the Internet. This is the application that received most attention in the UNL development so far and, at the same time, it is one of the most demanding.

To be adequate for multilingual generation, a UNLR should meet at least two requirements:

- it should preserve the meaning of the source text to a reasonable extent (i.e. without a significant loss);
- it should permit generation of the text bearing this meaning in all working languages.

Since the enconversion, i.e. transformation of the source text into UNLR, is not supposed to be fully automatic, we can address our encoding recommendations to a human who will produce UNLRs with the help of special tools. These tools may range from more or less sophisticated editors (cf. for example the EditorUNL developed by the Spanish group, [7]) to semi-automatic enconverters (cf. for example the UNL module of the ETAP-3 system developed by the Russian group [8]).

The UNLRs need not be literal. They should not necessarily preserve the structure of the original sentence, nor its lexical composition. The only thing required of them is to represent the original meaning in a satisfactory way. To do it, the UNL writer may paraphrase the text in any way he/she finds convenient, provided the meaning of the original and its communicative intention remain intact. In particular, long sentences may be divided into several shorter ones. Language-specific syntactic constructions and idioms may be replaced with simpler constructions and non-idiomatic synonymous expressions, or an equivalent English idiom, should it exist.

To give a simple example, consider Spanish sentence (1):

(1) *Los estudiantes tenemos que trabajar mucho.*

Literally, the sentence reads: ‘the students have to work much’. But this is not the whole meaning of the sentence. An idiosyncratic feature of this construction is that the predicate (*tenemos que* ‘have to’) has the form of the first person plural (= ‘we have to’) and therefore does not agree in the grammatical category of person with the subject (*estudiantes* ‘students’). Due to this grammatical peculiarity, the meaning of (1) is ‘we, students, have to work much’. What should be the adequate UNLR for (1)? A straightforward solution would be (1a) that directly reflects the structure of the source sentence:

(1a) aoj(must.@entry.@1-person, student.@pl)
obj(must.@entry.@1-person, work)
man(must.@entry.@1-person, much)

However, this UNLR should be discarded as too specific. It is the idiosyncratic property of Spanish to encode the information on the subject (‘we’) in the verb form. UNL should express this information in a less language-specific way:

(1b) aoj(must.@entry, we)
cnt(student, we)
obj(must.@entry, work(icl>do))
man(work(icl>do), much)
(1b = ‘we being students must work much’).

However, the freedom of replacing phrases with their paraphrases should be used with great caution. For example, special terms cannot be paraphrased and must be represented in the form in which they exist in English. For instance, *sustainable development* should not be represented as obj(sustain(icl>maintain).@ability, development.@entry). This UNLR, though it conveys a meaning close to the original phrase – “development that can be sustained”, – is unacceptable as a representation for a term.

3 Universal Words (UW)

As an element of the dictionary, a UW consists of two major parts: the headword and the restrictions[1].

3.1 Headwords

As defined in the UNL Specifications, any English word, phrase or sentence can be a headword for a UW. UNL corpora abound in headwords consisting of more than one word, such as *Ministry of Foreign Affaires*, *Telecommunication Development Bureau, sustainable development*, *week-long feast*, etc. In our opinion, multi-word headwords should be introduced with much care. When a multi-word expression is compositional, i.e. when its meaning is representable as a combination of meanings of words it is composed of, it is better to represent it as a combination of UWs linked with appropriate relations and not as one multi-word UW.

Examples[2]:

(2) *sustainable development*
(2a) mod(development, sustainable)
(3) *week-long feast*
(3a) dur(feast, week)
qua(week, 1)

An example of a non-compositional phrase that could with good reason generate a multi-word headword is (4):

(4) *look for*
(4a) look for(icl>do,agt>thing,obj>thing).

However, even in this case a multi-word UW is not the only alternative. One can consider *look for* as a realization of a special lexical meaning of *look*, but in this case the meaning should be accordingly restricted:

(4b) look(icl>search>do,agt>thing,obj>thing).

The reason for avoiding multi-word headwords is obvious: if any free word combination can be made into a UW, one can hardly hope that other partners will have matching UWs in their dictionaries.

On the other hand, the idea behind the multi-word UWs is to express the fact that they denote a single concept. It might be useful to keep this information. Then, a convenient compromise might be to enclose the UNLR in a scope:

(5a) mod:01(ministry.@entry, affair.@pl)
mod:01(affair.@pl,foreign)

[1] As an element of UNLR, a UW can be supplied with additional pieces of information such as ID number and attributes.

[2] For simplicity's sake, here and in some other examples I will omit restrictions that are not directly relevant for the discussion.

Another solution (proposed by Ch. Boitet) is to allow UWs to have internal structure:

(5b) mod(ministry,affair.@pl)&mod(affair.@pl, foreign)

It should be noted, however, that the second solution requires a considerable modification of the specifications and of the EnCo/DeCo software.

3.2 Restrictions

UW restrictions have three functions:

- Ontological function: locate the UW in the Knowledge Base. This is needed, in particular, to ensure understanding of the UW in the case that it is absent in the dictionaries of some working languages and to help semantic inference.
- Semantic function: restrict the meaning of the headword. This is needed, in particular, to ensure disambiguation of the headword and selection of the translation equivalent.
- Argument frame function: provide the argument frame for the UW.

It is important to emphasize, that the requirements imposed by these functions do not always coincide. A restriction that is good for one purpose is not necessarily adequate for another. For example, restrictions of the type (icl>thing) or (icl>how) or (icl>do) are often very efficient for disambiguation, since they differentiate nominal, adverbial and verbal meanings from each other. At the same time, they will not help us much to translate a UW, if we don't have this UW in the dictionary.

On the other hand, the word *pern* is monosemic and does not need disambiguation. But if we don't have an exhaustive list of different varieties of birds in the dictionary, the restriction (icl>bird) will be very helpful to provide an understandable translation for this word. It can also be of help in other situations in which it is useful to know that the word denotes a bird.

The third function of restrictions – specification of the argument frame of the word – should also be clearly separated from other functions. One may wish to restrict the meaning by specifying some semantic relation (first function), but it does not necessarily imply that this relation makes part of the argument frame of the word. The English verb *to land* denotes reaching the land both from the sky (*The airplane landed on time*) and from water (*We landed on a lonely island in the middle of the ocean*). In these situations, Russian uses different verbs – *prizemljat'sja* and *vysazhivat'sja,* respectively. To construct UWs for these verbs, we need to restrict the meaning of *to land.* An obvious way to do so would be to indicate the initial point of the movement (src relation): *prizemljat'sja* = land(src>sky); *vysazhivat'sja* = land(src>water). However, these verbs do not have argument slots for the initial point of movement.

Restrictions on the basis of which the UWs are arranged in the KB will be designated **KB restrictions**. Restrictions oriented primarily towards the second goal will be called **semantic restrictions.** Restrictions which specify the argument frame will be referred to as **argument frame restrictions**. A restriction may serve more than one goal. For example, restrictions in the UWs orange(icl>fruit), orange(icl>tree), orange(icl>colour) can equally well differentiate three different meanings of the noun *orange* and specify the KB position of each of them. However, we should keep in mind that in the general case semantic, argument frame and KB restrictions do not coincide.

The UNL dictionaries must have means by which we could distinguish between these three types of restrictions. KB restrictions are clearly separated from other types of restrictions, since they are only represented in the Master Dictionary and not in the UW dictionary. As a matter of fact, the difference between the Master Dictionary and the UW dictionary boils down to the presence/absence of the KB restrictions. As for the argument frame restrictions, in the present version of the UW dictionary they are represented very poorly and are not separated from semantic restrictions.

We will discuss semantic and argument frame restrictions in sections 3.2.1 and 3.2.2 respectively.

3.2.1 Semantic restrictions

As mentioned above, the function of semantic restrictions is to effectively separate the meaning of the UW from all other meanings which the headword may have. The major requirement imposed on semantic restrictions is as follows. Restrictions ascribed to a UW should not be equally applicable to other meanings of the same headword. For example, the UW <u>people(icl>human)</u> does not meet this condition, since the headword *people* has two different meanings, and both of them are covered by restriction <u>(icl>human)</u>: ‘persons’ (as in *many people*) and ‘nation’ (as in *peoples of Africa*). Similarly, all meanings of the noun *operator* can be characterized as belonging to the “thing” category. Therefore, restriction (icl>thing) is too broad and should be narrowed down. *Operator* in the context (6a) corresponds to UW (6b), and in the context (7a) – to UW (7b).

(6a) *a long distance operator*
(6b) <u>operator(icl>human)</u>
(7a) *addition operator*
(7b) <u>operator(icl>abstract thing).</u>

In order to conform to this requirement, to be consistent and to ensure similar decisions as to what meanings an English word has, it is expedient that all the partners use the same one or two good English dictionaries, preferably available on-line.

In inventing semantic restrictions for UWs, we should adopt a certain procedure which would make it possible for different UNL writers to produce the same or very similar UWs for the same meanings. As a first step towards elaborating such a procedure, it is proposed to proceed along the following lines:

- If a headword is unambiguous in English, and the meaning of this English word expresses the required meaning with sufficient precision, no semantic restrictions are needed. Example: *September*. (<u>NB: the absence of semantic restrictions does not mean that we should not supply KB restrictions in the master dictionary – September{icl>month}</u>).
- If a headword has several meanings in English, and one of them corresponds to the required meaning with sufficient precision, we have to compose a restriction in such a way as to distinguish this meaning from other meanings of the headword.

For example: answer(icl>do) (for cases like *answer questions*) – answer(icl>be) (for cases like *answer expectations*) – answer(icl>thing) (for cases like *know the answer*)[3].

- If no English word exactly corresponds to the meaning of the headword we need, we have to find the closest more general English word available and restrict it accordingly. Example: Russian *zhenit'sja* – marry(agt>male), *vyxodit' zamuzh* – marry(agt>female).

As the last example shows, a restriction can be formulated in terms of any relation which can connect UWs in a UNLR (agt, obj, gol, etc.). Besides them, there are several other relations which can only be used to restrict meanings. These are: icl, pof, equ, ant, com. Relations icl and pof have been envisaged by the Specifications from the very beginning. Relations equ[4], ant and com are proposed for inclusion now. Some comments on these relations are appropriate.

UNL makes extensive use of two traditional types of paradigmatic relations: hyperonymy (class/subclass relation, icl) and meronymy (part/whole relation, pof). Examples: September(icl>month), month(pof>year). However, it is often difficult to find a more general term (hyperonym) that, on the one hand, could distinguish different meanings of the word and, on the other hand, is easy to understand. In this case, it is convenient to recur to a synonym. I think it is worth introducing to UNL the traditional distinction between synonymy and hyperonymy, which is obviously extremely useful for inference, for example.

As in the case of more general terms, restrictions based on synonyms should not be equally applicable to various meanings of the headword. For example, UW wealth(equ>richness) does not meet the above requirement. The words *wealth* and *richness* both have two meanings – 'having many valuable things at one's possession' (*wealth/richness of the nation*) and 'abundance of something' (*butterfly species richness - the wealth of rainforest resources*) – and this restriction alone does not differentiate them. Therefore, some other restrictions should be used, e.g. wealth(icl>well-to-do-ness) – wealth(equ>abundance, obj>thing).

Besides icl, pof and equ relations, we propose to use two more relations. One of them is the traditional antonymy relation, which in some cases may conveniently supplement synonymy. Example:

(8a) *poor quality* ⇒ poor(equ>bad),
(8b) *poor people* ⇒ poor(ant>rich).

Nevertheless, if one takes the task of distinguishing between close lexical meanings of the same word seriously, one will find that the available relations are not sufficient. In many cases, distinctions between the meanings cannot be naturally reduced to rigid categories of hyperonymy, meronymy, synonymy or antonymy. For these cases, we propose to introduce a new relation – com, standing for 'component'. We

[3] This example shows that it is often useful to give examples and/or comments, to make UWs more easily understandable. We will come back to this in 3.3.

[4] The equ relation, originally included in the list of relations, is absent in the latest version of the UNL Specifications (v. 3.2). However, even when it existed, it had a different meaning from what we propose now. It was only used to introduce a definition of an abbreviation: UNL(equ>Universal Networking Language).

will write A(com>B) if B is an (unspecified) important component of the meaning of A. Examples:

(9a) *seniority* ('being older', as in *He is chairman by seniority*) ⇒ seniority(icl>property, com>age);

(9b) *seniority* ('having higher rank by reason of longer service', as in *workers with less than 5 years' seniority*) ⇒ seniority(icl>property, icl>rank);

(10a) *sensational* ('causing intense interest', as in *The effect of the discovery was sensational*) ⇒ sensational(mod<thing, com>interest);

(10b) *sensational* ('very good or impressive', as in *You look sensational in this dress*) ⇒ sensational(mod<thing, icl>good)

(11a) *series* ('several events or actions happening one after another', as in *a series of years*) ⇒ series(icl>set>abstract thing);

(11b) *series* ('a number of connected social events (tournaments, lectures, TV-programmes)', as in *League Championship Series*) ⇒ series(icl>set>abstract thing, com>social).

3.2.2 Argument frame restrictions[5].

UNL as a meaning representation language should have an ability to draw a distinction between the argument and non-argument links of predicates. It is well known that for correct generation, as well as for a wide range of other NLP purposes it is essential to know the argument structure of the predicates and the way each argument is expressed in the sentence. This idea does not seem to require justification, yet it has not been implemented in UNL so far. Since there is no consensus in the UNL community as to what an argument of the predicate is, I will briefly present the problem as I see it.

A is an argument of predicate *L* if *A* is integral to the meaning of *L*. *A* is **semantically obligatory**. This means that *L* cannot be semantically defined, or explained, without *A* being mentioned. *A* is **not always syntactically obligatory**. This means that some arguments can remain unmentioned in a sentence. As an example, let us consider the verb *to borrow.* To define the situation of borrowing, four arguments are necessary.

X borrows Y from Z for W (e.g. *He borrowed a bicycle from his friend for a couple of days*) =

- 'Z owns Y'
- 'X makes Z to give him Y'
- 'X promises Z to give Y back after period W expires'.

All four arguments are semantically obligatory, since borrowing cannot take place without any one of them. None of them is syntactically obligatory. In (12a) *W* is not mentioned. In (12b) no arguments at all are represented.

(12a) *He never borrows money from his friends.*

[5] The problem of arguments in UNL has been raised on several occasions. Our presentation here is a further elaboration of the proposal outlined in [9].

(12b) *Borrowing is tempting but dangerous.*

Still, in both (12a) and (12b), a situation of borrowing is referred to which presupposes the existence of all the four arguments. To feel the difference between arguments and non-arguments better, note that any action has a certain duration, e.g.

(13) *He has been sleeping for three hours.*

Therefore, the duration role is assigned in the Knowledge Base to the topmost UW denoting an action (do{dur>time}) and is inherited by all UWs lying below, including *borrow*. On the other hand, as definition (11) shows, *borrow* has a semantic argument *W* with the role 'duration'. Two functions of the duration with respect to *borrow* (argument and non-argument functions) can be exemplified with sentences (14a) and (14b):

(14a) *John borrowed $10,000 for three years.*
(14b) *John has been borrowing money for three years.*

In (14a) it is a semantic argument and characterizes the terms of the loan. In (14b) it is a free adverbial and characterizes the period of time in which borrowings took place; the terms of each loan are not specified. It is obvious that the difference between arguments and non-arguments is important for semantic processing: (14a) can answer the question on the terms of the loan, while (14b) cannot do so. As a matter of fact, the semantic argument of duration and the adverbial modifier can very well coexist in a sentence: *He has been borrowing money until payday all his life*.

Another example: any object can be used for some purpose. For example, we can use a stone to drive a nail, if no hammer is available. Does it mean that *stone* has a purpose argument? No. A stone has no obligatory conceptual link with the purpose. On the other hand, *a method* has. A method cannot exist without a purpose. Therefore, seemingly similar phrases like (15a) and (15b)

(15a) *a stone for driving nails*
(15b) *a method for calculating taxes*

differ with respect to arguments.

The UNL dictionary does not contain explicit information on the argument structure. Neither semantic nor ontological restrictions are meant for this purpose. To come back to the example above, each object can be used for some purpose, and therefore the purpose relation (pur) is assigned in the KB to UW thing, and is inherited by all UWs lying below. Nevertheless, as we showed above, some of the things do have a purpose argument, while some others do not.

How can arguments be introduced into UNL? First of all, argument structures should be assigned to all those UWs that have arguments. It can be done by means of restrictions, but argument frame restrictions should be clearly differentiated from semantic and ontological ones. One possible way to achieve this is to supply argument frame restrictions with a special symbol (@A, @B, @C). Then, the UW for *borrow* will look as follows:

(16) borrow(icl>do,agt.@A>volitional thing,obj.@B>thing,src.@C>volitional thing,dur>@D>time)

However, in the general case, the marking of the argument frame in the UW is not sufficient. In some cases, the same relation can attach to a UW both an argument, and a free adjunct – cf. (14a)-(14b) above. I will give another example to show that this situation is not unique. Emotional states (*be angry, be afraid, be surprised,* etc.) have an argument denoting the cause of the state. In sentence (17)

(17) *She is afraid to go out alone at night*

going out alone at night is the cause of her being in the state of fear. Therefore, relation rsn (=‘cause, reason’) between *afraid* and *go out alone at night* is appropriate. On the other hand, *afraid* can have a non-argument cause, as in (18):

(18) *She is afraid (to go out alone at night), because this area is not very safe.*

Even if *afraid* is assigned a cause as one of the arguments, we should know whether or not a rsn-link in the UNLR denotes this argument. This means that in order to generate correct text, it is not sufficient to know the semantic role of word A with respect to B. One also needs to know whether or not A is an argument of B.

A possible solution would be to mark the argument relation in the UNLR with a special label. Then, a relevant fragment of sentence (18) will be represented as (19)

(19) rsn.@A(afraid(rsn.@A>uw), go_out)
rsn(afraid(rsn.@A>uw), safe)

Obviously, it only makes sense, when the relation in question can in principle fulfill both functions. If a relation is unambiguously argumentative (as agt or obj), this label is superfluous.

This example shows also that the difference between arguments and non-arguments is essential for correct deconversion, since they can be expressed differently. In English, the rsn-argument of *afraid* cannot be expressed by preposition *because of,* which is typical for this role:

(20a) *She is afraid of darkness*.
(20b) **She is afraid because of darkness.*

3.3 Samples of UW dictionary entries.

As of now (end of 2004), UNL partners have collected large UNL dictionaries, that is sets of UWs linked with words of their languages. The value of these resources is impaired by several facts:

1. UWs do not sufficiently differentiate between different meanings of the headword.
2. There is no systematic information on the arguments.
3. Some restrictions are difficult to understand.
4. Dictionaries of different groups are not harmonized.

Ways to solve the first and the second problems have already been discussed above. The third shortcoming can be overcome if the dictionary entry is supplied with examples and/or comments that illustrate and clarify UWs in non-obvious cases. The fourth problem requires that all the UW dictionaries be put together and made a unified UNL lexical resource[6]. The table below shows what this resource could look like. It presents a group of words beginning with the letter *L*. For the reader's convenience, examples and comments to UWs are given in a separate column. Translation equivalents are only given for Russian and Spanish, but obviously other working languages should also be added.

Symbols outside the Specifications:
@ex – example
@com – comment
uw – any UW
* – a string of characters
icl>adj – restriction for all types of adjectival UWs (see 4.2 below)
asp – aspect relation (see 5.3 below)

Table 1. Samples of multilingual UNL dictionary entries.

UW	Examples&Comments	Russian	Spanish
label(icl>conctrete thing)	@ex: *a luggage label*	ярлык	etiqueta
label(icl>write>do, agt.@A>volitional thing, obj.@B>thing, cob.@C>*)	@ex: *Label the diagram* (obj) *as shown. The file* (obj) *was labelled "Top secret"* (cob) @com: cob>*: * is used because not only UWs are possible here but any string of symbols	поме-чать	etiquetar
label(icl>name>do, agt.@A>volitional thing, obj.@B>thing, cob.@C>*)	@ex: *the newspapers* (agt) *labelled him* (obj) *a troublemaker* (cob)	назы-вать	nombrar
laboratory(icl>institution>organization, pur.@A>uw)	@ex: *The National* (mod) *Renewable Energy* (pur) *Laboratory; laboratory for renewable energy research and development* (pur)	лаборатория	laboratorio
labor_day(icl>holiday>date)		День труда	Día del trabajo
labour_intensive(icl>adj)	@ex: *labour intensive*	трудоемкий	que necesita mucho trabajo
laborious(icl>difficult>adj)	@ex: *laborious task*	трудный	laborioso

[6] The idea to construct a multilingual dictionary with UWs serving as interlingual index has been put forward within the PAPILLON project in [10].

laborious(icl>slow>adj)	@ex: *laborious progress* @com: done slowly and with difficulty	медленный	penoso
labour_union(icl> institution>organization)		профсоюз	sindicato
labour(icl>work>action, agt.@A>person)	@ex: *building involves a lot of manual labour, his* (UW=he) (agt) *labour*	Труд	trabajo
labour(icl>person>living thing)	@ex: *skilled labour, labour shortage*	рабочая сила	mano de obra
labour(icl>event> abstract thing, agt>living thing)	@com: a process in which a baby is born	роды	parto
labour(icl>do, agt.@A>person>living thing, obj>thing)	@ex: *to labour at a task* (obj), *over the report* (obj)	трудиться	afanarse
labour(icl>party> organization)	@com: the British Labour party	лейбористская партия	Partido Laborista
labour(icl>adj)	@com: connected with the British Labour party	лейбористский	laborista
lack(icl>abstract thing, aoj.@A>thing, obj.@B>thing)	@ex: *lack of food* (obj); *their* (aoj) *lack of patience* (obj)	нехватка	falta
lack(icl>be, aoj.@A>thing, obj.@B>thing)	@ex: *he* (aoj) *lacks courage* (obj)	недоставать	faltar, carecer
lacking(icl>adj, obj.@A>thing)	@ex: *the crew is lacking in beef* (obj)	лишенный	carente (de)
lag(icl>period>time, obj.@A>time)	@ex: *a time* (mod) *lag of one month* (obj)	отставание	retraso
lag(icl>occur, equ>lag behind, obj.@A>thing, asp.@B>thing)	@ex: *Britain* (obj) *was lagging in the space race* (asp)	отставать	quedarse atras
lag behind(icl>occur, equ>lag, obj.@A>thing, asp.@B>thing)	@ex: *they* (obj) *worked badly and lagged behind; lag behind in development* (asp)	отставать	quedarse atras, retrasar-se
land(icl>area,ant>sea, ant>air)	@ex: *to travel by land* (via)	суша	tierra
land(icl>ground>thing)	@ex: *fertile land* @com: mostly when used for farming or building on	земля	tierra
land(icl>country>region)	@ex: *native land, visit distant lands*	страна	país

land(icl>property> abstract thing)	@ex: *his lands extend for several miles*	земля	terreno
land(icl>do, agt.@A>thing, plc.@B>thing,src>sky)	@ex: *the plane <we>* (agt) *landed at the Geneva airport* (plc)	приземляться	aterrizar
land(icl>do, agt.@A>volitional thing, obj.@B>functional thing, plc.@C>thing,src>sky)	@ex: *the crew* (agt) *finally landed the plane* (obj) *on the soft part of the runway* (plc)	сажать	aterrizar
land(icl>do, agt.@A>volitional thing, plc.@B>thing,src>water)	@ex: *land on a lonely island* (plc)	высаживаться	llegar a tierra
land(icl>do, agt.@A>volitional thing, obj.@B>living thing, plc.@B>thing,src>water)	@ex: *land somebody* (obj) *on a lonely island* (plc)	высаживать	poner en tierra
land(icl>do, agt.@A>volitional thing, obj.@B>concrete thing, plc.@C>thing)	@ex: *to land containers* (obj) *on the shore* (plc)	выгружать	poner en tierra
last(icl>recent>adj)	@ex: *last night, last edition, last harvest*	последний	pasado, ultimo
last(icl>adj,ant>first)	@ex: *last page, last bus*	последний	ultimo
last(icl>occur, obj.@A>abstract thing, dur.@B>period>time)	@ex: *the hot weather* (obj) *lasted for the whole month* (dur)	длиться	durar
lay(icl>put>do, agt.@A>living thing, obj.@>concrete thing, plt.@C>thing)	@ex: *lay the dress* (obj) *on the bed* (plt)	класть	poner
lay(icl>set>do, agt.@A>person, obj.@B>table)	@ex: *lay the table*	накрывать	poner
lay(icl>fix>do, agt.@A>person, obj.@B>thing, plt.@C>thing)	@ex: *lay the carpet* (obj) *on the floor* (plt), *lay bricks, pipelines* (obj)	укладывать	poner
lay(icl>produce>do, agt.@A>bird, obj.@B>egg)	@ex: *lay eggs*	откладывать	poner

Revising the UNL lexical resources along the lines suggested above is, in my opinion, the most important task facing the UNL community at the moment. It can only be solved if all the partners join their efforts.

4 Semantic categories of UW

Semantic classification of UWs is embodied in the Knowledge Base. This is a very large topic, which I cannot discuss at full scale. Here I will only touch upon the upper levels of this classification. All UWs are divided into four major classes: verbal, nominal, adjectival and adverbial concepts. Of these classes, I will only deal with two – the verbal and the adjectival concepts.

4.1 Verbal concepts

In linguistics, there are various classifications of predicates based on their fundamental semantic properties. The most important classes of predicates are:

(a) **actions**: they have an active initiator – an agent (normally, a human) that performs the action as a step to achieving some goal. Most of the actions have a natural limit - a point in its development at which the goal has been achieved and after which the action cannot continue. Examples: *kill, write, eat, solve.*

(b) **activities**: they denote a set of actions, often heterogeneous, that have a common goal. Examples: *work, trade, cooperate.*

(c) **events**: they have no agent and denote a situation in which something happens to an object. Examples: *the bridge broke, an accident happened, the stone fell.*

(d) **processes**: they have no agent and denote a situation that occupies a certain time span in which an object undergoes a change. Examples: *the tree grows, the temperature rises.*

(e) **states**: they differ from the processes in that they are homogeneous (do not denote a change). They characterize a thing during a certain period of its existence. Examples: *see, hear, ache, know, want, wait, hope, proud.*

(f) **properties**: they differ from the states in that they are atemporal, i.e. they normally characterize things during the whole period of their existence. They are often expressed by adjectives. Examples: *blind, red, clever.*

(g) **relations**: they differ from the properties in that they do not characterize a thing but a relation between two or more things. They are often expressed by nouns. Examples: *love, hate, equal, friend, father.*

In UNL, not all of these semantic types are distinguished – only three. All verbal concepts group into three classes designated by restrictions (icl>do), (icl>occur) and (icl>be).

Class (icl>do) contains actions and activities. They are initiated by some active force which can be either a voluntary human (or autonomous mechanism, as e.g. computer) or some inanimate factor: *He solved the problem. The storm broke the tree. The silence frightened the child.*

Class (icl>occur) consists of events and processes, which are not regarded as initiated by an active force.

Class (icl>be) is composed of states, properties and relations.
Some examples. *Include* is an action in (21a) but not in (21b):

(21a) *I included* (icl>do) *his name in the list.*
(21b) *The list includes* (icl>be) *his name*.
Open is an action in (22a), but not in (22b)
(22a) *I opened* (icl>do) *the door*.
(22b) *The window opened* (icl>occur),
because in the latter case no initiator is necessarily implied. Even in the sentence
(23a) *The forum opened*
we are dealing with an (icl>occur) verb, because it does not mean exactly the same as
(23b) *The forum was opened* (icl>do).
(23b) definitely says that somebody opened the forum, while (23a) doesn't say anything to this effect and in this sense is similar to (22b). If a UNL writer wishes to ignore this difference, he may choose any option.

It is natural that the semantic type of the predicates should agree with semantic relations that link them to their main argument[7]. Obviously enough, the main argument of actions is an agent (agt), events and processes require obj-relation, while states, properties and relations attach their main argument by the aoj-relation. For this reason, predicates like *know* and *regret* which denote a state and not an action cannot be heads of the agt-relation.

4.2 Adjectival concepts.

According to the UW Specification, all adjectival concepts are divided into two classes. The first class is characterized by restriction (aoj>thing) and the second by restriction (mod<thing). The difference between these classes is explained in the following way: "(aoj>thing) is for expressing a predicative concept, whereas (mod>thing) is for expressing a restrictive concept" [11]. This formulation introduces an opposition "predicative" vs. "restrictive" which is based on heterogeneous criteria. This is logically unacceptable. Let us consider the facts with some detail.

We are dealing here with two different properties of adjectives:

(a) a syntactic property: it is the question of whether the adjective is used predicatively (*Greeks are wise)* or attributively (*the wise Greeks*);
(b) a semantic property: which shows what the adjective means when used attributively: restriction or qualification.

We should clearly distinguish between the syntactic construction in which a modifier is preferably used (attributive vs. predicative) and the meaning it conveys (restrictive vs. qualificative). I will begin with the meaning.

[7] By way of simplification, one can say that the main argument is the one that normally corresponds to the syntactic subject.

This semantic difference was discussed at least as far back as 1933 by O. Jespersen [12]. This is what we find in a modern English grammar: from the semantic point of view, «the modification can be restrictive or non-restrictive [= qualificative – IB]. That is, the head [the modified noun - IB] can be viewed as a member of a class, which can be identified only through the modification that has been supplied (restrictive). Or, the head can be viewed as unique or as a member of a class that has been independently identified (for example, in a preceding sentence); any modification given to such a head is additional information which is not essential for identifying the head, and we call it non-restrictive». [13, 13.3]. For example, the adjective *wise* in the sentence (24) can be understood both restrictively and non-restrictively.

(24) *Wise Greeks diluted wine with water*

(24a) restrictive interpretation: 'Those Greeks who were wise diluted wine with water. Silly ones didn't'.

(24b) non-restrictive (qualificative) interpretation: 'Greeks were wise. They diluted wine with water'.

This opposition is only relevant for the attributive position (*the wise Greeks*). The predicative one (*The Greeks are wise*) only adds a characteristic without restricting the extension of the noun.

Which of these two properties is captured by means of restrictions (aoj>thing) and (mod<thing)? Preferential ability to be used in the attributive vs. predicative construction or preferential type of interpretation in the attributive construction? Even though these properties are correlated, they are quite different.

If UWs are to reflect the first opposition, it is not clear why we should wish to incorporate into UWs a syntactic difference between English words. Why should we treat this difference at the same level as the fundamental semantic difference between actions and states? This position is evidently untenable.

If we wish to capture the second opposition (which is much more reasonable), we should first of all take into account the distribution of adjectives between these classes. Some adjectives (such as *many*) can only be restrictive or are restrictive in the majority of cases:

(25) *Many dogs have curly hair.*

Some other adjectives (such as *damned* or *dear* – in the sense presented in (26b)) can only be non-restrictive:

(26a) *Get those damned dogs out of the room!*
(26b) *Dear colleagues!*

However, the overwhelming MAJORITY of adjectives can easily have BOTH interpretations. If we choose to convey this opposition by means of restrictions, we will have to split all these adjectives in two concepts, which is obviously rather strange. But this is not the most important shortcoming of this description. After all, it is technically possible to postulate two concepts for every adjective. The crucial fact is that the opposition restrictive/non-restrictive is not only relevant for adjectives, but also for other types of modifiers, such as relative clauses or prepositional phrases:

(27a) *The students(,) who are sitting in the corner(,) are waiting for the professor.*

(27b) *The students in the corner are waiting for the professor*.

The phrase *(who are sitting) in the corner* can be either restrictive (= 'those of the students who are sitting in the corner are waiting for the professor; others are not') or non-restrictive ('the students are waiting for the professor; they are sitting in the corner'). If we wish to mark this opposition for the adjectives, there is no reason not to do so for other types of modifiers.

Moreover, for such phrases it is even more important than for the adjectives, because in some languages restrictive and non-restrictive relative clauses have different punctuation and therefore should be differently treated by the deconverters. For example, in English and in Spanish restrictive relative clauses are not marked with commas, while non-restrictive necessarily are. Cf. synonymous English and Spanish sentences (28a) - (28b) and (29a) – (29b).

Restrictive:
(28a) *The old people who came a long way were tired.*
(28b) *Los viejos que habían venido de muy lejos estaban cansados.*
Non-restrictive:
(29a) *The old people, who came a long way, were tired.*
(29b) *Los viejos, que habían venido de muy lejos, estaban cansados.*

Thus relative clauses and other types of modifiers share with the adjectives the capacity to have restrictive and non-restrictive interpretations and should be treated in the same way. However in relative clauses and prepositional modifiers there is no UW to which a restriction can be assigned.

Therefore, I propose to renounce from the division of adjectives into (aoj>thing) and (mod<thing). In order to account for the opposition between restrictive and non-restrictive modifiers, two attributes are introduced (@restr, @non-restr) which can optionally be added to any modifier (an adjective, a prepositional phrase, a relative clause), if the UNL writer wishes to mark the restrictive or non-restrictive interpretation. As a general adjectival restriction, I would propose to introduce the one that is neutral to the restrictive/non-restrictive distinction, e.g. (icl>adj).

There are some more arguments to support the attribute solution:

(a) Attributes reflect the point of view of the speaker in the current situation and not the permanent property of the word. It is just the case with restrictive vs. non-restrictive interpretation of modifiers. It is the property of the given sentence and not the inherent property of the modifier. True, some of the adjectives cannot be used in one of these interpretations and for them this is a permanent property (see (25) – (26a,b)). But this does not in the least undermine the statement made above. Simply, these adjectives cannot be assigned one of the attributes @restr or @non-restr. It is the same with the nouns that have no plural form: they simply do not accept attribute @pl.

(b) The attribute is optional and need not be assigned if the UNL writer does not wish to specify his point of view. It is in fact not always easy to decide, whether or not a modifier is used restrictively. If we have two differently restricted UWs for an adjective, the UNL writer will always have to make a choice, very often irrelevant for the meaning he wishes to convey.

5 Relations

Currently, UNL disposes of 41 relations listed in the UNL Specifications. This set of relations has been tested in various encoding experiments and showed relative stability. However, the analysis of texts reveals three kinds of problems connected to UNL relations. First, some relations seem to be weakly differentiated and therefore difficult to use consistently. Second, the opposition between some relations seems to be based more on the semantic class of UWs than on the semantic relation that holds between them. Such distinctions should be avoided in a language designed for meaning representation. Third, some relations seem to be lacking. This topic deserves a special investigation that will be carried out later. Here I will only give several examples and, on a pilot basis, formulate some proposals.

5.1 Weakly differentiated relations

Example: gol (final state) – plt (final place); src (initial state) – plf (initial place).

According to the UNL manual (sec. 4.10) , examples like

(30) *John went to Brussels*

can be described both with gol and plt. The difference between the two is that gol characterizes Brussels as the final state of John, while plt – as the final place of the whole event "John went to Brussels". To put it mildly, it is difficult to understand what could be the final place of a movement as opposed to the final place of the moving object. The same applies to relations src and plf.

5.2 Distinction determined by the class of UWs

Example 1: mod (modification) – man (manner).

Both relations are very general and cover a wide range of situations which are not described by any specific relation, such as tim (time), plc (place), ins (instrument), etc. In practice, the difference between them boils down to the semantic class of the starting point of the relation: mod applies to things while man applies to situations.

(31a) *answered politely* (man)
(31b) *a polite answer* (mod)
(32a) *meet often* (man)
(32b) *frequent meetings* (mod)
(33a) *wrote in Japanese* (man)
(33b) *a letter in Japanese* (mod)

In my opinion, the difference between *to answer* and *an answer, to meet* and *a meeting*, or between *to write* and *a letter* has no bearing on the semantic relation in pairs (31a-b) – (33a-b). Relations man and mod can be safely merged into one relation. Any semantic difference between them, if it existed, is derivable from the context.

Example 2: plt (final place) – to (destination); plf (initial place) – frm (origin).

It is difficult to find any singnificant difference between relations in these pairs. They seem to differ only as to the type of the starting point of the relation: in case of plt it should be an event (action, process or state) while in case of to it should be a thing:

(34a) *The train is bound for Edinburgh* (plt).
(34b) *the train for Edinburgh* (to).

This difference does not seem to be fundamental enough to constitute different relations. The same is true for relations plf and frm.

Example 3: mod (modification) – agt (agent) / obj (object) / gol (final state) / …

According to the UNL Specifications, nominal UWs cannot be starting points for many argument relations, such as agt, obj, gol and some others. All these arguments are connected to nominal UWs by means of the mod relation. This approach is motivated by syntactic factors more than by semantic considerations. Phrases like (35a)

(35a) *arrival of the minister*

are described by means of the mod relation, obviously under the influence of the surface *of*-construction (cf. *decision of great importance*). Due to this, UNL fails to express the indentity of semantic relations in phrases like (35a) and (35b):

(35b) *The minister arrived*[8].

Besides that, UNL is unable to disambiguate phrases like (36)

(36) *invitation of the minister,*

which has at least two interpretations:

(36a) the minister invited (somebody) (agt)
(36b) (somebody) invited the minister (obj)

Obviously, a meaning representation language should have the means to establish identity of relations in (35a) and (35b), as well as to detect ambiguity in (36). This will be ensured, if argument relations like agt, obj, gol, etc. are allowed to go out of nominal UWs.

5.3 Missing relations

As is known, there exist no well-established criteria for deciding how many relations it is appropriate to have and what their semantic load should be. There is often a liberty of choice between introducing a relation for some specific meaning and expressing this meaning by other means. For example, how could we represent the difference between *after* and *before* in sentences (37) and (38), given that UNL has a relation for time (tim)?

(37) *He left after dinner.*
(38) *He left before dinner.*

[8] Moreover, the UNL Knowledge Base does not establish any link between semantic derivatives of the type *to arrive – arrival,* but this problem is beyond the scope of the present paper.

We have at least two options:

- use the existing relation tim and convey the difference between (37) and (38) by means of UWs after(icl>time) and before(icl>time):
  (37a) tim(leave(icl>do), after(icl>time))
  obj(after(icl>time), dinner(icl>event))
  (38a) tim(leave(icl>do), before(icl>time))
  obj(after(icl>time), dinner(icl>event))
- introduce special relations tim-after and tim-before and do without UWs after(icl>time) and before(icl>time):
  (37b) tim-after(leave(icl>do), dinner(icl>event))
  (38b) tim-before(leave(icl>do), dinner(icl>event))

These options are equivalent, although the first one is obviously preferable. It is better to keep the number of relations at the reasonable minimum, while the number of lexical units may be unlimited.

Nevertheless, there is a class of situations in which it might be more adequate to somewhat increase the number of relations. This is the area of relations between predicates and their arguments (cf. 3.2.2 above). The number of roles adopted by different authors for representing argument relations ranges from 14 in [14] to 57 in [15]. The list of relations in [15] proposed by Jury Apresjan is oriented towards the needs of deep semantic annotation of texts. For the UNL purposes it seems to be too detailed. However, some of the relations from this list deserve to be adopted in UNL. For example:

- cont (content): *he ordered us to attack; he proposed that; I think that…*
- top (topic) : *He knows nothing* (cont) *about women* (top); *review of the paper; the paper on UNL*
- rec (recipient): *He sent Mary flowers; He told me* (rec) *to come* (cont). *He informed us* (rec) *of his arrival* (cont)
- mot (motivation): *punish for disobedience, praise for achievements*
- asp (aspect): *differ in quality, distinguished for strength.*
  This is a topic for further discussion.

6 Hypernodes: their links and attributes

A UNLR is a hypergraph, i.e. a graph whose nodes are either simple or compound UWs (hypernodes, scopes). A compound UW is a subgraph consisting of simple or compound UWs linked with UNL relations. The major contribution of hypernodes in UNLR is their ability to bear relations and attributes of their own.

Each graph and subgraph (compound UW) contains a special node called the entry of the graph. Informally speaking, it represents the "main" element of the graph which normally corresponds to a syntactic top node of the corresponding part of the sentence. For example, the entry node of the phrase *music in Polynesia* is *music*, because it is this word that links the whole phrase to other words of the sentence. A phrase should be made a hypernode if its link to some element of the outer context is not semantically equivalent to the link of its entry node.

Situations where hypernodes are really necessary are rather rare. In the majority of sentences in which they are currently used, hypernodes are superfluous in the sense that their entry nodes effectively inherit their relevant properties. In other words, the replacement of a hypernode by a combination of simple nodes of which it consists does not result in any shift of meaning. Nevertheless, hypernodes are a useful and important formal device. In section 1 we saw one of the examples of its possible use for the representation of relatively fixed multi-word expressions. Below, I will show some situations when hypernodes are necessary as holders of relations and attributes.

6.1 Links of hypernodes

In the sentence

(39) *Music in Polynesia is extension of poetry*

there is no need to introduce a hypernode, because linking the phrase *music in Polynesia* to the verb is semantically equivalent to linking the noun *music* to this verb: ‘music in Polynesia is extension of poetry’ = ‘music is extension of poetry; this music is in Polynesia’. The same is true for the sentence

(40) *Music and dance are extensions of poetry.*

It can only be interpreted in the sense that both music and dance are extensions of poetry. Therefore, there is no need to merge *music and dance* in a hypernode. The situation is different in the sentence

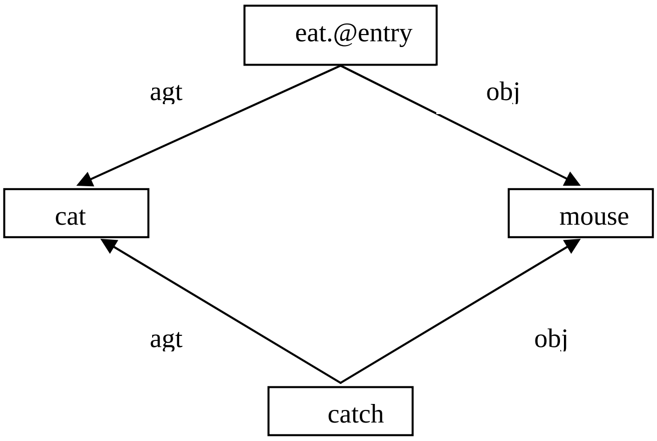
(41) *Music and dance in Polynesia are extensions of poetry.*

This sentence is ambiguous between two interpretations: ‘music in Polynesia and dance in Polynesia’ and ‘music (in general) and dance in Polynesia’. Under the first of these interpretations, the phrase *in Polynesia* is linked to the whole phrase *music and dance*, while under the second one it is only linked to *dance.* Thus, to assure proper understanding of the sentence, one has to introduce a hypernode *music and dance* for the first interpretation.

(41a) aoj(extension(icl>abstract thing.@entry, :01)
obj(extension(icl>abstract thing.@entry, poetry)
and:01(dance(icl>activity).@entry, music(icl>abstract thing))
plc(:01, Polynesia(icl>place))

As seen in (41a), UWs which constitute a hypernode do not have any links with UWs outside this hypernode. All the links which external UWs might have with UWs inside the hypernode are established with the hypernode itself. In (41a) these are links aoj and plc (*music and dance are extensions* and *music and dance in Polynesia*). An important question is whether it is possible that UWs inside the hypernode have direct links with UWs outside the hypernode. This question is raised by E. Blanc in [16]. Naturally, it is preferable to exclude this possibility. However, E. Blanc demonstrates a case where it is desirable, if not inevitable, to allow such a link. His example:

(42)



This UNLR can be verbalized in two different ways:

(42a) *The cat eats the mouse it caught.*
(42b) *The cat which caught the mouse eats it.*

Although these sentences describe the same situation, in a certain sense they are not equivalent, and this difference should not be lost in UNLR. E. Blanc proposes to express this difference by means of hypernodes. In (42a), a hypernode will look like obj:01(catch, mouse.@entry), while in (42b) it will be agt:01(catch, cat.@entry). Note that the entry nodes of these hypernodes are nouns and not verbs. The price paid for distinguishing (42a) and (42b) is admitting links going from the inside of the hypernode to the outside: agt(catch, cat) in the first case, and obj(catch, mouse) in the second case.

This proposal certainly solves the problem, but the price seems to be somewhat too high. The impermeability of hypernodes with respect to links from and to the outside is a property that is worth preserving as long as possible. To save this property, E. Blanc proposes to split one of the nodes in two identical and coreferential nodes. One of them would stay outside the hypernode, while the other would be included into it. This solution, however, implies a serious modification of the UNL specifications and the EnCo/DeCo software.

I would propose another solution which permits to distinguish interpretations (42a) and (42b) without violating the scope impermeability requirement and within the current specifications. What is the difference between (43) and (44)?

(43) *A girl is holding a peach*
(44) *a girl holding a peach*

These phrases describe the same situation but the meaning is organized in different ways. In (43) it is presented as an assertion, and in (44) as an object. How can we represent this difference in UNL? The first method is to declare *a girl* to be the entry node of (44). This inevitably leads to postulating a hypernode, since (44) may make part of a sentence where the predicate is also marked as entry (*I admired.@entry a* *girl holding a peach*). This will be exactly the type of a hypernode we saw in (42a) - (42b).

The second method of representing the difference between (43) and (44) is based on the fact that participles in the attributive position have dual nature. Participle *holding* in (44) conveys two different messages. The first message is purely semantic: a girl is the agent and a peach is the object of holding. The second message concerns the communicative organization of the meaning: the fact that a girl is holding a peach is presented as something that characterizes the girl, something that serves as a qualifier of the girl. In other words, *a girl* is the agent of *hold*, and at the same time *hold* is a modifier of *girl*. **This fact can be directly represented in UNL without recurring to hypernodes:**

(44a) agt(hold, girl)
mod(girl, hold)

(44a) makes explicit the dual nature of the attributive participle and thus effectively distinguishes (43) and (44). This approach can as well be applied to sentences (42a,b). In (42a), the link mod(mouse, catch) will be added, and in (42b) – mod(cat, catch).

6.2 Attributes of hypernodes

Example (41) shows that hypernodes may have links of their own which are not reducible to the links of their inner nodes. Now I will illustrate the situation where a hypernode has an attribute that cannot be assigned to any of its inner nodes.

The meanings that express the speaker's attitude towards the situation, such as 'not', 'can', 'must', etc. are expressed in UNL by means of attributes ascribed to a UW. For example, the meaning 'they do not sleep' is represented in UNL as aoj(sleep.@not, they). Consider sentences (45) and (46) which look similar but are opposed semantically:

(45) *They (do not sleep) because of the noise.*
(46) *They do not (quarrel because of money).*

(45) means 'noise is the cause of their not-sleeping', while (46) means 'money does not make them quarrel'. These readings differ in the scope of the negation, which we show by means of brackets. To express this difference in UNL, it is necessary to be able to attach the negation attribute to a hypernode:

(45a) rsn(sleep.@not.@entry, noise)
aoj(sleep.@not.@entry, they)
(46a) rsn:01(quarrel.@entry, money)
agt:01(quarrel.@entry, they)
:01.@not.@entry

Conclusion

I hope that interpretations and proposals presented here will be discussed by the participants of the UNL project, both at the UNL Workshop (Mexico, 2005) and at the forum for discussions. After that, two tasks seem to be the most important: revision of the UNL dictionaries according to the solutions taken during discussions and compilation of a corpus of UNL documents which incorporate all enconversion conventions which we will arrive at[9].

References

1. Boguslavsky, I. Guidelines for writing UNL expressions. FB2004: a showcase of UNL deployment. Technical document. Spanish Language Centre. Facultad de informática. UPM. Spain (2001)
2. Boguslavsky, I. Some remarks on the UNL encoding conventions. Proceedings of the First International UNL Open Conference “Building global knowledge”. SuZhou, China (2001).
3. Uchida H., Zhu M., Della Senta T. A Gift for Millenium. (1999) http://www.undl.org
4. Boguslavsky, I., Frid, N., Iomdin, L., Kreidlin L., Sagalova I., Sizov, V. “Creating a Universal Networking Lnguage Module within an Advanced NLP System”. Proceedings of the 18th International Conference on Computational Linguistics, 2000, p. 83-89.
5. Boguslavsky I., Cardeñosa J., Gallardo C., Iraola L. The UNL Initiative: an Overview. CICLING 2005 (in print).
6. Uchida, H. The UNL Specifications, v.3.2. (2003) http://www.undl.org
7. De la Calle Velasco G., Gallardo C., Cardeñosa J. Manual de instalación y uso del EditorUNL. Technical document. Spanish Language Centre. Facultad de informática. UPM. Spain (2003)
8. Boguslavsky, I., Iomdin, L, Sizov, V. Interactive enconversion by means of the ETAP-3 system. Proceedings of the International Conference on the Convergence of Knowledge, Culture, Language and Information Technologies, Convergences’03. Alexandria (2003)
9. Boguslavsky I. Some Lexical Issues of UNL. Proceedings of the First International Workshop on UNL, other interlinguas and their applications, Las Palmas, 2002, 19-22
10. Sérasset G., Mangeot-Lerebours M. Papillon Lexical Database Project: Monolingual Dictionaries & Interlingual Links. Proc. NLPRS'2001 The 6th Natural Language Processing Pacific Rim Symposium, Hitotsubashi Memorial Hall, National Center of Sciences, Tokyo, Japan, 27-30 november 2001, vol 1/1, (2001), pp. 119-125.
11. Uchida, H. The UW Specifications, v.2.0 (2002) http://www.undl.org
12. Jespersen O. Essentials of English Grammar. London: Allen & Unwin, 1933
13. Quirk R., Greenbaum S. A University Grammar of English. Longman, 1973, xi, 484 p.
14. Fry J., Bond F. Semantic annotation of a Japanese speech corpus, Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), 2000.
15. Apresjan Ju., Iomdin L., Sannikov A., Sizov V. Semantic Tagging in a Deeply Annotated Corpus of Russian. (in Russian, in print)
16. Blanc E., About and around the French enconverter and the French deconverter. In this volume.

[9] This paper is based on the research supported by the RFBR grant N 04-06-80148 and the Ramon y Cajal program of the Spanish Ministry of Science and Technology, whose assistance is gratefully acknowledged.

Some Lexical Issues of UNL

Igor Boguslavsky

Institute for Information Transmission Problems, Russian Academy of Sciences
19, Bolshoj Karetnyj, 101447, Moscow, Russia
bogus@iitp.ru

Abstract. The Universal Networking Language (UNL) developed by Dr. H. Uchida at the Institute for Advanced Studies of the United Nations University is a meaning representation language designed for multi-lingual communication in electronic networks, information retrieval, summarization and other applications. We discuss several features of this language relevant for correct meaning representation and multi-lingual generation and make some proposals aiming at increasing its efficiency.

1 UNL Approach to the Lexicon

The Universal Networking Language (UNL) developed by Dr. H. Uchida at the Institute for Advanced Studies of the United Nations University is a meaning representation language designed for multi-lingual communication in electronic networks, information retrieval, summarization and other applications.

Formally, a UNL expression is an oriented hypergraph that corresponds to a natural language sentence in the amount of information conveyed. The arcs of the graph are interpreted as semantic relations of the types agent, object, time, reason, etc. The nodes of the graph can be simple or compound. Simple nodes are special units, the so-called Universal Words (UWs) which denote a concept or a set of concepts. A compound node (hypernode) consists of several simple or compound nodes connected by semantic relations.

In addition to propositional content ("who did what to whom"), UNL expressions are intended to capture pragmatic information such as focus, reference, speaker's attitudes and intentions, speech acts, and other types of information. This information is rendered by means of attributes attached to the nodes.

After 6 years of the UNL project development, it is possible to take stock of what has been achieved and what remains to be done. In this presentation, I am going to concentrate on one of the central problems with which any artificial language is faced if it is designed to represent meaning across different natural languages. It is a problem of the language vocabulary.

I would like to single out three distinctive features of the UNL dictionary organization.

1. **Flexibility.** There is no fixed set of semantic units. There is only a basic semantic vocabulary that serves as a building material for free construction of derivative

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 101–108.

lexical units with the help of semantic restrictions. This makes it possible to balance to some extent the non-isomorphism of lexical meanings in different languages.

2. **Bottom-up approach.** The UNL dictionary consisting of Universal Words is not constructed a priori, top-down. Since it should contain lexical meanings specific to different languages, it grows in an inductive way. It receives contributions from all working languages. Due to this, one can expect that linguistic and cultural specificity of different languages will be represented more fully and more adequately than it would be possible under the top-down approach.
3. **Knowledge base.** As the UNL dictionary comprises unique semantic complexes lexicalized in different natural languages, we are facing the task of bridging the gap between them. It is supposed to be done by means of the Knowledge Base – a network of UNL lexical units connected by different semantic relations. Special navigation routines will be developed that will help to find the closest analogue to a lexical meaning not represented in the given language.

There are, however, some circumstances that impede full realization of these features, at least at the moment. Inductive storing of UWs from different languages is a good idea, but this process should be well organized. If a specific UW that is not self-evident is introduced to the UNL dictionary, it should necessarily be supplied at least by an informal comment to make it understandable to other users. Lucidity and easy interpretability of UWs is a goal at which all the developers of the UNL dictionary should aim.

Below, I am going to discuss in more detail two problems that have not so far received sufficient attention in UNL: the argument frames and lexical collocations.

2 Argument Frames

The need to introduce the information on the arguments does not seem to require justification. Any meaning representation language should have an ability to draw a distinction between the argument and non-argument links of predicates. In the UNL expressions, semantic links between the UWs are represented by means of UNL semantic relations. UNL disposes of an inventory of relations which, according to the latest specification, contains 41 items. Here are some examples of the UNL relations:

agt – agent (*John runs*),
obj – object (*read a book, A tree grows*),
ben – beneficiary (*He did not do anything for her*),
cag – co-agent (*I live with him*),
cob – co-object (*He fell into the river with the car*),
aoj – a thing which is in a certain state or is ascribed a property (*I love Mary; my brother is a student*).
dur – duration (*He worked nine hours*),
fmt – a range between two things (*He worked from Monday till Sunday*),
gol – final state (*turn red*),

ins – instrument (*observe with the telescope*),
met – method or means (*separate by cutting*),
pos – possession (*John's mother*),
rsn – reason (*They quarrel because of money*).

It is well known that for correct generation it is essential to know the argument structure of the predicates and the way each argument is expressed in the sentence. The UNL dictionary does not contain explicit information on the argument structure. According to the UW manual, the restrictions which should be included in the UW definitions are not meant for this purpose. As the UNL relations roughly correspond to semantic roles, it is supposed that each argument can be reliably identified based on its semantic role. However, this is not the case. Numerous attempts to construct a set of semantic relations, made over the last decades, showed that only a part of the relations between the words can be unambiguously interpreted in terms of semantic roles. In many cases this interpretation is largely arbitrary. This could not be a problem for the puproses of generation, if it were possible to assign semantic roles in a consistent way. Unfortunately, in practice it is hardly possible, especially when it is done by different people trained in different frameworks and working in different countries. The UNL texts compiled by the UNL project participants from 14 countries over the last years abound in mismatches in the representation of the same or very similar phenomena. Not surprisingly, most of them concern the representation of argument relations. For example, the phrase *base on respect* was interpreted by one team by means of the locative relation (lpl) and by another team by means of the comparative relation (bas), *freedom for all* was described with the purpose relation (pur) and with the beneficiary relation (ben), *bottleneck for the flow of information* received two labels – purpose (pur) and object (obj). Very often, the interpretation of a phrase in the corpus was motivated by the surface form rather than by its meaning. A typical example is *relations among nations* which was described by means of the locative relation obviously under the influence of the literal meaning of *among*. However, nations are by no means the place where relations occur. Rather, nations are participants of the "relations" situation and therefore are more likely to be objects (obj).

Sometimes the motivation behind the use of certain relations may be difficult to understand (at least, this is the case for the author of this paper). For example, in one of the sentences of the corpus, the argument structure of the verb *prevent* was presented as follows:

(1) *Nothing* (obj) *prevents members* (ben) *from discussing* (gol) *this problem.*

In our opinion, these problems are rooted not so much in the erroneous use of relations as in the fundamental impossibility of a consistent interpretation of all argument relations in terms of a small number of semantic roles.

What could one do to avoid the mismatches?

First, one could renounce using semantic roles in cases in which they are not obvious and replace them by semantically uninterpreted relations (subject, first object, second object, etc.). In this case, sentence (1) will receive a more transparent representation:

(2) *Nothing* (subject) *prevents members* (1 object) *from discussing* (2 object) *this problem.*

Obviously, it will be in many cases easier for those who write UNL expressions to develop a common approach to deciding which argument is the first object and which is the second than a common approach to finding appropriate semantic roles for them.

Second, one could accept the proposal of the French team and assign special markers to the case relations when they attach arguments (for example, @A would correspond to the first argument, @B – to the second, etc.). In this case, sentence (1) would be represented as:

(3) *Nothing* (obj.@A) *prevents members* (ben.@B) *from discussing* (gol.@C) *this problem.*

This would certainly reduce the area of uncertainty, but not eliminate it completely. To be able to interpret representation (3), the deconverter should know in advance the argument frame of the UW *prevent*. Otherwise, the uniformity of interpretation will still not be ensured. The only way to eradicate any ground for discordance between different users of the UNL language is to LIST ALL THE ARGUMENT STRUCTURES IN THE UNL DICTIONARY.

To incorporate this proposal, one need not introduce to the dictionary format any new possibilities: the existing apparatus of restrictions is quite sufficient. The only – but very serious – problem is to acknowledge that the argument frame should be explicitly and systematically specified in the UWs. If this is done, then one could keep using semantic roles in all the cases. For example, the word *bottleneck* (in the meaning of an obstacle) can receive the information that its syntactic object (*for something*) has the semantic role "pur" (or any other role which seems appropriate to the lexicographer). If every predicate is supplied with this information in the UNL dictionary, the discordance of opinion between different UNL users will become their private concern and the uniform treatment of the UNL relations in the most controversial zone – that of the argument relations – will be fully assured.

It should be emphasized however that in a general case the marking of the argument frame in a UW is not sufficient either. In some cases the same relation can attach to a UW both an argument and a free adjunct. For example, emotional states (of the type *be afraid, be surprised, be angry,* etc.) have an argument denoting a cause of the state. In sentence (4)

(4) *She is afraid to go out alone at night*

going out alone at night is what makes her to be in the state of fear. Therefore, relation "rsn" between *afraid* and *go out alone at night* is appropriate. On the other hand, *afraid* can have a non-argument cause, as in (5):

(5) *She is afraid (to go out alone at night), because this area is not very safe.*

Even if UW "afraid" is assigned a cause as one of the arguments (afraid(rsn>*)), we should know whether or not a "rsn"-link in the UNL expression denotes this argument. A good solution would be to mark the argument relation by a special label, as proposed in (3). Then, (5) will be represented as (6):

(6) rsn.@A(afraid(rsn>*), go out)
rsn(afraid(rsn>*), safe)

3 Lexical Collocations

Lexical collocations pose a serious problem for any language designed for representing meaning. Here are some examples of collocations from English: *give a lecture, come to an agreement, make an impression, set a record, inflict a wound; reject an appeal, lift a blockade, break a code, override a veto; strong tea, weak tea, warm regards, crushing defeat; deeply absorbed, strictly accurate, closely acquainted, sound asleep; affect deeply, anchor firmly, appreciate sincerely.* For simplicity, I will only dwell below on verbal collocations.

One of the problems such collocations raise is as follows. Some of the members of these collocations do not have a full-fledged meaning of their own. For example, the verb *give* in the collocation *give a lecture* does not denote any particular action. Its meaning, or rather its function, is the same as that of *take* in the collocation *take action*, or that of *make* in *make an impression*. The verbs *give, take* and *make* in these collocations are practically completely devoid of any meaning. Still, they have a very definite function – that of a support verb. This function is exactly the same in all the three cases, and nevertheless the verbs are by no means interchangeable. One cannot say **take an impression, *give action* or **make a lecture.* Moreover, this function is not only performed by different verbs with respect to different nouns. Very often, similar nouns in different languages require different verbs. For example, in Russian a lecture is not given but read, an action is not taken but accomplished, an impression is not made but executed.

How should these phenomena be treated in UNL? In particular, what UWs should be used for support verbs? The current practice suggests that UWs should be constructed on the basis of the source languages. Each language center should produce UWs for the words of its language, without any regard to other languages or any general considerations. A UNL expression and the UWs it consists of are considered adequate if they allow generating a satisfactory text in the same language they originated from. To what extent is this approach applicable to lexical collocations?

To answer this question, we will consider a concrete example. Suppose we have to convert to UNL Russian sentences with the meaning (7), (8), (9) or (10):

(7) *They began the war.*
(8) *We began the battle.*
(9) *The army suffered heavy losses.*
(10) *He took a shower.*

The problem is that in these contexts Russian uses quite different verbs than English. In Russian, correct sentences would be:

(7a) *They undid (razvjazali) the war.*
(8a) *We tied up (zavjazali) the battle.*
(9a) *The army carried (ponesla) heavy losses.*
(10a) *He received (prinjal) a shower.*

If UWs for support verbs in sentences (7a) – (10a) are constructed on the basis of Russian, they would look as follows: “undo(obj>war)”, “tie up(obj>battle)”, “carry(obj>loss)”, and “receive(obj>shower)”. These UWs will allow the Russian deconverter to produce perfect Russian sentences (7a) – (10a). In this case, the condition for adequacy mentioned above is met. Still, I would not consider UNL expressions based on these UWs adequate. They are produced without any regard for anything except the needs of Russian deconversion and are not fit for other purposes. In particular, these UWs are incomprehensible for anybody except Russians and it is doubtful that any other deconverter will be able to produce acceptable results from them. UWs originating from English will probably look like "take(obj>shower)", "begin(obj>thing)", "suffer(obj>loss)". To generate English sentences (7) – (10) from the UNL expressions constructed on the basis of (7a) – (10a), one would need to somehow ensure the equivalence of UWs "carry(obj>loss)" and "suffer(obj>loss)" in the Knowledge Base. This does not seem to be a natural and easy thing to do. Therefore, UWs for support verbs should not be constructed based on the lexical items of the source language.

Another possibility would be to make use of the co-occurrence properties of English lexical items. UNL vocabulary employs English words as labels for UWs and their meanings – as building blocks for UNL concepts which can be to a certain extent modified by means of restrictions. If lexical labels and meanings of UWs have been borrowed from English, their combinatorial properties can also be determined by the properties of corresponding English words. In this case, UWs and UNL expressions for sentences (7a) – (10a) will be identical to those for (7) – (10).

The advantage of this solution is obvious: since knowledge of English is indispensable for all the developers of X-to-UNL dictionaries, they can be sure that UWs for support verbs they produce are understandable and predictable. This solution has also drawbacks.

First, the inventories of support verbs in different languages are different. Therefore, we will often be faced with gaps in the lexical system of English and find no equivalent for a verb we need. Second, support verbs are bad candidates for the status of UWs. They do not denote any concept. Different support verbs often do not differ in meaning but only in their co-occurrence properties. It seems unreasonable to have different UWs to represent *take* (in *take action*), *make* (in *make an impression*) and *give* (in *give a lecture*), since the difference between these words is not semantic but only combinatorial. This difference should not be preserved in a meaning representation language.

The best solution would be to abstract from asemantic lexical peculiarities of support verbs and adopt a language-independent representation of these phenomena. Theoretical semantics and lexicography have long ago suggested a principled approach to the whole area of lexical collocations. It is the well-known theory of lexical functions by I. Mel'čuk implemented in the Explanatory combinatorial dictionaries of Russian and French (Mel'čuk 1974; Mel'čuk & Zholkovsky 1984; Mel'čuk *et al.* 1984, 1988, 1992, 1999). Possible use of lexical functions in NLP is discussed in (Apresjan *et al.* (in print)). Briefly, the idea of lexical functions is as follows. For more details, the reader is referred to the works mentioned above.

A prototypical lexical function (LF) is a general semantic relation R obtaining between the argument lexeme X (the keyword) and some other lexeme Y which is the value of R with regard to X (by a lexeme in this context we mean a word in one of its lexical meanings or some other lexical unit, such as a set expression). Sometimes Y is represented by a set of synonymous lexemes Y_1, Y_2, …, Y_n, all of them being the values of the given LF R with regard to X; e. g., MAGN (*desire*) = *strong / keen / intense / fervent / ardent / overwhelming.*

There are two types of LFs – paradigmatic (substitutes) and syntagmatic (collocates, or, in Mel'čuk's terms, parameters).

A substitute LF is a semantic relation R between X and Y such that Y may replace X in the given utterance without substantially changing its meaning, although some regular changes in the syntactic structure of the utterance may be required. Examples are such semantic relations as synonyms, antonyms, converse terms, various types of syntactic derivatives and the like.

A collocate LF is a semantic relation R between X and Y such that X and Y may form a syntactic collocation, with Y syntactically subordinating X or vice versa. R itself is a very general meaning which can be expressed by many different lexemes of the given language, the choice among them being determined not only by the nature of R, but also by the keyword with regard to which this general meaning is expressed. Typical examples of collocate LFs are such adjectival LFs as MAGN = 'a high degree of what is denoted by X', BON = 'good', VER = ‘such as should be’ and also support verbs of the OPER/FUNC family. Examples of the latter are OPER1 = ‘to do, experience or have that which is denoted by keyword X (a support verb which takes the first argument of X as its grammatical subject and X itself as the principal complement)’; OPER2 = ‘to undergo that which is denoted by keyword X (a support verb which takes the second argument of X as its grammatical subject and X itself as the principal complement)’; FUNC1 = ‘to originate from (a support verb which takes X as its grammatical subject and the first argument of X as the principal complement)’; FUNC2 = ‘to bear upon or concern (a support verb which takes X as its grammatical subject and the second argument of X as the principal complement)’.

If used in UNL, lexical functions will ensure a consistent, exhaustive and language-independent representation of support verbs and all other types of restricted lexical co-occurrence. For example, English and Russian support verbs we discussed above – *take* (*a decision, a shower*), *make* (*an impression*), *give* (*a lecture*), *suffer* (*losses*), *prinimat’* (*reshenie* ‘decision’, *dush* ‘shower’), *proizvodit’* (*vpechatlenie* ‘impression’), *chitat’* (*lekciju* ‘lecture’), *nesti* (*poteri* ‘losses’) – are correlates of the same lexical function – OPER1.

Being abstract and completely language-independent, lexical functions are devoid of all the drawbacks discussed above and can serve as an optimal solution to the problem of representation of the lexical collocations in UNL.

Acknowledgements This work has been supported by the Russian Foundation of Basic Research (grants Nos. 01-07-90405 and 04-06-80148).

References

Apresjan Ju., I. Boguslavsky, L. Iomdin, L. Tsinman (in print). Lexical function collocations in NLP.

Mel’čuk I. A., 1974. Opyt teorii lingvisticheskix modelej "Smysl – Tekst" [A Theory of Meaning – Text Linguistic Models"]. Moscow, Nauka, 314 p.

Mel’čuk I. A., Zholkovskij A.K., 1984. Tolkovo-kombinatornyj slovar' sovremennogo russkogo jazyka. [An Explanatory Combinatorial Dictionary of the Contemporary Russian Language] Wiener Slawistischer Almanach, Sonderband 14, 992 p.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Adèle Lessard, 1984. Dictionnaire explicatif et combinatoire du français contemporain, Recherches lexico-sémantiques I. Les Presses de l'Université de Montréal.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Louise Dagenais, Léo Elnitsky, Lidija Iordanskaja, Marie-Noëlle Lefebvre, Suzanne Mantha, 1988. Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques II. Les Presses de l'Université de Montréal.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Suzanne Mantha, 1992. *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques III.* Les Presses de l'Université de Montréal, 1992.

Mel’čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Suzanne Mantha et Alain Polguère, 1999. *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques IV.* Montréal.

The Representation of Complex Telic Predicates in Wordnets: the Case of Lexical-Conceptual Structure Deficitary Verbs

Palmira Marrafa

University of Lisbon, Faculdade de Letras de Lisboa – DLGR
Alameda da Universidade, 1600-214, Lisboa
Palmira.Marrafa@netcabo.pt

Abstract. This paper has a twofold aim: (i) to point out that telicity is both a lexical and a compositional semantic feature; (ii) to propose a straightforward solution to represent lexical telicity in wordnets-like computational lexica. The approach presented here subsumes the basic idea that lexicon is not a repository of idiosyncrasies. It is rather organized following a few general (universal or parametrical) constraints. In this context, despite the fact that the paper is mainly concerned with Portuguese, cross-linguistic generalizations can be captured, on the basis of a contrastive examination of data. The analysis focus on the behavior of complex telic predicates, in particular those which are deficitary with regard to their lexical-conceptual structure. In order to represent appropriately such predicates in wordnets, the specification of information regarding semantic restrictions, within the corresponding synsets, is proposed as well as a telic state relation.

1 Introduction

Telicity is mostly considered a compositional property of meaning. This paper attempts to make evident it is also a lexical feature and, as a consequence, it has to be represented in the lexicon. A concrete proposal to encode telic information of complex predicates in wordnets is provided.

This proposal emerges from the need of representing the predicates referred to in the Portuguese WordNet (WordNet.PT), which is being developed in the EuroWordNet framework.

From an empirical point of view, the work presented here mainly deals with complex telic predicates, in particular with those which involve lexical-conceptual structure (LCS) deficitary verbs, in the sense defined in previous work (cf. [4] and [5]).

The paper is divided in three main sections: the first one briefly describes the EuroWordNet model; the second one discusses the lexical-conceptual structure (in the sense of [7]) of complex predicates on the basis of a semantics of events, arguing for the lexical nature of telicity, and adduces evidences supporting the idea that some verbs define a deficitary lexical-conceptual structure; finally, the third main section presents an integrated proposal to encode LCS deficitary verbs and their troponyms in wordnets.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 109–116.

2 Wordnets: the EuroWordNet Framework

EuroWordNet is a multilingual database with individual wordnets for several European languages related by an Inter-Lingual-Index (ILI), as sketched in Figure 1.

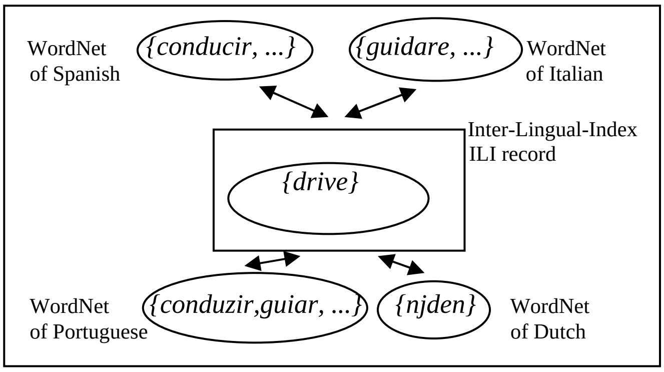


Fig. 1. EuroWordNet general architecture (adapted from [10])

Although initial conceived in the context of a European project, the EuroWordNet model is language independent. Therefore it is extendable to all languages of the world.

The individual wordnets are fundamentally structured along the basic lines of the Princeton WordNet ([1],[2]and[6]).

A wordnet is a conceptual-semantic network, in which the basic units are concepts, represented by sets of synonyms (synsets). A synset contains the set of lexicalizations for a given concept. For instance, the Portuguese expressions *bica, café expresso, cimbalino* are included in the same synset, since all of them are lexicalizations for the same concept (lexicalized in English by *espresso*).

The meaning of a lexical unit is derived from its relations with the other members of the same synset (lexical relations) and with other synsets (lexical-conceptual relations), as illustrated in Figure 2.

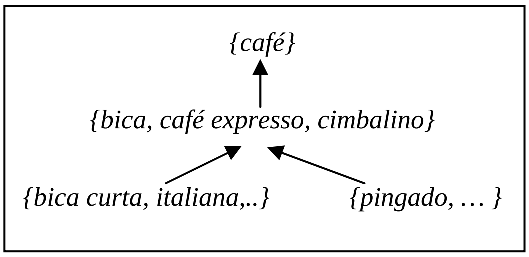


Fig. 2. Examples of relations in WordNet.PT

The meaning of *bica* is partially[1] derived from the synonymy relation with the other expressions inside the same synset and from the conceptual relations with the synset *{café}*, which represents a more general concept, and with the synsets *{bica curta, italiana,..}* and *{pingado,..}*, which represent more specific concepts.

Meaning emerges from the structure of the network. In a certain sense, it is constructed.

Though the conceptual-semantic relations are not the same for all lexical categories, as pointed out by Fellbaum [1], hierarchical relations are the major structuring relations. As well as nouns are mainly arranged by the hyperonymy/hyponymy relation, illustrated above, verbs are primarily organized by troponymy, a manner-of relation which also builds hierarchical structures, as exemplified below:

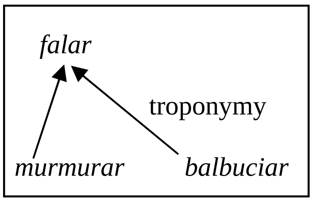


Fig. 3. Examples of troponymy relations

The verbs *murmurar* ("murmur") and *balbuciar* ("babble") are troponyms of *falar* ("talk"), specifying aspects related with volume and fluency of the talker, respectively.

The whole-part relation, or meronymy, is another major relation. For instance, *caffeine* is linked to *coffee* by meronymy.

A similar relation is specified for verbs, namely the sub-event relation. To give an example, *pay* is linked to *buy* by the sub-event relation.

The database also includes a set of relations which cover several aspects of semantic entailment. They are used to encode information on the participants typically involved in a given event.

3 Telic Complex Predicates

3.1 Lexical Conceptual Structure

This paper specifically deals with the so-called resultative constructions, like illustrated below:

(1) He painted the wall yellow.
(2) He washed the clothes white.

Both *yellow* and *white*, are resultative expressions. Sentence (1) entails that the wall became yellow as a result of painting. Similarly, sentence (2) entails that the clothes became white as a result of washing.

[1] Figure I does not describe exhaustively the relations specified for the synsets considered here.

Expressing the result of the event denoted by verb, the resultative expression integrates the predicate, as extensively discussed in [4] and [5]. In other words, the verb plus the resultative constitute a complex predicate.

As referred to by Stephen Wechsler (cf. [11]), "[i]f there is any aspect of resultatives that is completely uncontroversial, it is that they are telic: they describe events with a definite endpoint".

Despite this general assumption, there is a major controversy on whether or not the telic aspect of such constructions is an inherent feature of the meaning of the corresponding verbs.

The compositional hypothesis, defended by Verkuyl [9], has been argued for in recent works (see, for instance [3] and [8]) on the basis of contrasts like the following:

(3) a. John painted his house in one year / *for one year.
    b. John painted houses *in one year/for one year.

At a first glance, these examples suggest that (3)a. is telic and (3)b. is atelic and, consequently, that telicity depends on the nature of the internal argument, more precisely, on its quantifying system. Hence, telicity is a compositional feature of VP and not a lexical feature of V.

However, the relevant opposition seems to be transition *vs* process (or accomplishment *vs* activity, in other terms) and not telic *vs* atelic aspect.

As defended in [4], though the global event in (3)b. is a process, its main subevents are not atomic events, but transitions. Let us compare the structure of the global event of (3)a. and (3)b., represented by (4)a. and (4)b., respectively.

(4) a. $[_T [_P\ e_1 \ldots e_n]\ e_m]$
T, Transition; P, Process; e, atomic event
$e_m > e_n$
b. $[_P [_{T1} [_P\ e^1_1 \ldots e_n]\ e_{m1}] \ldots [_{Tt} [_P\ e^t_1 \ldots e_k]\ e_{m2}] \ldots]$
$e_{m1} > e_n, e_{m2} > e_k$

Similarly to $e_{m,}$ in a., e_{m1} and e_{m2}, in b., are telic states. This suggests that, although telicity is a compositional feature regarding the whole sentence, it is also an intrinsic feature of the verb.

By default, verbs like *paint* or *wash* are associated to the following LCS:

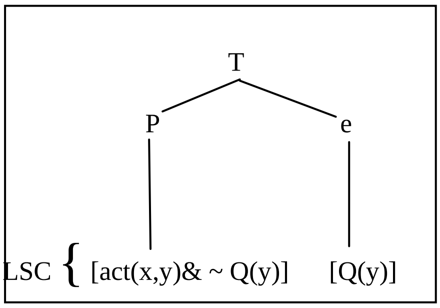


Fig. 4. Telic verbs LCS

Instantiating the variables with the data of sentences (1) and (2) we obtain (i) and (ii), respectively:

(i) [act(he,wall)& ~ painted_yellow(wall)],
[painted_yellow(wall)]
(ii) [act(he,clothes)& ~ washed_white(clothes)],
[washed_white(clothes)]

The absence of the resultative (*yellow* and *white* in these cases) does not have any impact on the LCS, as shown below:

(iii) [act(he,wall)& ~ painted(wall)],
[painted(wall)]
(iv) [act(he,clothes)& ~ washed(clothes)],
[washed(clothes)]

However, in the case of verbs like the Portuguese verb *tornar* ("make"), discussed below, it seems impossible to assign a value to *Q* independently of the resultative.

3.2 LCS Deficitary Predicates

Let us examine the following data:

(5) a. Ele tornou a Maria feliz.
"He made Mary happy"
b. [act(ele,Maria)& ~ feliz(Maria)],
[feliz(Maria)]
c. [act(ele,Maria)& ~ tornada_feliz(Maria)],
[tornada_feliz(Maria)]

The LCS of (5)a. seems to be (5)b. and not (5)c.. More concretely, *Q*, the telic state, is instantiated just with the resultative.

Additionally, the absence of the resultative induces ungrammaticallity, as shown:

(5)d. *Ele tornou a Maria.
"He made Mary"

Along the same lines of [4] and [5], verbs like *tornar* are defended here to be LCS deficitary, in the following sense:

Informal def.:

∀v((v a verb, ∃ ε, ε the LCS of v, ∃π, π the set of
content properties of ε, π=∅) => LCS_deficitary(v))

Since *π=∅*, the LCS can not bear an appropriate interpretation. In these circumstances, the ill-formedness of sentences like (5)d. is previewed. A syntactic structure that projects an anomalous LCS does not satisfy the requirement of full interpretation in Logical Form. Hence, it is ruled out.

In this particular case, the resultative not only expresses a lexical feature but it also fills the gap of the LCS of the verb.

These facts render evident that the representation of complex telic predicates in the lexicon, in particular the representation of those which are LCS deficitary, has to include information regarding the resultative, i.e, the telic expression.

4 Representation in WordNet

4.1 Synset Specifications

As referred to in the first section, concepts are represented in the network by synsets. Each synset includes the lexicalizations for a given concept. Therefore, synsets are supposed to include only lexicalized information.

As the analysis presented here has rendered evident, we have to extend synsets to another kind of information to represent the predicates at issue in an appropriate way.

It would not be adequate to overtly include in the synset all the expressions that can integrate the predicate, among other reasons, because they seem to constitute an open set.

A simple and plausibly solution is proposed below:

{tornar |SEM|REST <|TELOS|REST|<REL state|>||>||}

As observed, the telic expression of the predicate is represented by a feature structure description that partially specifies the semantic restrictions (SEM|REST) imposed by the verb.

The list of those restrictions includes the attribute TELOS, which stands for the entailed result, whose value includes the specification of a state, more precisely, a relation (REL) whose value is a state.

The pair *REL state* accounts for the fact that the state affects (expresses a relation with) an argument. We can even be more specific and include information to identify the concerned argument, but that is somewhat marginal to the main goals of this paper.

4.2 Telic State Relation

Verbs like *entristecer* (“sadden”) and *alegrar* (“make happy”) denote events that involve a change of state as well, but incorporate the expression that denotes the final state.

In order to capture the relation of the incorporated expression both with the corresponding verb and with the superordinate of that verb, a new relation – more precisely, the telic state relation – has to be included in the set of the internal relations of wordnets, since the existing sub-event relation is not specific enough to account for the facts discussed here. The sub-event relation stands for lexical entailment involving temporal proper inclusion. It has nothing to do with the geometry of the event.

On the contrary, the telic state relation regards the atomic sub-event (or state, in other words), which is the ending point of the global event and affects the theme.

In the case of verbs like *tornar*, that sub-event is implicit – but underspecified – in the meaning of the verb, as referred to above.

The troponyms of this class of predicates, on the other hand, incorporate the telic state, as (6) makes evident:

(6) a. Ele entristeceu a Maria.
"He saddened Mary"
b. *Ele entristeceu a Maria triste.
*He saddened Mary sad"

The representation proposed in Figure 4 accounts very straightforwardly for the facts discussed.

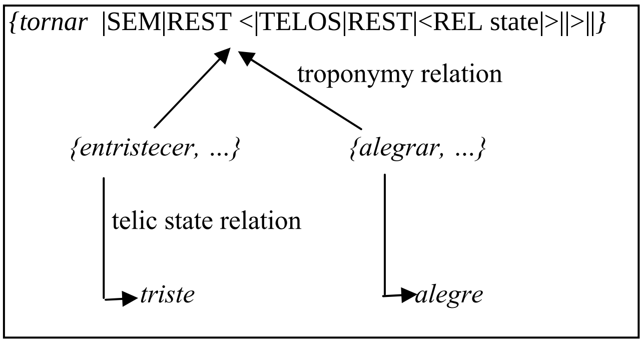


Fig. 5. Subnet for *tornar*

This representation both captures the troponymy relation with the semi-underspecified superordinate synset and relates the TELOS value of superordinate with the telic state incorporated of the troponym.

5 Conclusion

The proposal presented in this paper has a strong empirical motivation and enhance the expressive power of wordnets.

Feature structures are high flexible modelling structures and allow for the specification of information, be it semantic or syntactic, in a very principled way.

The new relation proposed allows for a more integrate and fine grained representation of the facts at issue.

Enriching wordnets in the sense proposed here will open new possibilities for the application of this powerful basic resource in the wide and challenging domain of language and information technologies.

Acknowledgment I thank Instituto Camões for having funded the WordNet.PT project first phase and Fundação Calouste Gulbenkian for funding current basic research which permits to continue to develop this project.

References

1. Fellbaum, C., "A Semantic Network of English: The mother of All WordNets", in P. Vossen (ed.) *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*, Dordrecht: Kluwer Academic Publishers, 1998a, 209-220.
2. Fellbaum (ed.) *WordNet: An Electronic Database*, MA: The MIT Press, 1998b.
3. Krifka, M., "Thematic Relations as Links between Nominal Reference and Temporal Constitution", in I. Sag and A. Szabolcsi (eds.), *Lexical Matters*, Stanford, CSLI, 1992.
4. Marrafa, P., *Predicação Secundária e Predicados Complexos – Modelização e Análise*, PhD. dissertation, Lisbon, University of Lisbon, 1993.
5. Marrafa, P., "Predicados Télicos Descontínuos: análise e computação", in P. Marrafa e A. Mota (orgs.), *Linguística Computacional: investigação fundamental e aplicações*, Lisboa, APL/Edições Colibri, 1999, 173-189.
6. Miller G., R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller, "Introduction to WordNet: An On-line Lexical Database", International Journal Of Lexicography, 3(4), 1990, pp. 235-244.
7. Pustejovsky, J., "The Syntax of Event Structure", *Cognition*, 41, 1991, 47-81.
8. Schmitt, C., Aspect of the Syntax of Noun Phrases, University of Maryland, PhD. Dissertation, 1996.
9. Verkuyl, H., On the Compositional Nature of Aspects", Dordrecht, D. Reidel, 1972.
10. Vossen, P., *EuroWordNet: General Document*, University of Amsterdam, 1999.
11. Wechsler, S., "An Analysis of English Resultatives Under the Event-Argument Homomorphism Model of Telicity", Proceedings of the 3rd Workshop on Text Structure, University of Texas, 2001.

Remaining Issues that Could Prevent UNL to be Accepted as a Standard

Gilles Sérasset and Étienne Blanc

GETA-CLIPS, IMAG, Université Joseph Fourier,
BP 53, 38041 Grenoble cedex 9,
Gilles.Serasset@imag.fr

Abstract. This paper presents practical issues when dealing with UNL (Universal Networking Language) documents. Some of these issues are at a purelly syntactical level, others are at a semantic level. Some of these issues introduce unnecessary difficulties when developing tools to handle UNL documents when others introduce unnecessary difficulties when encoding natural language utterances into UNL graphs.

1 Introduction

After several years of development, UNL (Universal Networking Language, [1, 2]) has proved its viability as a cross lingual data exchange format. Its expressive power makes it very useful for the development of multilingual information systems where it serves as a way to represent utterances in a language free manner. However, in order to be adopted as a standard, the UNL definition should be clarified or corrected in order to avoid common errors and misunderstandings.

As a UNL partner since 1998, the GETA (Groupe d'Étude pour la Traduction Automatique) group of the CLIPS (Communication Langagière et Interaction Personne-Système) lab develops and maintains a UNL deconverter for French. For this development, we are one of the few groups that decided to use our own existing tools (namely the ARIANE-G5 translator generator, [3–6]). As such, we had to develop several tools to parse and handle UNL documents and went accross some of the problems that will arise when UNL will be used by third party developers.

This paper presents some of the issues we faced and suggests some solutions. Our goal is to give UNL the opportunity to be largely adopted by third parties as a de-facto standard. After briefly presenting the UNL language and an example of an UNL document, we will begin by low level problems posed by the UNL syntax. After that, we will focus on middle level aspects involved when interpreting the UNL language at its computational level. Finally, we will present some of the higher level issues arising when we interpret UNL utterances as linguistic structures.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 117–124.

2 Motivations

The UNL (Universal Networking Language) provides a language independent knowledge representation formalism. Such a formalism allows for the development of Information Systems where all computation may be performed on UNL expressions and where natural language is only considered as a interface medium for humans. In this vision, enconversion and deconversion may be considered as the interface layer of such a Information System.

Such an information infrastructure is only possible if the UNL expressions are represented and interpreted in a coherent way. As such, the UNL has to be accepted as a standard by the developers of such Information Systems and by the developers of enconverters and deconverters.

However, even between the persons in charge of deconvertion and enconversion, we can see some discrepancies in the way natural language utterances may be encoded in UNL expressions are in the way UNL expressions may be interpreted.

Some of these discrepancies are coming from errors that are not detected by current tools and shows that the UNL infrastructure is still lacking a practical validation routine. Others are coming from the UNL specifications themselves. This papers focuses on the latter case, where the UNL specification should be corrected or clarified. We will also present some unnecessary difficulties we faced when developing our own French deconverter using standard java tools. We claim that such difficulties will slow the adoption of UNL as a standard by independant Information System developers.

3 Issues in UNL syntax

3.1 Overview of UNL

The purpose of this section is to briefly present the syntax of UNL documents. For more details, refer to [2].

UNL documents A UNL document is a set of UNL utterances, structured by proprietary tags ([D] for documents, [P] for paragraphs, [T] for titles and [S] for sentences). Each sentence in this structure contains a UNL utterance.

UNL utterances Hence, UNL utterances are interpreted as graphs where the nodes are annotated “Universal Words (UW)” and arcs are labeled by a relation.

Such graphs are to be denoted using a specific syntax that (usually) represents the list of arcs of the graphs.

Example of a UNL document The following one sentence document

```
[D: dn=sample document, on=French]
[S:1]
```

```
{org:fr}
Le chat attrape une souris.
{/org}
{unl}
agt(catch(agt>thing,obj>thing).@entry,   cat(icl>feline).@def)
obj(catch(agt>thing,obj>thing).@entry,   mouse(icl>rat).@indef)
{/unl}
[/S]
[/D]
```

shows the UNL representation of a French document containing the single sentence “Le chat attrape une souris” (“The cat catches a mouse”). Fig. 1 shows the graphical representation of the corresponding graph (where attribute `.@def` and `.@indef` have been hidden).

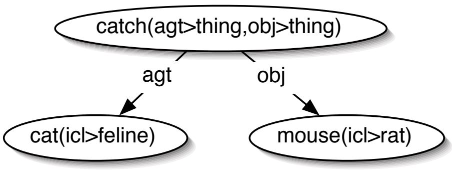


Fig. 1. UNL graph for the example sentence

3.2 Issues in UNL syntax

The current UNL syntax has several issues that leads to unnecessary complexity when parsing an UNL document.

Lexical lookahead is necessary In a UNL graph, the nodes are annotated UWs. For example, the node
`cat(icl>feline).@def`
consists in the UW `cat(icl>feline)` annotated by the `.@indef` attribute.

A UW consists in a headword (`cat` in our example) representing an English language word followed by an optional list of constraints (`(icl>feline)` in our example) that is sufficient to distinguish the correct sense of the UW among the word senses of the headword.

Some headword may contain the “`.`” character, as, `etc.` or `P.O. Box`. As some UWs with these headwords may not be constrained, a UNL graph may contain a node like `etc..@parenthesis` where the first “`.`” character is part of the headword and the second one introduces an attribute.

In order to correctly parse such graphs, one has to distinguish between both usages of the “`.`” character. This distinction may only be done with a 1 character lookahead in the parser, either at the lexical, or at the syntactic level. Doing this at the syntactic level makes the grammar significantly more complex and some of the standard tools that may be used don't properly handle such lookahead.

Encoding issues As most lexical items of the UNL are expressed using lower ASCII characters, most of the document may be parsed provided that the encoding used for the document is compatible with lower ASCII. This forbids the use of UTF-16 or EBCDIC encodings.

Free natural language occurrences may occur in a UNL utterance. Notably between the `{org}...{/org}` tags. Encoding of these parts may be indicated via the `{org:<l-tag>=<charcode>}` tag. However, each language may use it's, own encoding. Hence, when parsing a UNL document, a developer cannot treat a UNL document as a standard stream of characters but rather as a stream of bytes, as the byte to character transformation may vary within the document.

Moreover, there is no way to specify the encoding used for a UNL graph, but knowing this encoding is necessary. The reason is that headwords may contain characters that are not in the lower ASCII character set. We can find such UWs in the UNL specification document itself, as `soufflé(icl>food)`. Hence, to ensure the correct handling of a UNL document, the encoding should be made explicit in the graph opening tag (`{unl:...}`).

4 Computational Interpretation of UNL graphs

4.1 Use of Hyper-nodes

The introduction of UNL specification states that "*the UNL expresses information or knowledge in the form of semantic network with hyper-node*". Hence UNL is dealing with the computational model of hyper-graphs.

In the standard form of the UNL syntax, hyper-nodes are represented by a hyper-node ID, used to label relations appearing inside this node. For example, the English sentence "*The cat who caught the mouse eats*" will be encoded by the UNL graph:

```
agt:01(catch(icl>do).@present,  cat(icl>animal).@def.@entry)
obj:01(catch(icl>do).@present,  mouse(icl>animal))
agt(eat(icl>do).@present.@entry, :01)
```

which is interpreted as in Fig. 2.

4.2 Cross-scope relations

Example of UNL graphs using hyper-nodes, aka scopes, are numerous and the UNL syntax also allows the encoding of relations linking nodes across different scopes. For example, the English sentence "*The cat who caught the mouse eats it*" may be encoded by the the UNL graph:

```
agt:01(catch(icl>do).@present,  cat(icl>animal).@def.@entry)
obj:01(catch(icl>do).@present,  mouse(icl>animal))
agt(eat(icl>do).@present.@entry, :01)
```

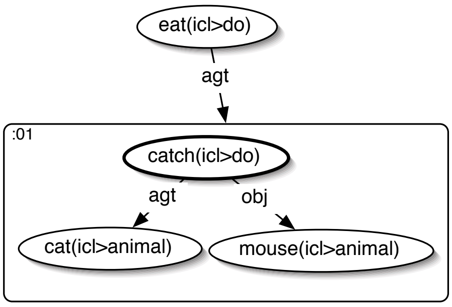


Fig. 2. Sample UNL hyper-graph

`obj(eat(icl>do).@present.@entry, mouse(icl>animal))`
which is interpreted as in Fig. 3.

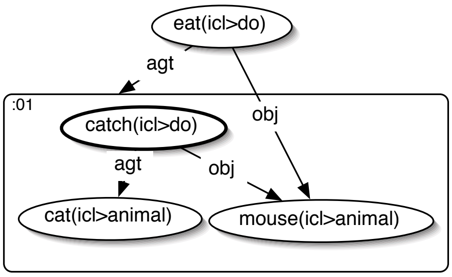


Fig. 3. A UNL graph with a cross-scope relation

One will note that the supplementary `obj` relation links a node (`eat`) at the top level to a node (`mouse`) which is inside the scope `:01`.

Hence, we conclude that the UNL syntax may be used to define classical graphs with Hyper-nodes, and may also be used to define cross-scope relations. However, this conclusion seems not to be shared by all UNL partners and such a possibility should be either explicitly forbidden, or illustrated, with a particular example in the UNL specification.

5 Linguistic Interpretation of UNL graphs

5.1 Scope of the "@not" attribute

The UNL specification chose to encode the speaker's view of aspect using particular attributes. As an example, the English sentence "*the car is about to work*"

will be encoded as
`agt(work(agt>thing).@entry.@begin.@soon,   car(icl>automobile))`

The negation is also expressed as an attribute. For example, the English sentence "*the car does not work*" will be encoded as
`agt(work(agt>thing).@entry.@not,   car(icl>automobile))`

But, the linguistic interpretation of the UNL graph
`agt(work(agt>thing).@entry.@begin.@soon.@not, car(icl>automobile))`
is not clear and may be:

- *the car is about not to work*
- *the car is not about to work*

The scope of the negation should be more clearly stated.

5.2 Encoding of predicates/arguments

One of the greatest difficulty when enconverting or deconverting a UNL graph is that relations are not always easy to select. This difficulty is at its highest when working with some predicate verbs or nouns. The reason is that relations between a predicates and its arguments are generally defined with respect to the predicate whereas UNL relations are defined independently of the UWs they connect.

As an example, in the English sentence "*I strive to work*", there is no real need to characterize the relation between "strive" and "work" as it is simply defined as the second argument of "strive". However in order to encode this sentence in UNL, one has to select, among UNL relation, the one that may represent this particular relation.

In this particular example, users usually hesitate between 2 relations:

- `obj` when the encoder considers that the action of working is directly affected by the action of strive or
- `pur` when the encoder considers that the agent of strive performs some actions which purposes are "to work".

Frequently, the final choice is adopted as a convention. As such, the chosen convention should be documented somehow. Hence, if the partners chose to encoded this relation as an `obj`, this choice should be reflected in the way the UW for "strive" is encoded, as in `strive(agt>human,obj>action)`.

If the other choice is made (which we personally prefer), another problem occurs, as the `pur` relation will be ambiguous in the context of "strive" as it will represent the second argument of the predicate and it may also be used as a circumstantial.

As an example, let's encode the English sentence "*John strives to work to survive*". First, let's get rid of the solution consisting to attach "survive" as the purpose of "work" as in

```
agt(strive(agt>person,pur>action).@entry, John)
pur(strive(agt>person,pur>action).@entry, work(agt>person))
pur(work(agt>person), survive(agt>person))
```

illustrated in fig. 4

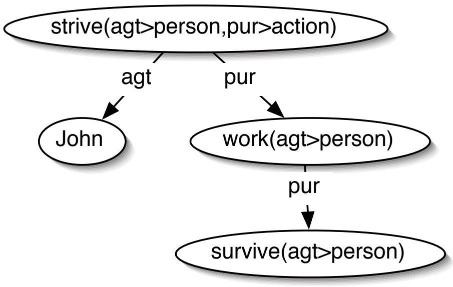


Fig. 4. Incorrect solution to encode "*John strives to work to survive*"

This solution is incorrect as it means that "work to survive" is the purpose of John's actions , as if, most of the time he was working, but not to survive.

Hence, "survive" has to be considered as the purpose of John when he strives to do anything.

Which means that the correct graph should be

```
agt(strive(agt>person,pur>action).@entry, John)
pur(strive(agt>person,pur>action).@entry, work(agt>person))
pur(strive(agt>person,pur>action).@entry, survive(agt>person))
```

as illustrated in fig. 5

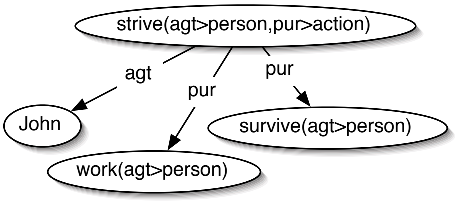


Fig. 5. Correct but problematic solution to encode "*John strives to work to survive*"

But this solution is problematic as no deconverter may decide between "*John strives to survive to work*" and "*John strives to work to survive*".

Hence, the UNL specification should provide a way to distinguish in a graph

- relations that links a predicate and its argument and
- relations that links predicates to circumstantials.

5.3 Encoding variables or untranslatable entities

According to the UNL specification, all UWs that are used in UNL graphs need to be defined in the UNL Knowledge Base (KB). However, several sentences contains elements that do not represent any concept and are not translatable. As an example, the English sentence "*Computers with part number 'ABCD' should be returned to the factory*" will be encoded by a graph containing a node labeled `ABCD` which has no reason to appear in any knowledge base.

In some cases, such untranslatable entities may even be ambiguous with legal UWs.

As there is no way to syntactically distinguish between a legal UW and an untranslatable entity, correct deconversion of such graphs is impossible.

6 Conclusion

As we have shown in this paper, several issues, of varying importance, are still to be addressed as the main ambition of UNL is to become a standard for representing knowledge and information in a language independent way. As such, any point that may be subject to user's interpretation is counter productive.

Also, the initial ambition of UNL is to represent information in an unambiguous way. Such an ambition is fully justified when the information is available without any ambiguity. However, much information systems that may use UNL will have to deal with ambiguous informations. Hence, UNL should define a way to encode ambiguous information.

References

1. Uchida, H., Zhu, M., Della Senta, T.: UNL: A Gift for a Millennium. Institute of Advanced Studies, The United Nations University, Tokyo, Japan (2000)
2. UNL Center: The Universal Networking Language (UNL) specifications Version 3 Edition 2. UNDL Fondation (2003)
3. Boitet, C.: Software and lingware engineering in recent (1980-90) classical mt : Ariane-g5 and bv/aero/f-e. In: ROCLing-III (tutorials). (1990) 21
4. Boitet, C.: Handling texts and corpuses in ariane-g5, a complete environment for multilingual mt. In Belguith, L., ed.: ACIDCA'2000, Corpora and Natural Language Processing, Monastir, Université de Sfax (2000) 7–11
5. Sérasset, G., Boitet, C.: Unl-french deconversion as transfer and generation from an interlingua with possible quality enhancement through offline human interaction. In Tsujii, J.I., ed.: MT Summit VII, Singapore, Asia Pacific Ass. for MT (1999) 220–228
6. Sérasset, G., Boitet, C.: On unl as the future "html of the linguistic content" and the reuse of existing nlp components in unl-related applications with the example of a unl-french deconverter. In Uszkoreit, H., ed.: COLING-2000, Saarbrücken, ACL (2000) 768–774

Semantic Analysis through Ant Algorithms, Conceptual Vectors and Fuzzy UNL Graphs

Mathieu Lafourcade

LIRMM, Université Montpellier II, 161, rue Ada,
34392 Montpellier cedex 5
mathieu.lafourcade@lirmm.fr

Abstract. In the context on the UNL project, we focus on the automatization of enconversion process, that is the building of UNL graphs from sentences. We present an extension of the UNL graph structure aiming at handling lexical and relational ambiguities. On this intermediate structure, we can apply ant algorithm propagation of conceptual vectors and other constraints. Graph nodes and relations have a level of excitement and when this level remains too low for too long they are deleted. This way, both acception and attachment selections can be performed.

1 Introduction

In itself, a text constitutes a complex system, but the computational problem is that the meanings are not strictly speaking active elements. In order to ensure the dynamicity of such a system, an active framework made of "meaning transporters" must be supplied to the text. These "transporters" are intended to allow the interactions between text elements and they have to be both light (because of their possible large number) and independent (word meanings are intrinsic values). Moreover, when some meanings stemmed from different words are compatible (*engaged* with *job* for instance), the system has to keep a trace of this fact. These considerations led us to adopt ant algorithms. Ant algorithms or variants of them have been classically used for optimisation problems like traveling salesman problem [*Dorigo* et al. 1997] among many others, but they were never used in Natural Language Processing (most probably because the NLP community contrary to the psycho-linguistics one, considered semantic aspects not very often as an optimization problem, nor explicitely modeled then as a dynamic complex system, [*Kawamoto* 1993] being a notable exception). However, [*Hofstadter* 1995] with the COPYCAT project, presented an approach where the environment by itself contributed to solution computation and is modified by an agent population where roles and motivations vary. Some properties of these models seem to be adequate for the task of semantic analysis, where word senses can be seen as more or less cooperating. We retain here some aspects that we consider as being crucial: (1) mutual information or semantic proximity is one key factor for lexical activation, (2) the syntactic structure of the text can

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 125–137.

be used to guide information propagation through possibly ambiguous relations. Finally, as pointed by [*Hofstadter* 1995], biased randomization (which doesn't mean chaos) plays a major role in this kind of model.

In the context on the UNL project, we focus on the automatization of enconversion process, that is the building of UNL graphs from sentences. We present an extension of the UNL graph structure, dubbed *fuzzy UNL graph*, aiming at handling lexical and relational ambiguities. On this intermediate structure, we can apply ant algorithm for propagating conceptual vectors and other constraints. Graph nodes and relations have a level of excitement and are deleted when this level remains too low for too long. This way, both acception and attachment selections can be performed. We construct fuzzy graphs on the basic of morpho-syntactic analysis trees which enumerate PP (prepositional phrase) attachments or are duplicated depending on the nature of syntactical ambiguities. Lexical ambiguities are represented as alternative nodes at leaf level.

The conceptual vector model is a *recall focused* approach which aims at representing thematic activations for chunks of text, lexical entries, locutions, up to whole documents. Roughly speaking, vectors are supposed to encode *ideas* associated to words or expressions. The main applications of the model are thematic text analysis and lexical disambiguation [*Lafourcade* 2001] and can find interesting approaches for vector refinement through the lexical implementation of taxonomies like the UNL knowledge base. Practically, we have built a system, with automated learning capabilities, based on conceptual vectors and exploiting monolingual dictionaries for iteratively building and refining them. For French, the system learned so far 165000 lexical entries corresponding to roughly 560000 vectors (the average meaning number being 5 for polysemous terms). We are conducting the same experiment for English.

In this paper, we first expose the main principles and assumptions about the treatment of ambiguities during the enconversion. Then, we present the conceptual vectors model and the fuzzy graph extension. The conceptual vector propagation through ant algorithm is then detailed with its consequences on weighting acception and relations. Some examples of fuzzy graphs are given, focusing mainly on simple acception selection and choice between *mod* (modifier) and *ins* (instrument) relations.

2 Holistic Algorithms for Disambiguation

Thematic representation and mutual information sharing The constraints present in the UNL knowledge base is instrumental for an automated enconversion process but is by far too scarce to be considered as a thematic (or semantic) representation. We use conceptual vectors to convey a rich meaning representation both for acceptions and for each entry of the knowledge base.

Analysis viewed as a Non-Ending Iterated Process Very often, the semantic analysis is viewed as processing sequentially more or less like an expert system. In our views, this process should be done incrementally by adding little pieces of informations (dubbed as *clues*) at a time, and letting some induction process structuring the result. The process may converge (it is the case most of the time), but for very ambiguous results some oscillations could occur. Furthermore, all kinds of semantic ambiguities are holistically processed, that is at the same time, with all representation clues being solicited.

Explicitly Managing Uncertainty More than often, uncertainty about domain or about data interpretation are considered as problems to be absolutely solved, and in case of irreconcilable constrains, some heuristics are called or experts questionned. We think that uncertainty should be explicitly represented and managed, as it can never be completely eliminated. This is why, we advocate that each relation in the graph to be associated with a *confidence value* or (depending on the view adopted) *excitement level.* This value may be increased (or lowered) according to the clues discovered or the induction undertaken. Distributional aspects of free texts are an excellent source for managing uncertainty on the basis of existing items and relations found in dictionaries.

Mixing Meanings and Vocables Lexical and syntactical ambiguities are the issues at stake. More than often in texts, word senses may not be clearly separated. Morevover, it is now well accepted in psycho-linguistics that language is processed at the same time at vocable (terms, compounds, etc.) and meaning (thematically and associatively) levels.

3 Conceptual Vectors and Fuzzy UNL Graphs

3.1 Conceptual Vectors

The Model of Conceptual Vector has already been presented the context of UNL in [*Lafourcade* et al. 2002] and what follows is a short description (towards the unfamiliar reader) of the main principles. Thematic aspects (or ideas) of textual segments (documents, paragraphs, syntagms, etc.) are represented thanks to vectors of interdependent concepts. Lexicalized vectors have been used in information retrieval for long [*Salton* et al. 1983] and for meaning representation by the LSI (Latent Semantic Indexing) model from latent semantic analysis (LSA) studies in psycho-linguistics [*Deerwester* et al. 90]. In computational linguistics, [*Chauché* 90] proposed a formalism for the projection of the linguistic notion of semantic field in a vectorial space, from which our model is inspired [*Lafourcade* 2001]. From a set of elementary notions, dubbed as *concepts*, it is possible to build vectors (conceptual vectors) and to associate them to lexical items. The hypothesis that considers a set of concepts as a generator to language has been long described in [*Roget* 1852] (*thesaurus hypothesis*). Polysemous words combine the different vectors corresponding to the different meanings considering several criteria as weights: semantic context, usage frequency, language level, etc. Concepts are defined from a thesaurus (in our prototype applied to French, we have chosen [*Larousse* 1992] where 873 concepts are identified to compare with the thousand defined in [*Roget* 1852]). To be consistent with the thesaurus hypothesis, we consider that this set constitutes a generator space for the words and their meanings. This space is probably not free (no proper vectorial base) and as such, any word would project its meaning(s) on this space.

Thematic Projection Principle and Angular Distance. Let be $\mathcal{C}$ a finite set of n concepts, a conceptual vector V is a linear combination of elements c_i of $\mathcal{C}$. For a meaning A, a vector $V(A)$ is the description (in extension) of activations of all concepts of $\mathcal{C}$.

Let us define $Sim(A, B)$ as one of the *similarity* measures between two vectors A et B, often used in information retrieval as their normed scalar product. We suppose here that vector components are positive or null. Then, we define an *angular distance* D_A between two vectors A and B as their angle.

$$\begin{aligned} Sim(A,B) = \cos(\widehat{A,B}) &= \frac{A \cdot B}{\|A\| \times \|B\|} \\ D_A(A,B) &= \arccos(Sim(A,B)) \end{aligned} \tag{1}$$

Intuitively, this function constitutes an evaluation of the *thematic proximity* and is the measure of the angle between the two vectors. We would generally consider that, for a distance $D_A(A, B) \leq \frac{\pi}{4}$, (i.e. less than 45 degrees) A and B are thematically close and share many concepts. For $D_A(A, B) \geq \frac{\pi}{4}$, the thematic proximity between A and B would be considered as loose. Around $\frac{\pi}{2}$, they have no relation. D_A is a real distance function. It verifies the properties of reflexivity, symmetry and triangular inequality. We can have, for example, the following angles (values are in degrees; examples are extracted from `http://www.lirmm.fr/~lafourcade`):

$$\begin{array}{lcl} D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{tit}\text{›}) & = & 0^\circ \\ D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{animal}\text{›}) & = & 32^\circ \\ D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{passerine}\text{›}) & = & 10^\circ \\ D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{joy}\text{›}) & = & 42^\circ \\ D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{bird}\text{›}) & = & 19^\circ \\ D_A(\text{‹}\textit{tit}\text{›}, \text{‹}\textit{sadness}\text{›}) & = & 65^\circ \end{array}$$

A ‹*tit*› is thematically closer to a ‹*passerine*› than a ‹*bird*› than an ‹*animal*›. Here the thematic proximity follows some kind of ontologic relation. However, ‹*cell*› nonewithstanding the polysemy begins to be poorly related. The term ‹*sadness*› has almost no thematic sharing with ‹*tit*›.

Meaning Selection. From a given thematic context under the form of a conceptual vector, it is possible to select (or weight) the meanings of a vocable. For a vocable w with k meanings $w_1 \ldots w_k$ and a context C, the weights α of the meanings are non-linearly related to the amount of mutual information between the context and a given meaning:

$$\alpha_i = \cot(D_A(V(w_i), C) = \frac{\cos(D_A(V(w_i), C)}{\sin(D_A(V(w_i), C)} \tag{2}$$

We recall that *cot* refers to the *cotangent* function, with $\cot(0) = +\infty$ and $\cot(\pi/2) = 0$. The rational is the following. The *similarity* between two objects A and B is the cosine of the angle between these two objects. Inversely the *dissimilarity* is the sine. The weight of selection of B towards A if the ratio between what is common (the similarity) on what is different (the dissimilarity).

For example, take the vocable ‹*frégate*› (Eng. frigate) with ambiguity between the boat and the bird. Let C be the vector related to ‹*plume*› (feather) which is itself ambiguous, we have the following values:

$$D_A(V(\text{‹}\textit{frégate(icl>boat)}\text{›}), V(\text{‹}\textit{plume}\text{›})) = 1.1$$
$$\alpha_i = cot(1.1) = 0.5$$

$$D_A(V(\text{‹}\textit{frégate(icl>bird)}\text{›}), V(\text{‹}\textit{plume}\text{›})) = 0.5$$
$$\alpha_i = cot(0.5) = 2.18$$

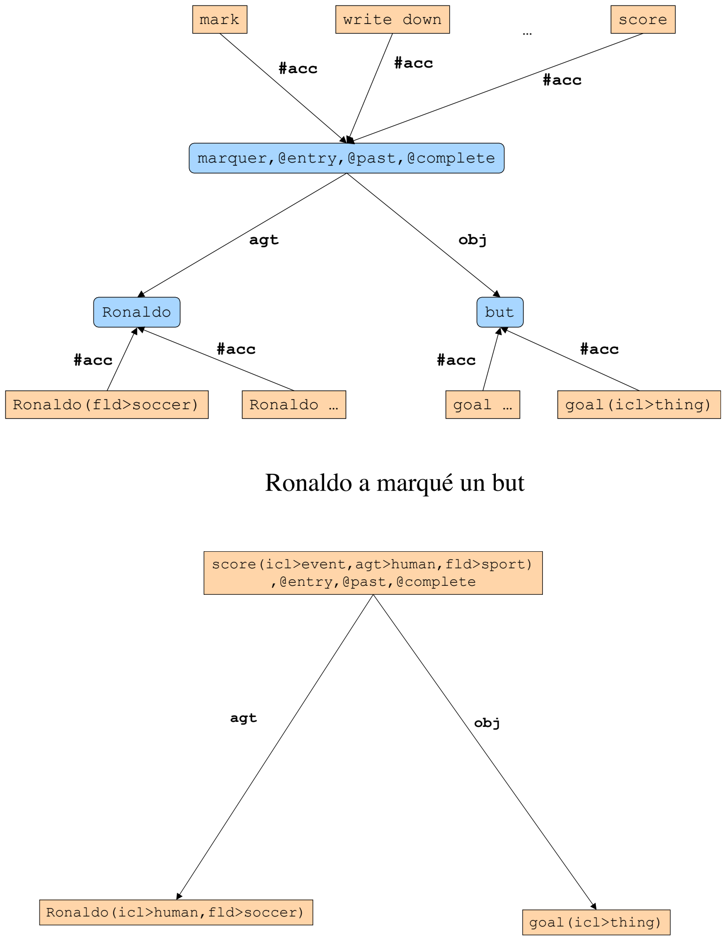
Thanks to the thematic context, the most activated meaning of ‹*frégate*› in the context of ‹*plume*› is the bird, as it has much more weight than the other interpretation. Although useful, this process may no be sufficient as more than often words and meanings are related while not being in the same semantic field. This is why, the construction and the exploitation of lexical and semantic network is necessary. The construction of such a network is done through templates but also by filtering through thematic proximity.

3.2 Fuzzy UNL Graphs

We extend UNL graph by adding to new types of nodes: lexical and relation nodes. These nodes are only instrumental in the process of choosing which acceptions or relations have to be selected (see Fig. 1 and Fig. 2) . To link these nodes to standard nodes we use two new types of arc: *acc* for linking acceptions to their corresponding lexical node and, *rel* for linking relation nodes to lexical nodes.

4 Ant Algorithm on Fuzzy UNL Graphs

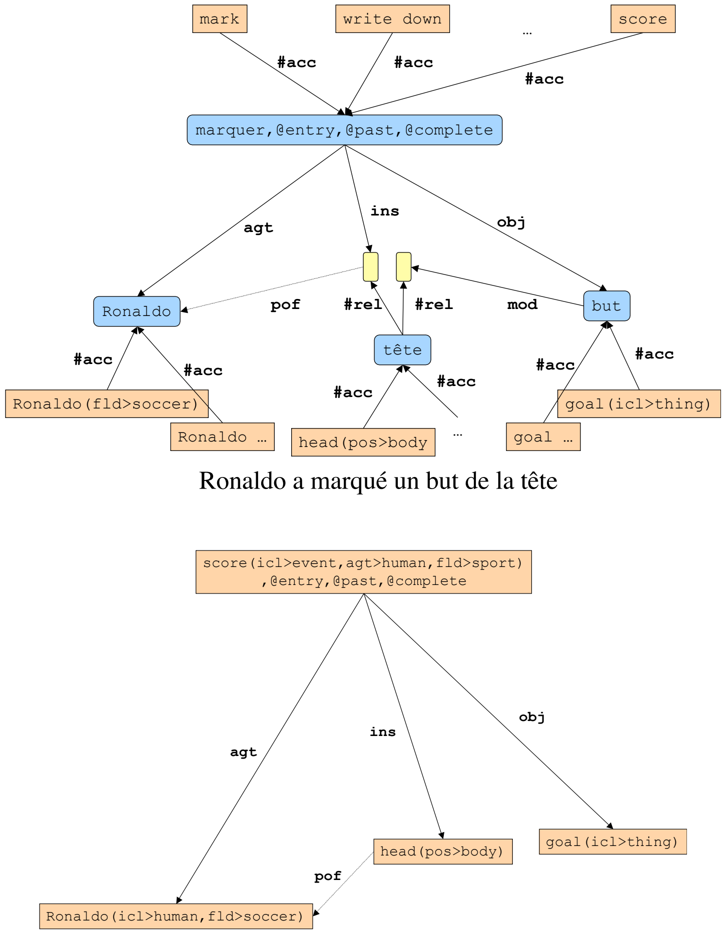
Each acception node behaves like an *ant nest* producing ants that propagate on the graph the conceptual vector associated to the acception. However, at each cycle of the simulation, the probability for a nest to create an ant is a function of its activation level $\mathrm{E}(N) \in [-\infty, +\infty]$. There is a cost ϵ (we set ϵ empirically to 0.1) for producing an ant, which is deducted from the nest energy. Each time, a nest produces an ant, its probability to generate another one at the next cycle is lowered. The probability of producing an ant, is related to a sigmoid function (see Figure 3) applied to the energy level of the nest. The definition of this function ensures that a nest has always the possibility to produce a new ant although the odds are low when the node is inhibited (energy below zero). A nest can still borrow energy and thus a word meaning has still a chance to express itself even



Ronaldo a marqué un but

Ronaldo a marqué un but (pure UNL graph)

Fig. 1. Example of the French sentence *Ronaldo a marqué un but.* (Lit. Eng. Ronaldo scored a goal). One the right, possible UNL graph. On the left, the fuzzy graph where each content word is represented through one lexical node which is linked to each corresponding acception. An example with *rel* relation is given with Figure 2.



Ronaldo a marqué un but de la tête

Ronaldo a marqué un but de la tête (pure UNL graph)

Fig. 2. Relation nodes are used (for example) when attachments are ambiguous. In the sentence *Ronaldo a marqué un but de le tête.* (Lit. Eng. Ronaldo scored a goal of the head), the GN *de la tête* may be a *mod* of *goal* or an *inst* of *score* (proper interpretation).

if the environment is very unfriendly. For a given lexical node, at each cycle at least one ant should be produced among the various acceptions.

Nests should count on ants of other nests to improve their energy level. In effect, in their wandering other ants may arrive at a given acception node (not their own) and give an amount of energy δ equal to :

$$\delta = \mathrm{DS}_A(N, A) = 1 - \frac{2D_A(V(A), V(N))}{\pi} \tag{3}$$

where $V(A)$ is the vector of the ant A and $V(N)$ the vector of the node N (N should not be the nest of A). W call this value DS_A (as Distance Similarity) as it is the distance D_A mapped from $[0, \frac{\pi}{2}]$ to $[1, 0]$. We see here that if A bears a vector that resembles very much the node it encounters, then a large amount of energy will be given. To induce some population control, each ant has a life span L of a finite number of cycles after which it dies (we found experimentally that $L = 30$ is a good trade-off between convergence of the simulation and resources).

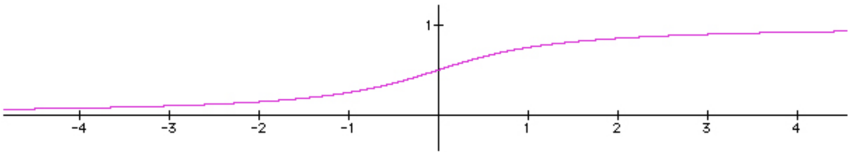


Fig. 3. Sigmoid function: $\mathrm{Sig}(x) = \frac{1}{\pi}\arctan(x) + 0.5$. Some values are: $\mathrm{Sig}(0) = 1/2, \mathrm{Sig}(1) = 0.75, \mathrm{Sig}(2) = 0.852, \mathrm{Sig}(-1) = 0.25, \mathrm{Sig}(-2) = 0.147$.

Each time an ant traverses an arc, the excitement level of this arc is increased (this is metaphorically a small amount of pheromones that gives its name to ant algorithms). This excitement slowly decays over time, and if this arc is not visited for a long time it may reach a null excitement and will be deleted. Only *rel* and *acc* links can be deleted. At the beginning of the simulation, each arc excitement is equal to 1. Each time an ant enters a node that is not an acception, it modifies slightly the node vector:

$$V'(N) = V(N) \oplus \alpha V(A) \qquad \text{with} \quad \alpha = 0.01$$

This way, each ant propagates the vector on the graph. The ant displacement behavior is directly related to node vectors. Before moving, an ant examines each nodes linking to its current position. The probability $P(N_k, A)$ for an ant A to choose a particular node N_k is computed as follows:

$$P(N_x, A) = \mathrm{DS}_A(N_k, A) / \sum_{1 \leq i \leq p} \mathrm{DS}_A(N_i, A) \tag{4}$$

At the beginning, only acception nodes have a conceptual vector. A node without vector is considered having the null vector. Over time, non acception nodes have vectors that correspond to the ant population distribution passing by them. From an acception, its vector slowly propagates outward, and ants may eventually find some *friendly* nests. The algorithm is purely altruistic as a nest will receive energy only by stranger ants. To be successful, which means being able to maintain a high level of energy and a large ant population, a nest should find some support in other nests.

After some cycles (around 300 for the examples given in this paper), the activations and vectors on the graph have converged. That is they are not much modified by further ant activity. A cleaning stage is then performed to obtain a standard UNL graph. On remaining *acc* and *rel* links related to a lexical node, only the most activated one is kept, others are deleted. Then, inaccessible nodes are suppressed. Finally, each lexical node are replaced by the unique acception left.

4.1 Examples with only lexical nodes

In the sentence presented in Fig. 1 we have only a lexical ambiguity with *marquer*, *but* and possibly *Ronaldo*. Each acception is producing ants that are slowly spreading their conceptual vector. Notice that each produced ant decreases the energy level of its nest, thus the ant production, after an initial burst, tends to rapidly decrease. However, if we focus our attention on the node *score*, even early in the simulation, most of the ants attaining this node come from acception sharing much information (namely *goal(fld>soccer)* and *Ronaldo(icl>human,fld>soccer)*). Other acceptions are not able to maintain their population level and the graph is eventually swarmed by ants from activated acceptions. Figure 4 illustrates an intermediary step where everything seems to be already settled.

4.2 Example with lexical and relation nodes

In the French sentence *Il regarde la fille avec un telescope*, we focuse our attention on relations and attachments (Fig 5). The acceptions *watch* and *telescope* support mutually more than any others. Furthermore, the whole path between both acceptions is shorter through the *ins* relation which induces less information dissipation. Eventually, the *rel* link related to the *mod* relation disappears. We should note here, that for *fille* the thematic context doesn't help, other insformation like acception distribution should be used.

In the French sentence *Ronaldo a marqué un but de la tête*, we have the situation of Figure 4 plus an attachment and relation difficulty similar to Figure 5. The lexical desambigation is reinforced with *tête* as an instrument of *score* and a part-of *Ronaldo*. Furthemore, the sharing between acceptions of *tête* and *but* is too low to compete and maintain.

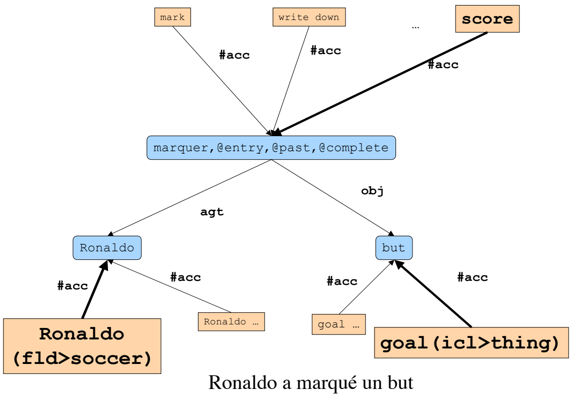


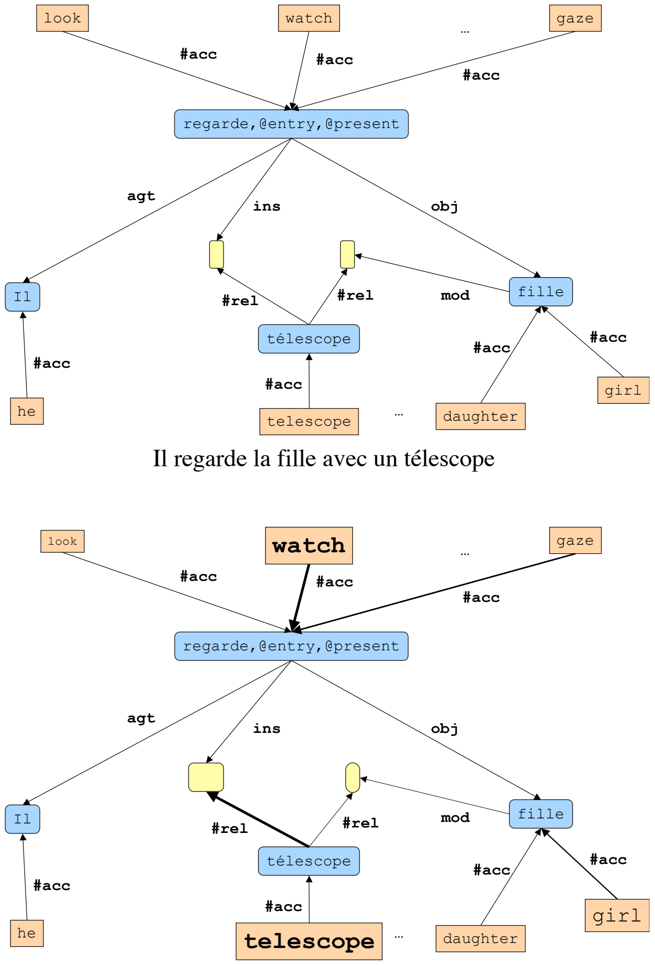
Fig. 4. By mutual information sharing with conceptual vectors, the ant circulation quickly converges between selected acceptions. After some time, poorly activated nodes are not able to maintain any population level and related links disappears.

5 Conclusion

This paper has presented an approach extending UNL graph by including lexical and relation nodes and links, such a way to accommodate word senses and attachment ambiguities. This *fuzzy* UNL graph is created by some transformation on a morpho-syntactic tree. On this structure, we do propagate constraints to performs a disambiguation task. The propagation is directly inspired from *ant algorithm* and is formally identical to the Traveling Salesman Problem. The information exploited for the ant propagation are the topology of the graph and the mutual information between the conceptual vectors used for meaning representation.

We have defined some underlying principles to our approach. First, it is interesting to combine rich thematic representation like conceptual vectors and symbolic constraints as found in the UNL knowledge base. Then, uncertainty should be tackled explicitly and globally both under lexical and relation aspects. If we consider how vocables and knowledge are processed psycholinguistically, we have definitive advantages to mix vocable nodes and meaning nodes. This last aspect is very instrumental for the selection process.

Our strategies have been prototyped and tested on various French sentences and shorts texts. The obtained UNL graphs are very satisfactory, and all in all the approach seems very promising. For texts, sentense graphs were sequentially linked to each other by an abstract *text* node. It is also used for comforting conceptual vector calculation and detecting inconsistencies either in thematic association or in relations between vocables. Nevertheless, in some quite difficult cases, activation level of the graph nodes do not converge but oscillates between



Il regarde la fille avec un télescope

Il regarde la fille avec un télescope

Fig. 5. Because of the mutual support between *watch* and *telescope*, the *ins* relation emerges compared to the *mod* relation.

states. This is especially true of humorous sentence with double entendre. A desirable extension of our model is to enrich the representation with other types of constraints like lexical preferences, statistical co-occurences, to name a few.

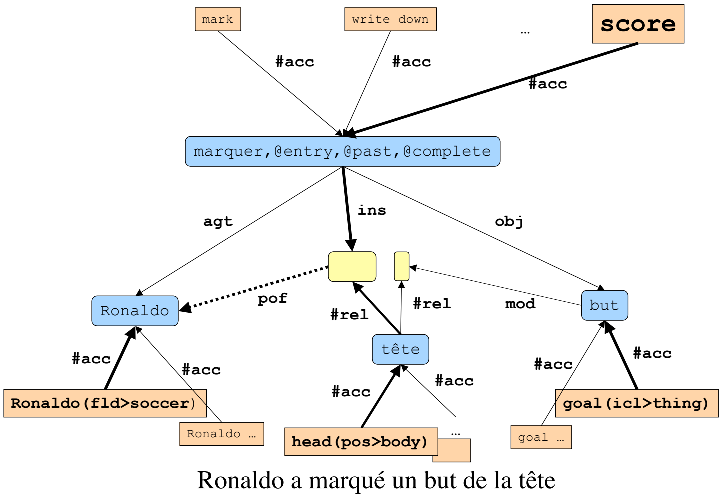


Fig. 6. Lexical selection induces relation selection (of *ins* opposed to *mod* in this example), which in turn reinforces acception activation.

References

[Cha90] J. Chauché. Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance. *TAL Information*, 1990.

[DDL$^+$90] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[Laf01] M. Lafourcade. Lexical sorting and lexical transfer by conceptual vectors. In *Proceeding of the First International Workshop on MultiMedia Annotation*, Tokyo, January 2001.

[Rod52] P. Rodget. *Thesaurus of English Words and Phrases.* Longman, London, 1852.

[SM83] G. Salton and M. McGill. *Introduction to Modern Information Retrieval.* McGrawHill, 1983.

References

[*Chauché* 90] Jacques Chauché., *Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance.* TAL Information, 31/1, pp 17-24, 1990.

[*Deerwester* et al. 90] Deerwester S. et S. Dumais, T. Landauer, G. Furnas, R. Harshman., *Indexing by latent semantic anlysis.* In Journal of the American Society of Information science, 1990, 416(6), pp 391-407.

[*Dorigo* et al. 1997] Dorigo M., and L. Gambardella., *Ant colony system : A cooperative learning approach to the travelling saleman problem.*, *IEEE Transactions on Evolutionary Computation*, 1(1), p. 114-128, 1997.

[*Hofstadter* 1995] Hofstadter, D. R., *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought (together with the Fluid Analogies Research Group)*, NY: Basic Books, 1995.

[*Lafourcade* 2001] Lafourcade M., *Lexical sorting and lexical transfer by conceptual vectors.* Proc. of the First International Workshop on MultiMedia Annotation (Tokyo, Janvier 2001), 6 p.

[*Lafourcade* et al. 2002] Lafourcade M. , and Ch. Boitet., *UNL lexical Selection with Conceptual Vectors* Proc. of LREC'2002, Las Palmas, Canary Island, Spain, May 27, 2002.

[*Larousse* 1992] Larousse., *Thésaurus Larousse - des idées aux mots, des mots aux idées.* Larousse, ISBN 2-03-320-148-1, 1992.

[*Lowe* 2000] Lowe, W., *Topographic Maps of Semantic Space*, *PhD Thesis*, institute for Adaptive and Neural Computation, Division of Informatics, Edinburgh University, 2000.

[*Kawamoto* 1993] Kawamoto, A. H., *Nonlinear dynamics in the resolution of lexical ambiguity: A parallel distributed processing account.* Journal of Memory and Language, 32, 474-516.

[*Miller* 1991] G.A Miller and C. Fellbaum., *Semantic Networks in English.* in Beth Levin and Steven Pinker (eds.) *Lexical and Conceptual Semantics* , 197–229, Elsevier, Amsterdam, 1991.

[*Pasteels* et al. 1987] J.M. Pasteels, J.L. Deneubourg, S. Goss, D. Fresneau, and J.P. Lachaud., Self-organization mechanisms in ant societies (ii): learning in foraging and division of labour. *Experientia Supplementa*, 54:177–196, 1987.

[*Prince* et al. 2003] Prince V. and Lafourcade M., *Mixing Semantic Networks and Conceptual Vectors: the Case of Hyperonymy.* Proc. of ICCI-2003 (2nd IEEE International Conference on Cognitive Informatics), South Bank University, London, UK, August 18 - 20, 2003.

[*Roget* 1852] Roget P., *Thesaurus of English Words and Phrases.* Longman, London, 1852.

[*Salton* et al. 1983] Salton G. and M.J. MacGill., *Introduction to modern Information Retrieval* McGraw-Hill, New-York, 1983.

[*Yarowsky* 1992] Yarowsky D., *Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora*, *COLING'92*, Nantes, p. 454-460, 1992.

Term-Based Ontology Alignment

Virach Sornlertlamvanich, Canasai Kruengkrai, Shisanu Tongchim,
Prapass Srichaivattana, and Hitoshi Isahara

Thai Computational Linguistics Laboratory
National Institute of Information and Communications Technology
112 Paholyothin Road, Klong 1, Klong Luang, Pathumthani 12120, Thailand
{virach,canasai,shisanu,prapass}@tcllab.org, isahara@nict.go.jp

Abstract. This paper presents an efficient approach to automatically align concepts between two ontologies. We propose an iterative algorithm that performs finding the most appropriate target concept for a given source concept based on the similarity of shared terms. Experimental results on two lexical ontologies, the MMT semantic hierarchy and the EDR concept dictionary, are given to show the feasibility of the proposed algorithm.

1 Introduction

In this paper, we propose an efficient approach for finding alignments between two different ontologies. Specifically, we derive the source and the target ontologies from available language resources, i.e. the machine readable dictionaries (MDRs). In our context, we consider the ontological concepts as the groups of lexical entries having similar or related meanings organized on a semantic hierarchy. The resulting ontology alignment can be used as a semantic knowledge for constructing multilingual dictionaries.

Typically, bilingual dictionaries provide the relationship between their native language and English. One can extend these bilingual dictionaries to multilingual dictionaries by exploiting English as an intermediate source and associations between two concepts as semantic constraints.

Aligning concepts between two ontologies is often done by humans, which is an expensive and time-consuming process. This motivates us to find an automatic method to perform such task. However, the hierarchical structures of two ontologies are quite different. The structural inconsistency is a common problem [1]. Developing a practical algorithm that is able to deal with this problem is a challenging issue.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 provides the description of the proposed algorithm. Section 4 presents experimental results and findings. Finally, Section 5 concludes our work.

2 Related Work

Chen and Fung [2] proposed an automatic technique to associate the English FrameNet lexical entries to the appropriate Chinese word senses. Each FrameNet

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 138–144.

lexical entry is linked to Chinese word senses of a Chinese ontology database called HowNet. In the beginning, each FrameNet lexical entry is associated with Chinese word senses whose part-of-speech is the same and Chinese word/phrase is one of the translations. In the second stage of the algorithm, some links are pruned out by analyzing contextual lexical entries from the same semantic frame. In the last stage, some pruned links are recovered if its score is greater than the calculated threshold value. Ngai *et al.* [3] also conducted some experiments by using HowNet. They presented a method for performing alignment between HowNet and WordNet. They used a word-vector based method which was adopted from techniques used in machine translation and information retrieval. Khan and Hovy [4] presented an algorithm to combine an Arabic-English dictionary with WordNet. Their algorithm also tries to find links from Arabic words to WordNet first. Then, the algorithm prunes out some links by trying to find a generalization concept.

3 The Algorithm

In this section, we describe an approach for ontology alignment based on term distribution. To alleviate the structural computation problem, we assume that the considered ontology structure has only the hierarchical (or taxonomic) relation. One may simply think of this ontology structure as a general tree, where node of each tree is equivalent to a concept.

Given two ontologies called the source ontology $\mathcal{T}_s$ and the target ontology $\mathcal{T}_t$, our objective is to align all the concepts (or semantic classes) between these two ontologies. Each ontology consists of concepts, denoted by $\mathcal{C}_1, \ldots, \mathcal{C}_k$. In general, the concepts and their corresponding relations of each ontology can be significantly different due to the theoretical background used in the construction process. However, for the lexical ontologies such as the MMT semantic hierarchy and the EDR concept dictionary, it is possible that the concepts may contain shared members in terms of English words. Thus, we can match the concepts between two ontologies using the similarity of the shared words.

In order to compute the similarity between two concepts, we must also consider their related child concepts. Given a root concept $\mathcal{C}_i$, if we flatten the hierarchy starting from $\mathcal{C}_i$, we obtain a nested cluster, whose largest cluster dominates all subclusters. As a result, we can represent the nested cluster with a feature vector $\mathbf{c}_i = (w_1, \ldots, w_{|\mathcal{V}|})^T$, where features are the set of unique English words $\mathcal{V}$ extracted from both ontologies, and w_j is the number of the word j occurring the nested cluster i. We note that a word can occur more than once, since it may be placed in several concepts on the lexical ontology according to its sense.

After concepts are represented with the feature vectors, the similarity between any two concepts can be easily computed. A variety of standard similarity measures exists, such as the *Dice coefficient*, the *Jaccard coefficient*, and the *cosine* similarity [5]. In our work, we require a similarity measure that can reflect the degree of the overlap between two concepts. Thus, the Jaccard coefficient

Algorithm 1: OntologyAlignment

```
input  : The source ontology $\mathcal{T}_s$ and the target ontology $\mathcal{T}_t$.
output : The set of the aligned concepts $\mathcal{A}$.

begin
    Set the starting level, $l \leftarrow 0$;
    while ${\mathcal{T}_s}^{\langle l \rangle} \leq {\mathcal{T}_s}^{\langle max \rangle}$ do
        Find all child concepts on this level, $\{\mathcal{C}_i\}_{i=1}^{k} \in {\mathcal{T}_s}^{\langle l \rangle}$;
        Flatten $\{\mathcal{C}_i\}_{i=1}^{k}$ and build their corresponding feature vectors, $\{\mathbf{c}_i\}_{i=1}^{k}$;
        For each $\mathbf{c}_i$, find the best matched concepts on $\mathcal{T}_t$,
            $\mathcal{B} \leftarrow$ FindBestMatched($\mathbf{c}_i$);
            $\mathcal{A} \leftarrow \mathcal{A} \cup \{\mathcal{B}, \mathcal{C}_i\}$;
        Set $l \leftarrow l + 1$;
    end
end
```

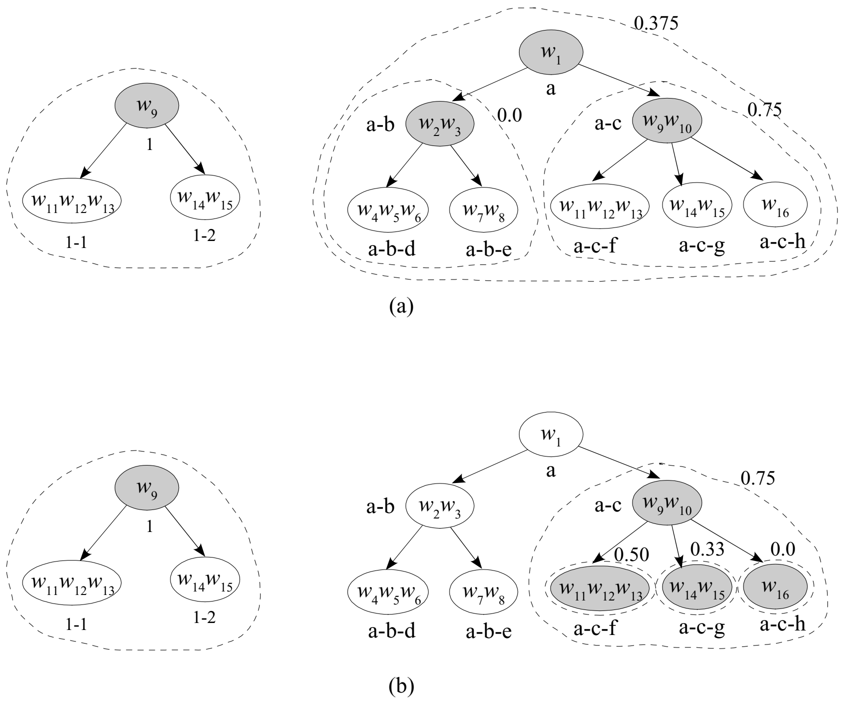
Algorithm 2: FindBestMatched($\mathbf{c}_i$)

```
begin
    Set the starting level, $l \leftarrow 0$;
    $BestConcept \leftarrow \mathcal{T}_t$(root concept);
    repeat
        $s_{tmp} \leftarrow JaccardSim(\mathbf{c}_i, BestConcept)$;
        if ${\mathcal{T}_t}^{\langle l \rangle} \leq {\mathcal{T}_t}^{\langle max \rangle}$ then
            return $BestConcept;$
        Find all child concepts on this level, $\{\mathcal{B}\}_{j=1}^{h} \in {\mathcal{T}_t}^{\langle l \rangle}$;
        Flatten $\{\mathcal{B}_j\}_{j=1}^{h}$ and build corresponding feature vectors, $\{\mathbf{b}_j\}_{i=1}^{h}$;
        $s_{j^*} \leftarrow \text{argmax}_j JaccardSim(\mathbf{c}_i, \{\mathbf{b}_j\}_{j=1}^{h})$;
        if $s_{j^*} > s_{tmp}$ then
            $BestConcept \leftarrow \mathcal{B}_{j^*}$;
        Set $l \leftarrow l + 1$;
    until $BestConcept$ does not change;
    return $BestConcept$;
end
```

is suitable for our task. Recently, Strehl and Ghosh [7] have proposed a version of the Jaccard coefficient called the *extended Jaccard similarity* that can work with continuous or discrete non-negative features. Let $\|\mathbf{x}_i\|$ be the L_2 norm of a given vector $\mathbf{x}_i$. The extended Jaccard similarity can be calculated as follows:

$$JaccardSim(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\|^2 + \|\mathbf{x}_j\|^2 - \mathbf{x}_i^T \mathbf{x}_j} \ . \tag{1}$$

We now describe an iterative algorithm for term-based ontology alignment. As mentioned earlier, we formulate that the ontology structure is in the form of the general tree. Our algorithm aligns the concepts on the source ontology $\mathcal{T}_s$ to



(a)

(b)

Fig. 1. An example of finding the most appropriate concept on $\mathcal{T}_t$ for the root concept $1 \in \mathcal{T}_s$

the concepts on the target ontology $\mathcal{T}_t$ by performing search and comparison in the top-down manner.

Given a concept $\mathcal{C}_i \in \mathcal{T}_s$, the algorithm attempts to find the most appropriate concept $\mathcal{B}^* \in \mathcal{T}_t$, which is located on an arbitrary level of the hierarchy. The algorithm starts by constructing the feature vectors for the current root concept on the level l and its child concepts on the level $l + 1$. It then calculates the similarity scores between a given source concept and candidate target concepts. If the similarity scores of the child concepts are not greater than the root concept, then the algorithm terminates. Otherwise, it selects a child concept having the maximum score to be the new root concept, and iterates the same searching procedure. Algorithms 1 and 2 outline our ontology alignment process.

Figure 1 shows a simple example that describes how the algorithm works. It begins with finding the most appropriate concept on $\mathcal{T}_t$ for the root concept $1 \in \mathcal{T}_s$. By flattening the hierarchy starting from given concepts ('1' on $\mathcal{T}_s$, and 'a', 'a-b', 'a-c' for $\mathcal{T}_t$), we can represent them with the feature vectors and measure their similarities. On the first iteration, the child concept 'a-c' obtains the maximum score, so it becomes the new root concept. Since the algorithm cannot find improvement on any child concepts in the second iteration, it stops the loop and the target concept ‘a-c’ is aligned with the source concept ‘1’. The algorithm proceeds with the same steps by finding the most appropriate concepts on $\mathcal{T}_t$ for the concepts ‘1-1’ and ‘1-2’. It finally obtains the resulting concepts ‘a-c-f’ and ‘a-c-g’, respectively.

4 Evaluation

4.1 Data Sets

In order to study the behavior of the proposed algorithm, two dictionaries are used in our experiments. The first one is the EDR Electronic Dictionary [6]. The second one is the electronic dictionary of Multilingual Machine Translation (MMT) project [8].

The EDR Electronic Dictionary consists of lexical knowledge of Japanese and English divided into several sub-dictionaries (e.g., the word dictionary, the bilingual dictionary, the concept dictionary, and the co-occurrence dictionary) and the EDR corpus. In the revised version (version 1.5), the Japanese word dictionary contains 250,000 words, while the English word dictionary contains 190,000 words. The concept dictionary holds information on the 400,000 concepts that are listed in the word dictionary. Each concept is marked with a unique hexadecimal number.

For the MMT dictionary, we use the Thai-English Bilingual Dictionary that contains around 60,000 lexical entries. The Thai-English Bilingual Dictionary also contains sematic information about the case relations and the word concepts. The word concepts are organized in a manner of semantic hierarchy. Each word concept is a group of lexical entries classified and ordered in a hierarchical level of meanings. The MMT semantic hierarchy is composed of 160 concepts.

In our experiments, we used a portion of the MMT semantic hierarchy and the EDR concept dictionary as the source and the target ontologies, respectively. We considered the ‘animal’ concept as the root concepts and extracted its related concepts. However, in the EDR concept dictionary, the relations among concepts are very complex and organized in the form of the semantic network. Thus, we pruned some links to transform the network to a tree structure. Starting from the ‘animal’ concept, there are more than 200 subconcepts (containing about 7,600 words) in the EDR concept dictionary, and 14 subconcepts (containing about 400 words) in the MMT semantic hierarchy. It is important to note that these two ontologies are considerably different in terms of the number of concepts and words.

4.2 Preliminary Results

Table 1 shows alignment results generated by our algorithm. Here we divide the mapping into two types: *exact* and *subset*. The exact mapping occurs when the MMT concept exactly matches the EDR concept. The subset mapping occurs when the definition of a given MMT concept does not appear in the EDR concept

Table 1. Results of aligned concepts between the MMT semantic hierarchy and the EDR concept dictionary

MMT concept	EDR concept	Mapping
vertebrate	vertebrate	exact
warm-blood	mammal	subset
mammal	mammal	exact
bird	bird	exact
cold-blood	reptile	subset
fish	fish	exact
amphibian	toad	subset
reptile	reptile	exact
snake	snake	exact
invertebrate	squid	subset
worm	leech	subset
insect	hornet	subset
shellfish	crab	subset
other sea creature	squid	subset

dictionary, so the algorithm tries to find the most suitable concept. Since the EDR concepts are more fine-grained than the MMT concepts, the definition of the resulting concept often is the subset of the source concept.

From 14 MMT concepts, 6 concepts are exactly matched with the EDR concepts, e.g. ‘mammal’, ‘bird’, and ‘fish’ concepts. The remaining 8 concepts are mapped to the closely related EDR concepts. For example, the ‘warm-blood’ concept in MMT is mapped to the ‘mammal’ concept in EDR. Although the ‘warm-blood’ concept does not occur in the EDR concept dictionary, some words in this concept appear to be a part of the ‘mammal’ concept in EDR. Moreover, a child concept of the ‘warm-blood’ concept is the ‘mammal’ concept. Thus, the algorithm decides to align the ‘warm-blood’ concept with the most similar EDR concept, which is the ‘mammal’ concept.

Figure 2 shows an example of aligned concepts found by our algorithm. The exact mapping can be found if two ontologies have the equivalent concepts and their elements overlap enough for resulting the maximum matching score. Also, the algorithm can yield the most appropriate concepts located on an *intermediate* level of the target ontology.

5 Conclusion and Future Work

This paper has described our first attempt to deal with the problem of automated ontology alignment. We present an efficient algorithm to align concepts between two ontologies based on the similarity of the shared terms. Our algorithm aligns the concepts between the source ontology and the target ontology by performing search and comparison in the top-down manner. Preliminarily experimental results show that the proposed algorithm can find reasonable concept mappings between two ontologies.

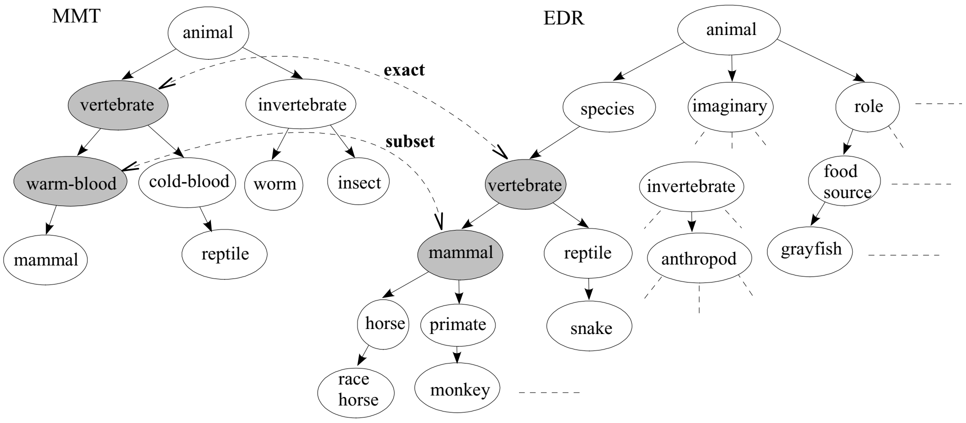


Fig. 2. An example of aligned concepts found by our algorithm

In future work, we plan to investigate our algorithm with larger data sets. Furthermore, we anticipate to apply a model selection technique such as Minimum Description Length (MDL) for generalizing the resulting concepts onto more coarse-grained concepts.

References

1. Ide, N. and Véronis, J.: Machine Readable Dictionaries: What have we learned, where do we go?. Proceedings of the International Workshop on the Future of Lexical Research, Beijing, China (1994) 137–146
2. Chen, B. and Fung, P.: Automatic Construction of an English-Chinese Bilingual FrameNet. Proceedings of Human Language Technology conference, Boston, MA (2004) 29–32
3. Ngai, G., Carpuat , M. and Fung, P.: Identifying Concepts Across Languages: A First Step towards a Corpus-based Approach to Automatic Ontology Alignment. Proceedings of the 19th International Conference on Computational Linguistics, Taipei, Taiwan (2002)
4. Khan, L. and Hovy, E.: Improving the Precision of Lexicon-to-Ontology Alignment Algorithms. Proceedings of AMTA/SIG-IL First Workshop on Interlinguas, San Diego, CA (1997)
5. Manning, C. D., and Schütze, H.: Foundations of statistical natural language processing. MIT Press. Cambridge, MA (1999)
6. Miyoshi, H., Sugiyama, K., Kobayashi, M. and Ogino, T.: An Overview of the EDR Electronic Dictionary and the Current Status of Its Utilization. Proceedings of the 16th International Conference on Computational Linguistics (1996) 1090–1093
7. Strehl, A., Ghosh, J., and Mooney, R. J.: Impact of similarity measures on web-page clustering. In Proceedings of AAAI Workshop on AI for Web Search (2000) 58-64
8. CICC: Thai Basic Dictionary. Center of the International Cooperation for Computerization, Technical Report 6-CICC-MT55 (1995)

Universal Networking Language: A Tool for Language Independent Semantics?

Amitabha Mukerjee, Achla M Raina, Kumar Kapil,
Pankaj Goyal, Pushpraj Shukla

Indian Institute of Technology, Kanpur, India
{amit,achla,kapil,pankajgo,praj}@iitk.ac.in

Abstract. Given source text in several languages, can one answer queries in some other language, without translating any of the sources into the language of the questioner? While this task seems extremely difficult at first sight, it is possible that the ongoing UN sponsored Universal Networking Language (UNL) proposal may hold some clues towards achieving this distant dream. In this paper we present a partially implemented solution which shows how UNL, though not designed with this as the primary objective, can be used as the predicate knowledge base on which inferences can be performed. Semantic processing is demonstrated by Question Answering. In our system as of now, both the text corpus and the questions are in English, but if UNL can deliver on its promise of a single homogeneous language-independent encoding, then it should be possible to achieve question answering and other semantic tasks in any language.

1 Semantics Models And UNL

Many organizations worldwide are grappling with problems like the following: Given source text in several European languages, would it be possible to demonstrate semantic understanding in some other language (like Hindi) without explicitly translating any of the sources into the language of the questioner? This is, of course, an extremely difficult task, perhaps even an impossibly difficult task. We trust the reader will realize that this paper is merely a very preliminary investigation as indicated by the hesitant "?" at the end of the paper's title. The key insight driving this research is the realization that if there is a mechanism for mapping any language into a uniform language-independent predicate structure, then it would constitute an important tool in this direction. While no system worldwide is anywhere near succeeding in this effort, the ongoing work on Universal Networking Language (UNL) [2] appears to hold the highest promise in terms of delivering on this dream.

UNL was developed as a universal knowledge-encoding mechanism, and is being primarily driven by the needs of the MT community. UNL provides for a uniform concept vocabulary (called "universal words" or UW's – the same concept in any language results in the same UW, which is written out using English orthography). These UW's are connected by a small set of about thirty-eight binary relations to obtain a set of predicate expressions that can encode the linguistic content of any sentence in any language of the world. One of the philosophical issues of course, is that the same con-

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 145–154.

cept, as expressed in different languages, does not define an identical chunk of conceptual space and at best, the UW's are approximations to the overlapping part of this concept. Despite such philosophical indulgences, a number of groups around the world are working on constructing a UNL KB (a knowledge structure linking the concepts underlying the UW's in terms of the probability of certain relations holding between them), and on constructing enconverters (from NL to UNL) and deconverters (from UNL to NL) for several languages across the world. In this latter sense, the UNL may be thought to be an inter lingua, but UNL has a number of other features that make it better suited for semantic inference than most other interlinguas. In particular, the following features of UNL motivate this work:

1. The set of Universal Words with well defined universal interpretations,
2. a small, simple predicate structure with only binary predicates,
3. a knowledge base connecting the UW's as a weighted graph of relations.
4. ontological information that is built-in to the UWs (eg. cholera(icl<disease) characterizes cholera as a type of disease).
5. The world wide effort in developing mechanisms for converting language into UNL and vice versa.
6. The dream of language independent semantic analysis.

Even aside from the language independence claim, there are merits to using a coherent labeling structure as provided by the UW's. Models for semantics require a basic set of predicates into which sentences from Natural Language would be mapped. All such efforts, e.g. CYC[10] have been plagued by considerable divergence in semantic analysis. By removing multiple models of reflecting the same concept at source, UW's help this objective significantly. It may be argued that other tools such as WordNet [11] provide a richer ontology and lexical knowledge for this task, but they do not provide the predicate structure, or the en/de -converting tools of UNL.

2 Present Work

This work makes two major claims:

- that a substantial amount of logical inference is possible on the UNL representation of language as UNL expressions,
- that for the question answering task in particular, given that enconverters to UNL and deconverters from UNL are indeed available, the only task remaining to achieve such an objective is to construct the question to answer-template UNL mapping for the target language.

Some of these goals are clearly far in the future and even if it happens that some aspects of the UNL experiment may not quite succeed, it is still likely that the effort would lead to insights applicable to any other model for language-independent semantics.

In our implementation we demonstrate both inference, and question answering - though at this point, both of these operate on English alone. The Q&A is achieved using a content-level HPSG lexicon tagged with the appropriate UNL relations. In earlier work we have used the same HPSG structure for looking at English, Hindi and also codemixed bilingual structures, and in future, we hope to demonstrate the Q&A aspects in Hindi with the source text in English itself.

In practically implementing this interface, we have to deal with an actual corpus written in some language, and also a procedure for testing the degree of success in modeling the semantics. In this case, we have chosen an English-based corpus for safe drinking water and as a test procedure we have developed an English based question and answer system, in the course of which we have also developed a lexicon of transformational rules for a subset of English questions. Since semantic modeling requires models of a body of ”commonsense” knowledge and associated pragmatic rules, which in this instance need to be created manually, we have restricted ourselves to a limited domain—that of drinking water.

The Question Answering module comprises traditional modules such as a syntactic parser, logical representation of the text (UNL), built in ontologies (UNLKB), inference engine, question processing, document retrieval, answer extraction and others [1].

2.1 From Natural Language to UNL

The corpus for the present work was built from documents obtained from the official websites of EPA and WHO [12]. Since the enconverters mapping NL corpus into a UNL document are not yet available [2], the mapping was done manually. To build the corpus KB, we marginally edited the source document, e.g. dropping the phrase “*just like man-made chemicals*”in the source sentence “*At certain levels, minerals, just like man-made chemicals, are considered contaminants that can make water unpalatable or even unsafe*” because we could not find a clear definition for *’just like’* phrases in the UNL Specifications[3].

The resulting corpus was annotated and processed to generate the UNL document(the UNL expression for the corpus). The manual annotation of the corpus was done making use of a format specified by UNL [3]. For example, the NL corpus sentence, *At some level, minerals are considered contaminants that can make water unpalatable,* is annotated as follows:

```
<c>At some{<qua,>n} level.n.@pl.@entry</c>{<man,>p} mineral.@pl{<gol,>p} are
consider.p contaminant{1}.@pl{<obj,<p} that{<1}{<agt,>p} can
make.p.@possible water{<obj,<p} <c>unpalatable{<or,>p} or even{<man,>p}
unsafe.p.@entry</c>{<gol,<p}.
```

The corpus sentence in its annotated form is input to the UNL parser to generate the UNL parsed graph, represented as a list of relations, given below:

```
unl
obj(consider(icl>think(agt>volitional thing,gol>uw,obj>thing)):25,
contaminant:2G.@pl)
man(consider(icl>think(agt>volitional thing,gol>uw,obj>thing)):25,
:01)
gol(consider(icl>think(agt>volitional thing,gol>uw,obj>thing)):25,
mineral(icl>matter):1G.@pl)
qua:01(level:0K.@entry.@pl, some(mod<thing):06)
/unl
```

```
unl
gol(make(agt>thing,obj>thing,src>thing):3U.@possible,:02)
obj(make(agt>thing,obj>thing,src>thing):3U.@possible,
water(icl>liquid):4B)
agt(make(agt>thing,obj>thing,src>thing):3U.@possible,contaminant:2G)
or:02(unsafe(aoj>thing):5U.@entry,unpalatable(aoj>thing):4T)
man:02(unsafe(aoj>thing):5U.@entry,even:5G)
/unl
```

3 Inference Engine

Although UNL structure provides a first order logic encoding of natural language, it is not designed for making semantic inferences, and an inference engine needs to be built for this purpose. In addition, world knowledge about the domain is needed to provide context information which would be commonly known to the human reader but is not available from the text itself. For the purpose of the Question Answering system, the inference engine also provides a set of inferred facts which can be eventually matched with a pseudo-UNL form of the natural language Query in order to obtain an answer.

The domain used here is that of “drinkable water”. The UNL form of the input text consists of a set of UNL expressions, each of which is a binary predicate corresponding to one of the UNL relations. The arguments to these predicates are Universal Words, possibly modified by one or more attributes.

3.1 Pragmatic knowledge

A set of manually created rules encode the pragmatic knowledge in the system. These include facts such as the following:

- Water is essential for human life.
- Communicable diseases are caused through physical contact.
- Water-borne diseases are communicable.

The last rule would have a UNL structure as follows in the pragmatic rule base:

```
aoj(communicable(aoj>thing),water-borne
disease(icl>disease).@pl.@entry)
```

3.2 Semantic Equivalence

The same information may be expressed in very different ways:

1. Safe water can be obtained through boiling and distillation.
2. We can obtain safe water through boiling and distillation.
3. We can get safe water through boiling and distillation.
4. The methods for making safe water are boiling and distillation.

5. One can make safe water by boiling or distillation.
6. While distilling results in pure water, for practical purposes, boiling is sufficient to make water safe for drinking.

Fortunately, a part of this problem (e.g active vs passive voice) is resolved by the UNL encoding process – thus (1) and (2) will result in the same UNL structure:

agt$_1$(get$_2$(agt>thing,obj>thing,src>thing).@possible, we);
obj(get(agt>thing,obj>thing,src>thing), water(icl>liquid));
mod(water(icl>liquid), safe(mod<thing));
man(get(agt>thing,obj>thing,src>thing), through(icl>how(obj>thing)));
obj(through(icl>how(obj>thing)), :01);
and :01(boiling(icl>act), distillation(icl>act));

Even (3) which uses the word "get" which is used here in the same sense as "obtain" results in the same universal word

get(agt>thing,obj>thing,src>thing)

and thus result in the same UNL structure. However sentences (4 and 5) use "make" which has a different UW, and these are handled in the inference engine by using rules for unifying similar UWs when used in the context of water. Very wide variations such as (6), which requires added pragmatic knowledge such as "pure water is safe", and also results in a set of two conjunctive UNL expressions (one for the "while" clause, and the other for the main clause) can be handled but since the set of such constructs is very large, they are not handled in the current version.

3.3 First order Inference Rules

These rules implement the First Order Logic in order to obtain new inferences. For example, given the facts *Water- borne diseases are caused by ingestion of contaminated water*. and *cholera is a water-borne disease*., one may infer that *cholera is caused by ingestion of contaminated water*.

A meta-rule for this situation, incorporated as part of the inference rulebase is that, given

agt(cause(icl>abstract thing).@entry:1);
obj(cause(icl>abstract thing).@entry:2);
nam(2:3);

which says that variable 1 causes 2, and 3 is a type of 2. Given this set of UNL relations, the meta-rule says that one can infer:

agt(cause(icl>abstract thing).@entry:,1:);
obj(cause(icl>abstract thing).@entry:,3:);

i.e. 3 is caused by 1. The current system is designed to be tested only on a simple Question and Answer mechanism. We use single-tiered inferences, and construct a complete set of all possible inferences that can be made from the given text and the pragmatic rules. This is the final UNL knowledge base which is to be matched with a UNL form of the question to obtain the answer.

4 The Question And Answer Module

We use a structure matching approach to search for the answer to a question. This is done by building an answer template that represents the form of the potential answer corresponding to the question. This template is input to an HPSG Parser [7, 8] which outputs a pseudo UNL expression corresponding to the question as well as the answer template. The pseudo UNL expression is subsequently subject to a structure matching with the UNL document (the corpus Knowledge Base)

4.1 Question Processing

We generate an answer template that represents the form of the answer corresponding to a question with the "exact answer" slot filled in with an unknown variable "X". The existing literature on Q/A systems suggests several ways of building the template such as generic extraction using shallow parsing rules[4]. In this work we use a set of transformational rules to arrive at the answer template. The transformation from the question to the answer template is governed by a rule base with approximately 50 rules which range over various "wh" and other question formats, such as the "yes/no" question. The rules introduce the variable "X" at the location of the keyword or the key phrase in the answer pattern.

For example, the rule, *how:aux:1:V(ppl) > 1:aux:V(ppl) :by:X*, works upon a question such *as How is water contaminated?* which is transformed to its corresponding answer template with the variable "X" - *Water is contaminated by X*.

Taking another example, a rule of the form, what:does:1:V(base) >1:V(pres):X, maps the question, What does skin or eye contact with water cause? into the answer template Skin or eye contact with water causes X.

4.2 The Pseudo-UNL Enconverter

The answer template is converted into a pseudo UNL representation by a parser [8] which operates on a lexicon specifying the semantic selection (as against the categorical selection) properties of heads. Semantic relation attributes are used instead of syntactic subcat features since the parsed answer form needs to be unified with a database that is in the UNL format, i.e. the UNL Document. The UNL structure uses relations that are defined in terms of semantic features such as agency, place, etc. Therefore, these relations need to be identified in the parsed answer form for structure matching to be possible. To take an example of a lexical entry stating the said semantic feature information:

<make>@V(base){agt|!|obj|~gol}

In the entry for the verb "make" above, the description "{agt|!|obj|~gol}" captures the fact that the verb phrase headed by the verb "make" takes the form of an Agent followed by the verb itself and then an Object and an optional Goal". Note that agt, obj and gol are all UNL relations. The Nominal heads which take on the roles agent, object and goal are entered in the lexicon as follows:

```
<impurities>@agt(pl){~qua|~mod|!|~plc}
<water>@obj(sg){~qua|~mod|!|~plc}
<unpalatable>@gol{!}
```

The HPSG parser reads the lexicon and states relations as given in lexical entries. From the parsed tree thus obtained, we can get the relation between two nodes, which would essentially be the label attached with the child node. To take an example of how the pseudo-enconverter works, given the question, *What makes water unpalatable*?, we generate an answer template, *X makes water unpalatable*, with the transformational rule,

what:V:1 > X:V:1

The parsed output is as follows:

```
( ( Xagt(sg)) makesV(base) (waterobj(sg)) (unpalatablegol) )
+-X makes water unpalatable
+-X_agt(sg)
+-makes water unpalatable
+-makes_V(base)
+-water_obj(sg)
+-unpalatable_gol
```

The list of relations produced is -

```
agt(makes,X)
obj(makes,water)
gol(make,unpalatable)
```

Similarly, for the question, *How is cholera caused*?, we generate an answer template, *Cholera is caused by X*, with the transformational rule,

how:aux:1:V(ppl) > 1:aux:V(ppl):by:X

The parsed output is as follows:

```
( ( choleraobj(sg) ) (isaux) causedV(ppl) ( by ) ( Xagt(sg) ) )
+-cholera is caused by X
+-cholera_obj(sg)
+-is_aux
+-caused by X
+-caused_V(ppl)
+-by_by
+-X_agt(sg)
```

The list of relations produced is:

```
obj(caused,cholera)
aux(caused,is)
by(caused,by)
agt(caused,X)
```

In this case the output is filtered to retain the UNL relations (semantic relations) only i.e.

```
obj(caused,cholera)
agt(caused,X)
```

4.3 Answer extraction from UNL

Given the answer form of the question in pseudo-UNL format, it has to be matched with the final UNL knowledge base to see if an answer can be provided. First, each sentence in the knowledge base (as generated in section 2) is converted into an UNL graph, with two arguments as nodes, connected by a link with the label of the relation. Next, we convert the psuedo-UNL answer template as described in section 3.1 into the UNL graph. For example, given the question *How is Colera caused?*, one obtains the ”Query UNL Graph” as in Figure 1.

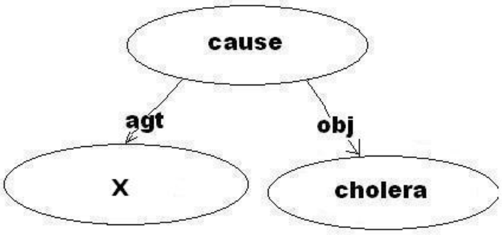


Fig. 1. UNL graph for the question ”How is cholera caused?”

Finding an answer involves matching this Query UNL graph with a UNL sub graph from the knowledge base, If one of the nodes in the query has a variable X then the match returns the value of this variable. If there is no variable then the match returns T. If no match is found, the system returns F.

For the above example, there is an inferred fact in the knowledge base for which the corresponding graph is shown in Figure 2. After sub graph matching, X is bound to the left-child of the node ”cause” in Figure 2.

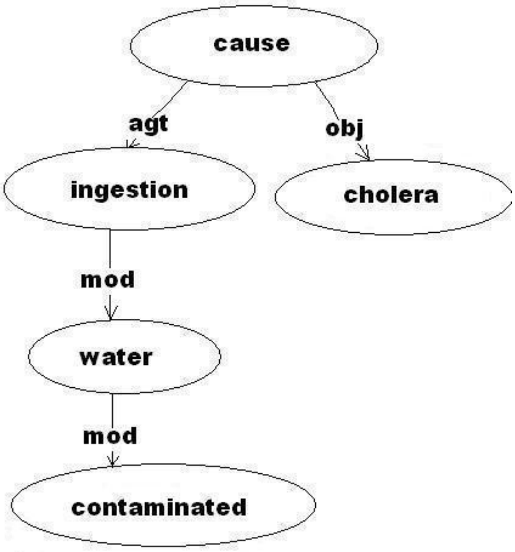


Fig. 2. UNL graph for the fact ”Cholera is caused by ingestion of contaminated water.”

In fully implemented UNL situations, this graph can now be pasted, into the answer template and passed to a deconverter which would then generate the full answer sentence. In our case, since no deconverter is being used, we label the sub graphs in the knowledge base with English strings, which are then used in the answer generation process to obtain answers as English sentences. Note that if the deconverter is for a different language, then the answers can also be generated in that language. In this case, the resulting answer is Cholera is caused by ingestion of contaminated water. Similarly, the question ”What makes water unpalatable?” results in the graph as shown in Figure 3 below, and after matching, results in the answer: "Contaminants make water unpalatable."

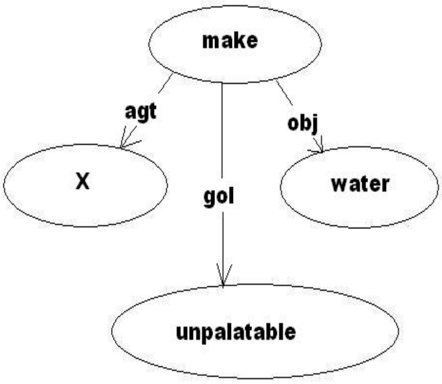


Fig. 3. UNL graph for the question ”What makes water unpalatable?”

5 Conclusion

This work takes a structure intended to represent the structure of a source language and convert it into other languages, and uses it as a query system that can answer questions based on textual databases, possibly in other languages. This is clearly only the first step – a lot more needs to be done to validate the feasibility of this process. A number of important issues remain. While the UNL structure is a First Order Predicate form, there are remarkable differences with normal logical models. For one, UNL structures do not provide for an implication connective, and also use the disjunction relation “or” rather sparingly. A rigorous mapping to more traditional logical structures is needed for more extensive UNL based logical inference. Efforts are on in this direction.

Also, the manual process of designing the pragmatic knowledge-base is expensive—it needs to be seen if further synergies can be gained by unifying this effort with parts of the UNL KB. Despite these shortcomings, we hope the present work will provide a start towards this difficult yet important problem. The Q/A module reported here can be successfully extended to other languages without any basic changes in the system design. A UNL-based Q/A system for Hindi, which can work on the Water domain, is expected to be implemented shortly.

Acknowledgment We are thankful to the UNDL Foundation, Geneva for the Universal Parser, which was used to generate the UNL expressions from the manually annotated UNL document.

References

1. Moldovan, D., C. Clark, S. Harabagiu and S. Maiorana, "COGEX: A Logic Prover for Question Answering", *Proceedings of HLT-NAACL 2003*, Main Papers, Edmonton, 2003, 87–93.
2 http://www.undl.org
3 UNL Specifications, http://www.unlc.undl.org/unlsys/
4. Diekema, A. R., J. Chen, N. McCracken, N. E. Ozgencil, M. D. Ta_et, O. Yilmazel, and E. D. Liddy, "Question Answering: CNLP at the TREC-2002 Question Answering Track", *http://www.cnlp.org*
5. Robertson, S. and S. Walker, "Okapo/Keebow at TREC-8", *Proceedings of the Eighth Text Retrieval Conference(TREC-8)*, Gaithesburg, Maryland,17-19 November,1999, 151-162.
6. Moldovan, D., M. Pasca, S. Harabagiu and M. Surdeanu, " Performance Issues and Error Analysis in an Open-domain Question-Answering System", *40th Annual Meeting of the Association for Computational Linguistics(ACL)*, Philadelphia, July, 2002, 33– 40.
7. Sharma, D., K. Vikram , M.R. Mital, A. Mukerjee, A.M. Raina, Saarthaka—an Integrated Discourse Semantic Model for Bilingual Corpora, *Online Proceedings of the International Conference on Universal Knowledge and Language,* Goa, India, 25-28 Nov, 2002.
8. Sharma, D., K. Vikram , M.R. Mital, A. Mukerjee, A.M. Raina, Saarthaka—A generalized HPSG parser for English and Hindi, *Recent Advances in Natural Language Processing - Proceedings ICON-2002*, Mumbai, India,18-21 Dec,2002.
9. Hong, M. and O. Strieter, *Overcoming the Language Barriers in the Web: The UNL- Approach*.
10. http://www.cyc.com/tech.html/cycl.
11. Fellbaum, C., "*WordNet: An Electronic Lexical Database"*, MIT Press, 1998.
12. Documents on Water Sanitation and Hygiene, http://www.who.int/inf-fs/en/fact112.html, http://www.epa.gov/safewater/dwh/contams.html.

APPLICATIONS

About and Around
the French Enconverter and the French Deconverter

Etienne Blanc

GETA, CLIPS-IMAG
BP53, F-38041 Grenoble Cedex 09
etienne.blanc@imag.fr

Abstract. We briefly describe the French Enconverter and the French Deconverter. We discuss then a few general points concerning the possibility of designing dependency trees equivalent to UNL graphs, the treatment of the ambiguity and anaphora resolution, and the structure of the compound nodes.

1 Introduction

In a previous paper [1], we described the basic principle of our French Enconverter, in which the UNL input graph is processed into an equivalent Dependency Tree, which is in turn applied to the entry of a rule-based French generator. We developed similarly a French enconverter, in which a French Analyser provides a representation of the text meaning as a Dependency Tree, which is further processed into an equivalent UNL graph.

In this paper, we will first briefly present the structure of the French Deconverter and Enconverter. We will then recall and discuss a little further than in our previous paper the general problem of the equivalency between UNL graph and dependency tree. And finally briefly comment on three topics we had to deal with when devising our Enconverter and Deconverter : Ambiguity and Anaphora Resolution, Processing of the Unknown Word, the exact structure of the Compound Node of a UNL graph.

2 Overall Structure of the French Deconverter and Enconverter

The French Enconverter and the French Deconverter are written on ARIANE-G5.

ARIANE-G5 is a generator of MT systems, which is an integrated environment designed to facilitate the development of MT systems. These MT systems are written by a linguist using specialised languages for linguistic programming. ARIANE is not devoted to a particular linguistic theory. The only strong constraint is that the structure representing the unit of translation (sentence or paragraph) must be a decorated tree.

Fig.1 shows an overview of a classical transfer MT system using the ARIANE environment. The processing is performed through the three classical steps: analysis, transfer and generation. An interactive disambiguation module may be inserted after

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 157–166.

the analysis step. Deconversion or Enconversion cannot be performed straightforward by ARIANE, whose inputs and outputs are texts or trees. Thus additional external modules are necessary, transforming a graph into an equivalent tree, or inversely.

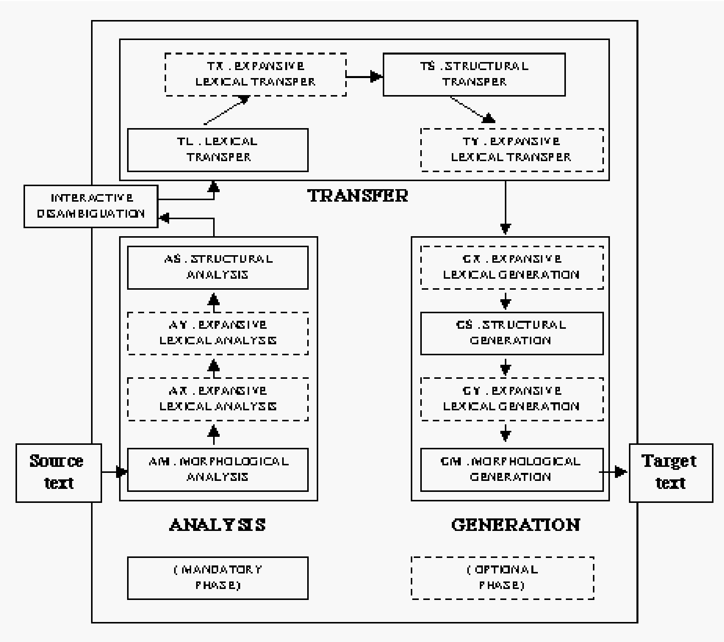

TX . EXPANSIVE LEXICAL TRANSFER
TS . STRUCTURAL TRANSFER
TL . LEXICAL TRANSFER
TY . EXPANSIVE LEXICAL TRANSFER
INTERACTIVE DISAMBIGUATION
TRANSFER
AS . STRUCTURAL ANALYSIS
AY . EXPANSIVE LEXICAL ANALYSIS
AX . EXPANSIVE LEXICAL ANALYSIS
AM . MORPHOLOGICAL ANALYSIS
GX . EXPANSIVE LEXICAL GENERATION
GS . STRUCTURAL GENERATION
GY . EXPANSIVE LEXICAL GENERATION
GM . MORPHOLOGICAL GENERATION
Source text
Target text
ANALYSIS
GENERATION
(MANDATORY PHASE)
(OPTIONAL PHASE)

Fig. 1. The Ariane-G5 environment as used for generating a transfer MT system

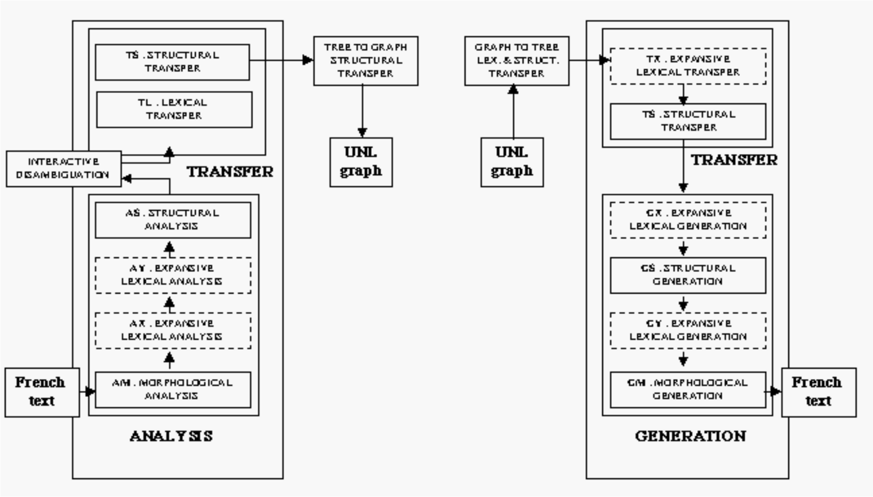

TS . STRUCTURAL TRANSFER
TL . LEXICAL TRANSFER
TREE TO GRAPH STRUCTURAL TRANSFER
UNL graph
GRAPH TO TREE LEX. & STRUCT. TRANSFER
UNL graph
TX . EXPANSIVE LEXICAL TRANSFER
TS . STRUCTURAL TRANSFER
INTERACTIVE DISAMBIGUATION
TRANSFER
TRANSFER
AS . STRUCTURAL ANALYSIS
AY . EXPANSIVE LEXICAL ANALYSIS
AX . EXPANSIVE LEXICAL ANALYSIS
AM . MORPHOLOGICAL ANALYSIS
GX . EXPANSIVE LEXICAL GENERATION
GS . STRUCTURAL GENERATION
GY . EXPANSIVE LEXICAL GENERATION
GM . MORPHOLOGICAL GENERATION
French text
French text
ANALYSIS
GENERATION

Fig. 2. French Enconverter (left) and French Deconverter (right) using Ariane-G5

Fig. 2 shows the overall structure of the French enconverter (left) and of the French Deconverter (right).

Enconversion takes place in two steps :

– Analysis of the French text producing a representation of its meaning in the form of a dependency tree (ARIANE Analyser).
– Lexical (from the French lemmas to the Universal Words) and structural (from tree to graph) transfer from the dependency tree to an equivalent UNL graph (External Transfer module).

Similarly, Deconversion takes place in the two following steps:

– Lexical (from the Universal Words to the French lemmas) and structural (from graph to tree) transfer from the UNL graph to an equivalent dependency tree (External Transfer module).
– Generation of the French sentence (ARIANE Generator).

3 Dependency trees equivalent to UNL graphs

Several cases are to be considered.

3.1 Graph with tree structure

The simplest case is when the graph has in fact a tree structure. The only difference between the graph and the tree is then that the semantic relations are attached to the arcs for the graph, to the target nodes for the tree. This is shown on Fig. 3.

For the sake of clarity, in this figure as in the following ones, restrictions and attributes are omitted. The entry node of a graph (or of a compound node) is indicated by a bold border.

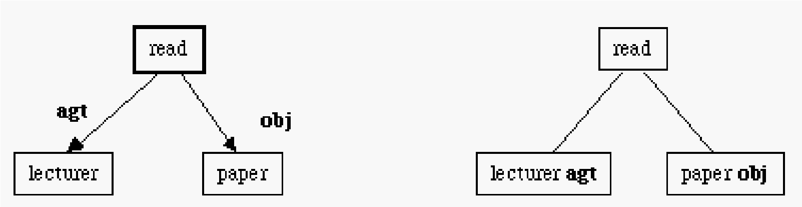


Fig. 3. A graph with tree structure (left), and its equivalent tree (right). *The lecturer reads a paper*

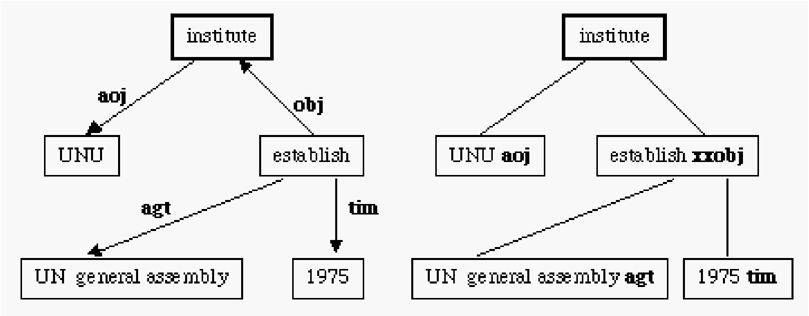


Fig. 4. A graph where the entry node has a mother node. The **obj** relation is inverted in the tree (**xxobj**). *UNU is an Institute which was established by the UN General Assembly in 1975*

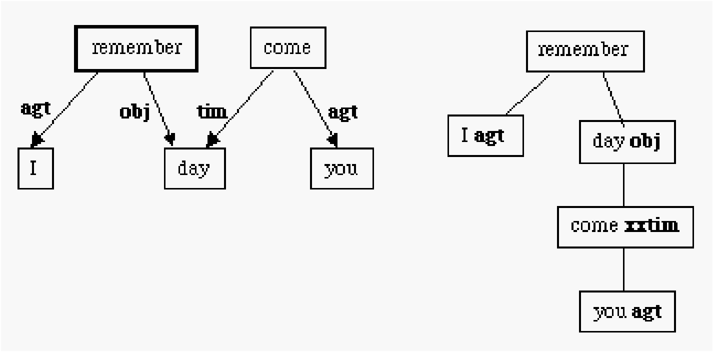


Fig. 5. The node "day" has two mother nodes in the graph. The **tim** relation is inverted in the tree (**xxtim**). *I remember the day where you came.*

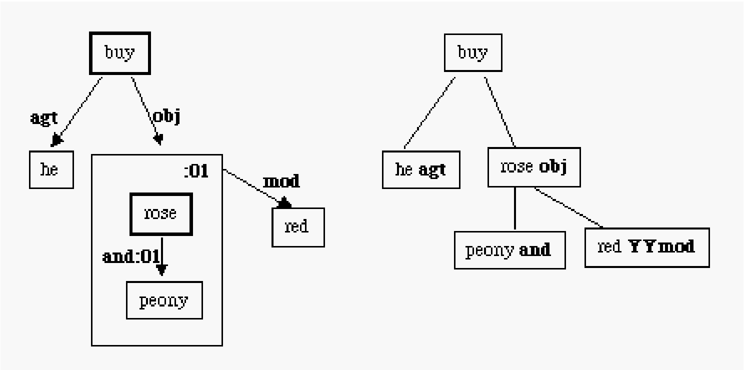


Fig. 6. A graph with a closed circuit and its equivalent tree. *The lecturer reads his paper.*

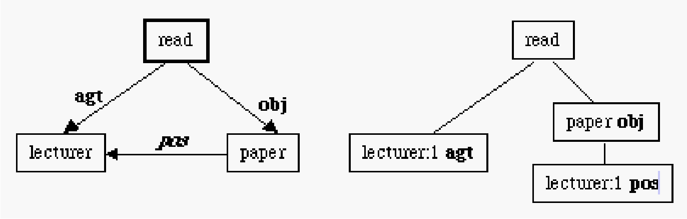


Fig. 7. A graph with a compound node. In the tree, the **yymod** relation of the "red" node indicates that the **mod** relation applies to the dependants of its mother node "rose" as a whole. *He buys red roses and red peonies.*

3.2 Graphs containing nodes having more than one mother node, or an entry node having a mother node

In a tree, the root node has no mother node, and the other nodes have only one mother node. This is of course generally not the case for a graph, where all the nodes (including the entry one) may have several mother nodes.

Let’s for instance consider the graph of fig. 4, representing the meaning of the sentence "The University of the United Nations is an Institute founded by the United Nations General Assembly in 1975". In this graph, the entry node (« institute ») has a mother node (« establish »), and the arc joining both nodes bears the *obj* relation: In order to get a tree structure, the direction of this arc is inverted, and the *obj* relation replaced by an "inverted *obj* relation" we denote by *xxobj*. The transfer into an equivalent tree is then straightforward. In the original graph, « institute » is the *obj* of establish, whereas what is expressed in the tree by the *xxobj* relation is that « establish » has « institute » as *obj*. Such an "inverted relation" is usually deconverted into French as a relative clause. The deconverted French text reads *"L'université des Nations Unies est un institut que l'Assemblée Générale des Nations Unies a fondé en 1975".*

Fig. 5 shows a graph where a node has two mother nodes. In the same manner, one of the arcs is inverted, and a *xxtim* relation replaces the *tim* relation. And again a relative clause will appear in the deconverted sentence *"Je me souviens du jour où tu es venu" ("I remember the day when you came")*

3.3 Graph containing a closed circuit

An equivalent tree structure of a graph containing closed circuits may be obtained by opening the circuits, splitting one of their nodes as shown on fig.6, where the node "lecturer" has been split into two nodes.

The new created node bears the same id number as the original one, indicating that it refers to the same object. In this example, this new node will be translated in French by the possessive "son" (its).

3.4 Graph containing compound nodes (scopes)

Fig. 7 shows a graph containing a compound node. The head :01 of the compound node does not appear in the corresponding tree. But the attributes and the dependants of the compound node as a whole are distinguished from the dependants and attributes of the entry node by specific variables, like *yymod* (for a node dependant of the scope) to be compared with *mod* (for a node dependant of the entry node).

4 Ambiguity and Anaphora Resolution

The problems of ambiguity and anaphora resolution have in principle not to be considered in the Deconversion process, a correct graph being unambiguous, and without any anaphoric pronouns.

They are on the other hand of course essential in Enconversion.

In the French Enconverter we develop, disambiguation is realised automatically, but we plan to introduce in the future interactive disambiguation using the methodology developed at Geta [2], and complete if necessary by a revision of the graph using a graph editor [3].

Anaphora resolution is interactive, as shown by the example of Fig. 8

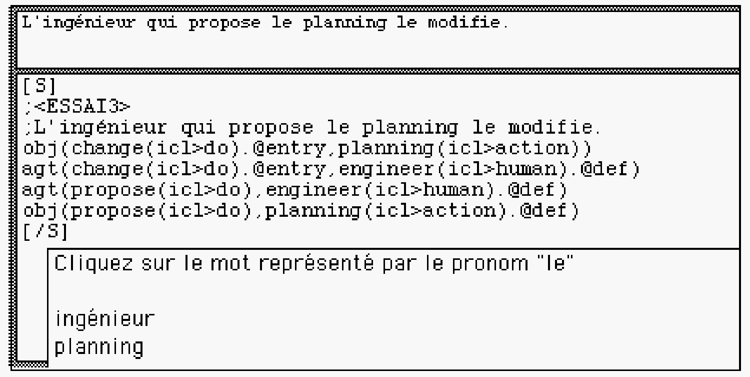
5 The Unknown Word

The problem of the unknown word arises as well in Enconversion as in Deconversion, but is generally more important for deconversion, where the user should use the deconverter as a black box providing the best result without any intervention.

Fortunately, in the case of deconversion, the very principle of UNL offers two means for deducing the part of speech of the target word associated to the UW of a given node. The first and most straightforward one is to deduce it from the UW restriction. The second one is to look at the relations in which the node is involved. For instance a node related to a daughter node by the *agt* relation, is very probably a predicate.

Both methods are implemented in our French Deconverter. The result is illustrated by the following example. The graph of fig. 9 contains 5 UWs, 4 of them corresponding to chemical terms unknown by our dictionaries. The structure of the deconverted sentence " Le? <<alcoylbenzene>> est obtenu en <<reduce>> le? <<group>> <<carbonyle>>." is nevertheless correct, and the sentence is comprehensible (The correct sentence would read "L'alcoylbenzène est obtenu en réduisant le groupe carbonyle"). The unknown words are represented by the headwords of the corresponding UWs put between <<>> marks. The question marks indicate that the articles could not be correctly calculated due to the lack of information about the gender.

Such a processing is particularly effective for technical texts where words are often similar in many languages.



L'ingénieur qui propose le planning le modifie.

```
[S]
;<ESSAI3>
;L'ingénieur qui propose le planning le modifie.
obj(change(icl>do).@entry,planning(icl>action))
agt(change(icl>do).@entry,engineer(icl>human).@def)
agt(propose(icl>do),engineer(icl>human).@def)
obj(propose(icl>do),planning(icl>action).@def)
[/S]
```

Cliquez sur le mot représenté par le pronom "le"

ingénieur
planning

Fig. 8. Anaphora resolution in the French Enconverter. The upper field contains the input text. The field in the middle the output graph. The lower field is the dialog window appearing during the Enconversion process.

```
  [S]
  obj(obtain(icl>do).@entry,alcoylbenzene.@def)
  met(obtain(icl>do).@entry,reduce(icl>do,field>chemistry))
  obj(reduce(icl>do,field>chemistry),group(icl>thing,field>chem
istry).@def)
  nam(group(icl>thing,field>chemistry).@def,carbonyle)
  [/S]
  Le? <<alcoylbenzene>> est obtenu en <<reduce>> le? <<group>>
<<carbonyle>>.
```

Fig. 9. An example of processing an unknown word.

6 Connection between nodes internal and external to a compound node

The question of the possibility of relating nodes external and internal to a given compound node seems not to be settled yet. There appears to be cases where this possibility would be very useful, if not necessary.

Let's consider for instance the left graph of figure 10. This graph is obviously ambiguous, it may express "The cat eats the mouse it caught" as well as "The cat which caught the mouse eats it". The ambiguity may be solved by introducing a compound node, but with the necessity of having an arc relating the predicate inside the compound node to its object or its agent outside the compound node.

Another possibility, avoiding arcs relating nodes inside and outside a compound node, is illustrated fig.11: the outer node is duplicated in the compound node, with the attribute @anaf indicating the peculiar nature of this duplicated node (it will often be deconverted into a pronoun).

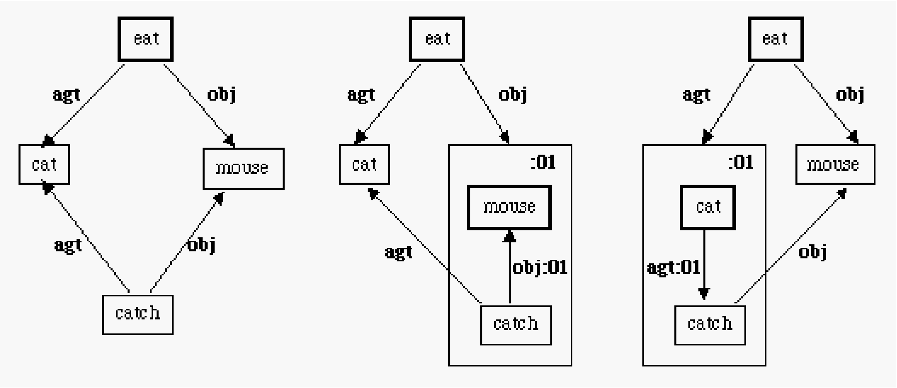


Fig. 10. The left graph is ambiguous: *The cat eats the mouse it caught / The cat which caught a mouse eats it.* The ambiguity may be solved using a scope with an arc emerging from it. The second graph expresses the meaning *The cat eats the mouse it caught,* the third one *The cat which caught a mouse eats it*

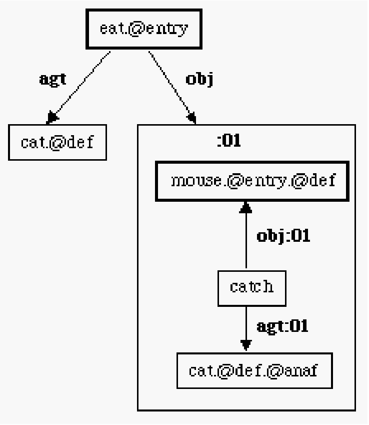


Fig. 11. Avoiding arcs connecting nodes internal and external to a compound node.

7 Evaluation and Conclusion

Evaluating the performances of a Deconverter or of an Enconverter is more difficult than evaluating a Natural Language MT system, which itself is well known to be a not so easy task.

The difficulty of evaluating an Deconverter lies in the fact that one has not only to devise the content of the test corpus, but to ensure the "linguistic" quality of this test corpus, which is of course not a problem for Natural Languages. The same applies for the evaluation of the output of an Enconverter.

As a result of several years of common work of the various UNL teams, an agreement about the correct use of the language is emerging. Nevertheless remaining discrepancies may influence the quality of the processing.

For instance, we had recently to deconvert two sets of graphs corresponding to the same source text, but encoded by two different teams. For a number of graphs, the quality of the output was not the same depending on their origin. Let's take two examples :

Example 1:

Source sentence :*The general conference adopts the universal declaration on cultural diversity*

Graph 1 :

```
agt(adopt(agt>thing,obj>thing).@entry,conference(icl>meeting))
obj(adopt(agt>thing,obj>thing.@entry,declaration(icl>information))
mod(conference(icl>meeting),general(mod<thing))
aoj(universal(aoj>thing),declaration(icl>information))
```

Graph 2:

```
agt(adopt(icl>accept(icl>do)).@entry.@present,conference(
icl>meeting).@def)
obj(adopt(icl>accept(icl>do)).@entry.@present,
declaration(icl>document).@def)
mod(conference(icl>meeting).@def, general(mod<thing))
mod(declaration(icl>document).@def, universal(aoj>thing))
```

Deconversion of graph 1: *Une conférence générale adopte une déclaration qui est universelle sur une diversité qui est culturelle.*

Deconversion of graph 2: *La conférence générale adopte la déclaration universelle sur la diversité culturelle.*

Comment : The quality of the deconversion of the second graph is quite better. The main problem lies here in the choice between the aoj relation (graph 1) and the mod relation (graph 2). We consider that the mod relation corresponds to an attributive use of the adjective, whereas the aoj relation corresponds to a predicative one. But the agreement seems to be not quite complete on this topic.

Example 2:

Source sentence *The general conference is aware of the specific mandate which has been* ***entrusted to UNESCO****, within the United Nations system, to ensure the preservation and promotion of the fruitful diversity of cultures*

Graph 1:

```
obj(aware(aoj>person,obj>thing).@entry,mandate(icl>authority).@def)
obj(entrust(agt>thing,gol>person,obj>thing).@complete,mandate(icl>author
ity).@def)
mod(mandate(icl>authority).@def,specific(mod<thing))
pur(entrust(agt>thing,gol>person,obj>thing).@complete,ensure(agt>thing,o
bj>thing))
man(entrust(agt>thing,gol>person,obj>thing).@complete,within(icl>how(obj
>thing)))
gol(entrust(agt>thing,gol>person,obj>thing).@complete,UNESCO(equ>United
Nations Educational, Scientific, and Cultural Organization))
obj(within(icl>how(obj>thing)),system(icl>functional thing).@def)
mod(system(icl>functional thing).@def,United Nations)
obj(ensure(agt>thing,obj>thing),:01.@def)
mod(:01.@def,diversity(icl>property).@def)
```

```
and:01(promotion(icl>activity).@entry,preservation(icl>state))
mod(diversity(icl>property).@def,culture(icl>abstract thing).@pl)
aoj(fruitful(aoj>thing),diversity(icl>property).@def)
```

Graph 2:

```
aoj(aware(mod<thing).@entry, conference(icl>meeting).@def)
mod(conference(icl>meeting).@def, general(mod<thing))
obj(aware(mod<thing).@entry, mandate(icl>authority).@def)
mod(mandate(icl>authority).@def, specific(mod<thing))
obj(entrust(icl>do).@complete, mandate(icl>authority).@def)
gol(entrust(icl>do).@complete, UNESCO(iof>institution).@def)
scn(entrust(icl>do).@complete, bosom(icl>abstract thing).@def)
pof(bosom(icl>abstract thing).@def, system(icl>abstract thing).@def)
pos(system(icl>abstract thing).@def, United
Nations(iof>institution).@def)
aoj(consist(icl>be), mandate(icl>authority).@def)
obj(consist(icl>be), ensure(icl>do))
obj(ensure(icl>do), promotion(icl>action).@def)
and(promotion(icl>action).@def, preservation(icl>action).@def)
obj(promotion(icl>action).@def, diversity(icl>abstract thing).@def)
mod(diversity(icl>abstract thing).@def, culture(icl>abstract
thing).@def.@pl)
mod(diversity(icl>abstract thing).@def, fruitful(mod<thing))
```

Deconversion of graph 1:

*La conférence générale est consciente du mandat spécifique qui est **confié à l'unesco** pour assurer la préservation et une promotion de la diversité de cultures qui est <fructueux> dans le système des <nations_unies>*

Deconversion of graph 2:

*La conférence générale est consciente du mandat spécifique qui est **confié pour l'unesco** dans le sein du système des <nations_unies> qui consiste que la préservation et la promotion de la diversité <fructueux> des cultures sont assurées*

Comment : We will only comment on the part of the graphs we printed in bold, corresponding to the words "***entrusted to UNESCO***" of the source text. Here the deconversion of graph 1 "***confié à l'Unesco"*** is more satisfactory than the deconversion of graph 2 "***confié pour l'Unesco"*** . The reason is that the restriction of the uw **`entrust(agt>thing,gol>person,obj>thing)`** indicates that a *gol* relation corresponds with a high probability to an argument of entrust, and not to a mere circumstantial. This allowed the enconverter to choose the right preposition. The uw **`entrust(icl>do)`** used in graph 2 didn't allow to choose the correct preposition.

But no doubt further cooperative work will soon smooth out the remaining difficulties in the use of the Universal Networking Language.

References

1. Blanc E, Sérasset G, Tsai W.J. *Structural and Lexical Transfer from a UNL Graph to an equivalent NL Dependency tree.* Proceedings of the First International Workshop on UNL, other Interlinguas and their Applications, LREC Conference. (2002)
2. Boitet, C. & Blanchon, H. *Multilingual Dialogue-Based MT for Monolingual Authors: the LIDIA Project and a First Mockup.* in Machine Translation. (1995). Vol. 9(2) : pp 99-132.
3. Tsai W.J. *La coédition Langue <-> UNL pour partager la révision entre langues d'un document multilingue.* Thèse d'Université. Université Joseph Fourier, Grenoble (2004).

A UNL Deconverter for Chinese

Xiaodong Shi, Yidong Chen

Institute of Artificial Intelligence
School of Information Sciences and Technologies, Xiamen University
361005 Xiamen, China
{mandel, ydchen}@xmu.edu.cn

Abstract. This paper describes the internal working of a novel UNL converter for the Chinese language. Three steps are involved in generating Chinese from UNL: first, the UNL expression is converted to a graph; second, the graph is converted to a number of trees. Third, a top-down tree walking is performed to translate each subtree and the results are composed to form a complete sentence. Because each node is visited exactly once, the algorithm is of linear time complexity and thus much faster than the standard deconverter provided by the UNL center. A manual evaluation effort was carried out which confirmed that the quality of the Deconverter output was better than that of the standard deconverter.

1 Introduction

Although the UNL [1],[2] center provides a language independent generator [3] which can deconvert UNL expressions into any language provided that a UW dictionary, a set of deconversion rules, and optionally a co-occurrence dictionary are available for that language, that deconverter has a number of deficiencies: First, the deconversion rules are rather difficult to write because of the cryptic formats imposed by the deconversion specification. Second, although the power of the deconverter is claimed to be that of the Turing machine [4], its speed is rather slow and thus unsuitable for the main web application, embedded multilingual viewing of a UNL document that is one of the key goals of the UNL. Third, most importantly, the deconversion software is not open-sourced, so that fixing any bugs or introducing much-needed improvements is at the mercy of the UNL center, which has been rather lacking in technical support and in releasing new versions. So we think it is necessary to develop our own deconverter for Chinese. This paper describes such an endeavor. However, it should be noted that although we concentrate on generating Chinese from the UNL expressions, nothing in our deconverter is inherently related to Chinese, thus the deconverter is also language independent.

This paper is organized as follows: Section 2 will describe the main components of the deconverter and the algorithms involved. Section 3 will focus some issues in generation, especially those related to the Chinese language, and in Section 4 we will briefly discuss related work in the literature. In Section 5 we will give example uses of the deconverter and finally we will present the conclusions.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 167–174.

2 The Main Components of the Deconverter

The deconverter has three components: the first converts a UNL expression into a graph, the second breaks the graph into a number of trees, and in the third, a recursive top-down tree walking is performed to translate each subtree and the results are composed to form a complete sentence. They will be described in more detail below.

2.1 Graph Construction

Converting a UNL expression into a graph is straightforward. In this respect, the list form of a UNL expression is simpler to convert than the normal form:

```
{unl}
[W]
language(icl>abstract thing).@present.@entry:00
UNL(icl>language).@topic:01
common(aoj>thing):02
use(icl>do).@present:03
communication(icl>action).@pl:05
network(icl>thing):06
[/W]
[R]
00aoj01
00mod02
03obj00
03pur05
05mod06
[/R]
{/unl}
```

Fig. 1. The list form of a UNL expression.

In essence, the normal form is converted into the list form, from which a *directed* graph is constructed. The nodes correspond to UWs in the UNL expression and the edges of the graph are labeled with relations, pointing from head to the dependents (if a relation is of the form UWID1 **rel** UWID2, then UWID1 is the head, and UWID2 is the dependent). In the case of a compound UWID, a node corresponding to it is also created because it can have attributes attached, e.g. some of the attributes from the ITU corpus are @def, @topic, @double_slash, etc, which usually apply to all the UWs in the scope denoted by the compound UWID. It should be noted that two nodes with different IDs but otherwise identical information cannot be collapsed into one, as co-referential nodes share the same ID (or use a UW with no ID at all). One node of the graph is of central importance: the entry node.

2.2 Graph-to-tree Conversion

The second component is called the graph-to-tree conversion. If a node in the graph has **two** or more edges (or **one** or more edges for the entry node) pointing to it, then all the directed edges except one must be severed. Exactly which one is retained depends on a breath-first traversal of the graph starting from the entry node. See the following paragraph for the detailed discussion of this operation. The edges encountered in this traversal are called the forward edges and other edges will be severed, but the traversal will continue at the head of the severed edges. For the UNL expression shown in Fig. 1, the graph conversion will produce the following two trees:

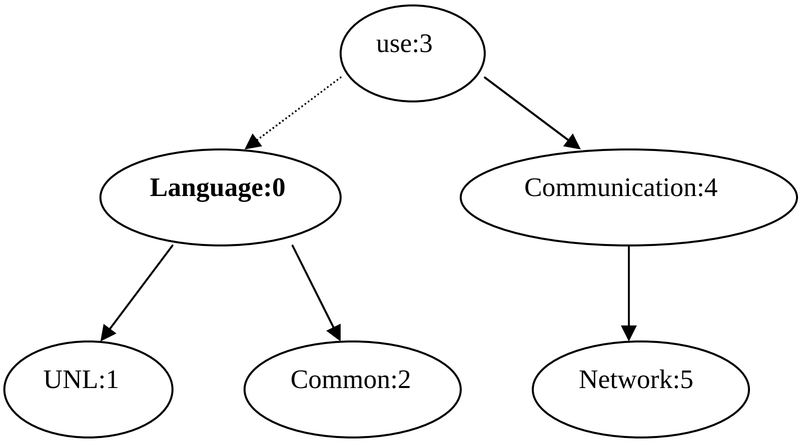


Fig. 2. The two trees converted from the graph. The severed edges are in dotted line (from node 3 to 0) and the node number indicates the graph traversal order. The edge is severed because node 0 is the entry node with an in-degree of 1.

There are two cases in the handling of a node with a severed link depending on whether it is **duplicated** or **attributive**. (The latter means that it is modified by an attributive clause, or in other words, it is an entry node of a scope, but with a governing head). And the reason for choosing this is somewhat **syntactic**. In our opinion, the introduction of the scope node also has a very strong syntactic flavor, besides making UNL more hierarchical. In general, an underspecified UNL graph can mean different things to different people when syntactic information in the original sentence is not represented with a scope node (or a Compound UW)[1]. The following UNL graph illustrates this point (slightly adapted from [12]):

[1] This observation benefited from a discussion with Dr Etienne Blanc while I was visiting GETA-CLIPS in 2004.

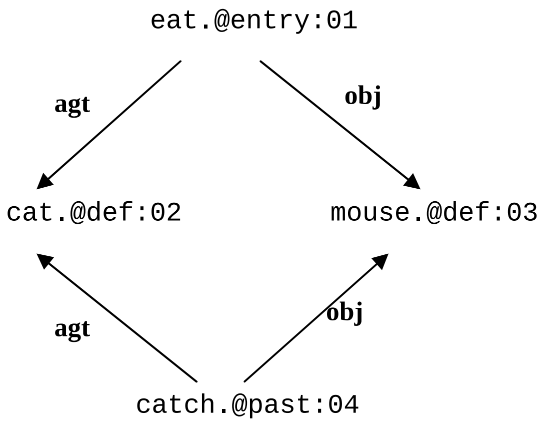


Fig. 3. A UNL graph for either “The cat eats the mouse it caught” or “The cat which caught the mouse eats it”.

When node 02 is duplicated and node 03 is made attributive, we could get the deconverted result: “The cat eats the mouse it caught” (or “the cat eats the mouse the cat caught” if no pronominalization is implemented); when node 02 is made attributive and node 03 is duplicated, we could get the deconverted result: “The cat eats the mouse it caught”. (The third possibility: “The cat eats the mouse. The cat caught the mouse.” is not implemented, as might be the result when both nodes 02 and 03 are duplicated)

In the implementation, the severed edges are still retained in the graph, and this *conversion* is done on the fly just before generation, for optimal performance.

2.3 Natural Language Generation from the Trees

A simple hypothesis of the generation algorithm is the **compositionality**. It states that the generation of the whole tree can be constructed from its subtrees. Suppose a node V has the set of adjacent nodes: A(V), and let T be a function from the subtree dominated by V to its translation, and C be a composition function from the *subpart*s (translation of the subtree, henceforth), then we have

$$T(V)=C(W(V), T(V_1), T(V_2), T(V_3),...),\quad \cup V_i = A(V) \qquad \textbf{(1)}$$

where W(V) is the proper word translation of the node V.

There are several things to note during the composition process. First an ordering function **O** determines the relative orders of the subparts according to the roles played by them. Relation Labels (RL) on the tree edges provide most of the information. Then the particular choice of the head word translation W(V) may have its own subcategorization requirement and ordering constraints. So the ordering function can be specified as follows:

$$\mathbf{O}\text{：} W(V), RL_1, RL_2,\ldots RL_n, \rightarrow I^{n+1} \qquad \textbf{(2)}$$

where I is the set of integers. After the orders are determined, the subparts are simply concatenated.

Second some words (*glue* words) may have to be prepended or appended to the subparts to make case roles explicit. This is the case for Chinese. Other languages, e.g. Tamil [5], may require morphological generation. The generation algorithm can easily handle these minor divergences.

Note that translations produced by the severed nodes are also composed. These are mostly realized as attributive clauses. The node is aware of whether it is generating standalone, or as a severed subtree.

A simple way of avoiding loops in generation (because the tree conversion is a logical process) and gathering information from the already generated nodes is to mark each node as they are visited.

The deconverter is very impressive in its speed. Because each node is visited exactly once, the algorithm is of linear time complexity and thus much faster than the standard deconverter provided by the UNL center. As to the quality of the deconverter, a manual evaluation effort of the translations of a few hundred sentences was carried out, which confirmed that the quality of the output was also superior to that of the standard deconverter.

3 Some Issues in Generation

We should note that in general, the compositionality hypothesis is not always correct. For examples, in some idiomatic constructions, one subpart has to be imbedded in another subpart. And although UNL encodes the natural language sentences in a language neutral way, its particular choice of UWs is inevitably influenced by the English language, which is used as a specification language to make UNL accessible to a larger audience. So some UWs may not have appropriate lexicalization in another language and so awkward translations may result.

One particular pitfall in UNL is the specification of the conjunction relation: the UNL expression is ambiguous as to whether a conjunction is at sentence level or predicate (mostly verbal or adjectival concept) level. The correct function must be inferred from other relation label which may be present. Other relation labels may also present problems, as there have been lots of argument among the language centers as to whether a particular relation is need, but as long as UNL expressions are consistent in this respect (admittedly a hard goal to attain), no serious problem should follow.

One important aspect of the Chinese generation is the insertion of appropriate classifier, or measure word, for nouns that can be counted. This has been notoriously difficult to handle in the original rule set developed for the standard deconverter of the UNL center, and more than one hundred rules are given, each for a different classifier. In our implementation, the classifier is treated as a *glue* word, and is generated directly from the head noun using a classifier table which can be modified separately.

Some UWs expressing prepositions in English are problematic in generating correct Chinese. They are typically realized into two Chinese words surrounding the governed noun phrases. To circumvent this problem, we introduced the concept of a *parametric* attribute, which share a common prefix. One word of the Chinese translation is chosen as the main word which corresponds to the UW, the other word is expressed using a *parametric* attribute (concatenating the prefix with the second word). In generation, when a *parametric* attribute is found to be present, the second word is extracted and properly appended.

To resolve the problem of ambiguity in lexicalization (as is the case for near-synonyms), a co-occurrence dictionary is also used, but these data are collected automatically from a huge Chinese corpus with more than 8 billion Chinese characters. Since automatic parsing for Chinese is still not very reliable, we only extract bigrams using **segtag**, a segmentation and tagging program developed by the Center for Language Technology[2] of Xiamen University, which can be downloaded for free from http://clt.xmu.edu.cn.

4 Related Work on UNL Deconverter

Although there are 15 language centers listed on the UNL Universe [6], only a few deconverters are working, e.g. the Russian, French, and Spanish ones. And the details on their implementation are scanty and scarce.

[7] describes a UNL to French deconverter which also utilizes a graph to tree converter. The tree output is then fed into an Ariane-G5 transfer program to reuse the MT facility. The generation is much more complex than described here.

[5] describes the UNL to Tamil deconverter which focus more on the syntactic and morphological generation**.**

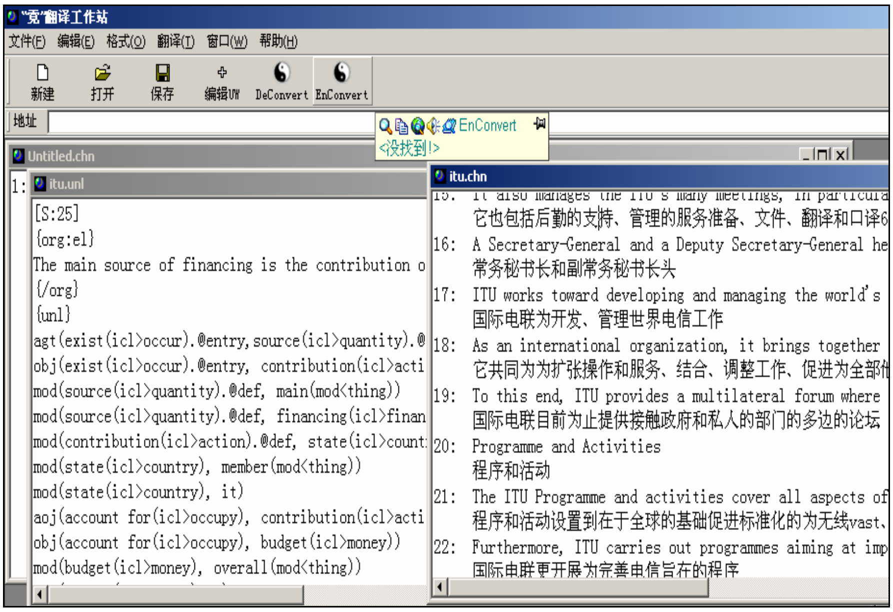
[8] describes a deconverter for Chinese. That work was done by the Chinese center before the authors came to work there. It is because of the deficiency of that deconverter that the need for a new one is called for.

The main characteristics of the deconverter described here is its simplicity, speed and effectiveness. We think that it can be applied to other natural languages to the equal benefit exhibited by Chinese and so we make it available for download from our website http://ai.xmu.edu.cn/. Interested parties can contact the authors for the source code.

5 Deconverter in Use

The deconverter is developed on Microsoft Windows platform. We have built an IDE integrating enconversion, deconversion and UW editing. [9] describes the technical aspects of both the deconverter and enconverter in more detail. The following figure shows the IDE interface.

[2] The Institute of Artificial Intelligence is a major research branch of the Center for Language Technology, which also includes faculty from the Humanities departments.



```
1: [S:25]
{org:el}
The main source of financing is the contribution o
{/org}
{unl}
agt(exist(icl>occur).@entry,source(icl>quantity).@
obj(exist(icl>occur).@entry, contribution(icl>acti
mod(source(icl>quantity).@def, main(mod<thing))
mod(source(icl>quantity).@def, financing(icl>finan
mod(contribution(icl>action).@def, state(icl>count
mod(state(icl>country), member(mod<thing))
mod(state(icl>country), it)
aoj(account for(icl>occupy), contribution(icl>acti
obj(account for(icl>occupy), budget(icl>money))
mod(budget(icl>money), overall(mod<thing))
```

```
15:  It also manages the ITU's many meetings, in particula
     它也包括后勤的支持、管理的服务准备、文件、翻译和口译6
16:  A Secretary-General and a Deputy Secretary-General he
     常务秘书长和副常务秘书长头
17:  ITU works toward developing and managing the world's
     国际电联为开发、管理世界电信工作
18:  As an international organization, it brings together
     它共同为为扩张操作和服务、结合、调整工作、促进为全部
19:  To this end, ITU provides a multilateral forum where
     国际电联目前为止提供接触政府和私人的部门的多边的论坛
20:  Programme and Activities
     程序和活动
21:  The ITU Programme and activities cover all aspects of
     程序和活动设置到在于全球的基础促进标准化的为无线vast、
22:  Furthermore, ITU carries out programmes aiming at imp
     国际电联更开展为完善电信旨在的程序
```

Fig. 3. The UNL enconversion and deconversion IDE.

We have also built a language server [10] (not in the sense of the UNL center), which is implemented as a SOAP-compliant Web Service [11]. The server runs as a Apache module and effectively offers a cross-platform Remote Procedure Call (RPC) for enconversion and deconversion. It also provides an RPC endpoint for MT via UNL. If other enconverters or deconverters were available, MT would be available for those languages.

6 Conclusions

This paper describes our implementation of a UNL deconverter for Chinese with graph construction, graph-to-tree conversion, and recursive top-down generation as three components. It's very fast and gives better performance than the standard deconverter with a Chinese rule set. Although it is developed with Chinese in mind, it is also a language neutral software and thus can be used for other languages.

Acknowledgements The work described in this paper was mostly done while the authors were in the Chinese Language Center, the Graduate School of the Academy of Chinese Sciences, in the year 2001. The authors wish to express thanks to Professor Wen Gao, the director of the Chinese Language Center, and Dr Hiroshi Uchida, of the UNL Center in Tokyo, for the support provided.

We would also like to thank the anonymous reviewer for very constructive comments on the first version of this paper.

This work is partially sponsored by China high-tech program grant No. 2001AA114103 and grant No. 2002AA117010.

References

1. UNU/IAS/UNL Center: The Universal Networking Language (UNL) Specifications 3.0, August 2000
2. Martins R.T., Rino L.H.M., Nunes M.G.V., Montilha G., Oliveira Jr. O.N.: An interlingua aiming at communication on the Web: How language-independent can it be? Workshop on Applied Interlinguas: Practical Applications of Interlingual Approaches to NLP. April 30, 2000. Seatle, Washington, USA.
3. UNL Center, Deconverter Specifications, Version 2.5, UNL-TR-2001-001
4. http://www.undl.org/unlsys/public/deco.html
5. Dhanabalan T., Geetha T.V.: UNL Deconverter for Tamil. International Conference on the Convergences of Knowledge, Culture, Language and Information Technologies, December 2003, Alexandria, EGYPT
6. UNL Resources by ISI, http://cui.unige.ch/isi/unl/
7. Blanc É. & Sérasset G.: From Graph to Tree: Processing UNL Graphs using an Existing MT System. Proc. The first UNL open Conference, Suzhou, China, 22-26 November 2001
8. Xiong, W.X: Some Problems on Machine Translation via Interlingua, Applied Linguistics. 3 (1998) 69-75
9. Shi X.D.: UNL enconversion and Deconversion Based on the Neon MT system. Technical Report, Chinese Language Center, 2002.
10. Shi X.D.: The UNL Language Server for Chinese, Chinese Language Center, 2002.
11. Curbera F., Nagy W., and Weerawarana. S.: Web services: Why and how. In Workshop on Object Orientation and Web Services OOWS2001, 2001
12. Sérasset G., Blanc E.: Remaining Issues that could Prevent UNL to be Accepted as a Standard, Convergences 2003, Alexandria, Egypt, 26 December 2003.

Flexibility, Configurability and Optimality in UNL Deconversion via Multiparadigm Programming

Jorge Marques Pelizzoni, Maria das Graças Volpe Nunes

Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação
Av. do Trabalhador São-Carlense, 400. CEP 13560-970. São Carlos – SP – Brasil
{jorgemp, gracan}@icmc.usp.br
http://www.nilc.icmc.usp.br

Abstract. The fulfillment of the UNL vision is primarily conditioned on the successful deployment of deconverters, each translating from the UNL into a target language. According to current practice, developing deconverters ultimately means configuring DeCo, the deconversion engine provided by the UNDL Foundation. However, DeCo has a number of limitations that hinder productivity and might even preclude quality deconversion. This paper discusses some of these shortcomings and introduces an alternative deconversion model – Manati, which is the result of work on UNL-mediated Portuguese-Brazilian Sign Language human-aided machine translation. With Manati we attempt to exemplify how multiparadigm – namely, constraint, object-oriented and higher-order – programming can be drawn upon not only to specify an open-architecture, optimum-searching deconversion engine but also and above all to rationalize its configuration into deconverters for target languages.

1 Introduction

The fulfillment of the UNL vision [10, 11, 18] is primarily conditioned on the successful deployment of deconverters, each translating from the UNL into a target language. UNL deconversion is actually an instance of Natural Language Generation (NLG), which refers to rendering linguistic form to input in a non-linguistic representation. As pointed out by e.g. Reiter & Dale [13], Cahill & Reape [3], and Paiva [12], NLG can be a very complex task involving processing both linguistic (e.g. lexicalization, aggregation and referring expression generation) and otherwise (e.g. content selection and layout planning). The good news is that UNL deconversion is in fact restricted to the linguistic aspect of NLG, which can be termed **linguistic realization** and comprises the usual macro-level tasks of microplanning and surface realization. Therefore, one should naturally expect UNL deconversion to benefit from recent advances in Natural Language Generation and software development practice, for which reason UNL developers may need to go beyond the model underlying the DeConverter – or simply DeCo, the generic deconversion engine provided by the UNDL foundation.

In this paper we analyze DeCo both as a formal object and a software product, with an emphasis on discussing DeCo's features that may hinder productivity. In this analysis we adopt configurability (i.e. ease of configuration into full-fledged decon-

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 175–194.

verters), flexibility to accommodate application-specificities and support for optimality (i.e. search for optimal solutions) as meta-requirements for an ideal deconversion model. As a first attempt to meet these requirements and overcome DeCo's limitations, we conceived Manati, an alternative linguistic realization/UNL deconversion model. Manati exemplifies how multiparadigm – namely, constraint, object-oriented and higher-order – programming can be drawn upon not only to specify an open-architecture, optimum-searching deconversion engine but also and above all to rationalize its configuration into actual deconverters for target languages. In order to present important aspects of Manati's rationale, we introduce LIBRAS, the Brazilian Sign Language, and PUL∅, a UNL-mediated Portuguese-LIBRAS machine translation project, as it was PUL∅ that provided (i) the opportunity to experiment with DeCo and (ii) specificities that promptly exposed its limitations.

The paper proceeds as follows. Section 2 shortly introduces DeCo to the unacquainted; Section 3 states the LIBRAS case and introduces LIST (LIBRAS Script for Translation), a notation employed in some examples; Section 4 discusses DeCo's limitations; and Section 5 briefly describes Manati.

2 Meet DeCo

In this section, we review only those features of DeCo's which are essential to our discussion, i.e. just enough to illustrate how most effort is expended in DeCo's application. This is a very simplified overview especially to cater for the unacquainted with DeCo's abstract machine. For a thorough description, please refer to the *DeConverter Specifications* document provided by the *UNL Centre/UNDL Foundation*. It is worth mentioning that the terminology used in this section slightly differs from that of the referred document.

2.1 Configuration

In order to configure DeCo, i.e. prepare it to translate UNL hypergraphs into text in a specific target language, one must feed it with at least two basic *language-specific* resources, namely a UNL-target language **dictionary** and an ordered set of **deconversion rules**.

In short, each dictionary entry has a twofold function: (i) to declare a possible mapping of a UW[1] *Src* into a target language word or morpheme *Target*[2] and (ii) to state a set of **atomic (i.e. non-structured) features**[3] that should be assumed for *Target* whenever the declared mapping happens to be used. For example, supposing one intended to state that the UW "I" should be translated into English "I", "me", "my" or

[1] UWs (Universal Words) are UNL words. Formally, they are possible node labels in UNL graphs.

[2] Though rather unusual, *Target* might also be an intermediate symbol later to be erased.

[3] The term "feature" is herein employed much in the grammatical sense. In computer jargon, "flag" would be more appropriate.

“mine” under mutually exclusive conditions, there would usually be at least four distinct dictionary entries, as shown in **Table 1**. It is worth mentioning that developers are free to design their own set of possible features, as well as their respective meanings. The developers of the entries in **Table 1** seem to have found it interesting to encode the grammatical cases each English pronoun can accept (by means of features SUBJ and OBJ), parts-of-speech (PRO and DET), and person-cum-number information when needed (1PS, 3PS, and 3PP). Nevertheless, they might as well have found it more convenient e.g. to split the latter into independent features, some for person (1P and 3P) and others for number (PLU and SING), the only actual requirement being consistence. Finally, it should be noticed that a feature set belongs to the entry/mapping, not to the target word proper, as one would expect e.g. English “mine” to have a rather different feature set were the source UW “mine(icl>source)”.

Table 1. Example UNL-English dictionary entries mapping the UW “I” into “I”, “me”, “my” or “mine”

UW	English	Features
I	I	{SUBJ, PRO, 1PS …}
I	me	{OBJ, PRO, 1PS …}
I	my	{DET, 3PS, 3PP …}
I	mine	{SUBJ, OBJ, PRO, 3PS, 3PP …}

In turn, the set of deconversion rules specify exactly *how* deconversion should be carried out, including when to access the dictionary. These rules are intrinsically procedural and somehow encode the grammar of the target language in terms of operations (sensing/writing/erasing) on features. Deconversion rules can only be correctly understood with DeCo’s abstract machine in mind.

2.2 Abstract Machine

Roughly speaking, DeCo can be regarded as a non-deterministic Turing Machine fitted with some dictionary and graph lookup facilities. Its output is gradually built on an **extensible/retractable tape**, initially empty, by a pair of **ever-contiguous read/write (RW) heads**. At the end of deconversion, the sequence of tokens on the tape is printed out verbatim from left to right, and that should constitute the target text. However, if the tape were a mere string of tokens, it would have been of little use. As depicted in

Fig. 1, it is actually a string of cells each containing not only one single output token but also control data in the form of a rewritable feature set and *usually* a UW and a set of relations to nodes in the input UNL graph. The UW and relations can only be present if the cell results from the **transference** of an input node onto the tape, and those are precisely the UW of the referred node and all **unexplored** relations it has to other input nodes. A node is selected for transference by means of a relation it has to some **focused cell** (i.e. currently under one RW head); if the transference succeeds, then the referred relation is said to have been *explored.* Node transference is illustrated in **Fig. 2** and further explained below.

Much like bits of a Turing Machine's transition relation definition, deconversion rules are always relative to the pair of RW heads. Among other things, every rule may specify some of the following: (i) **preconditions** on the features, tokens or UWs of several existing cells (the two focused cells and variable-length sequences of cells on each side of and even *between*[4] the heads); (ii) similar preconditions on a new potential cell to be inserted; if this potential cell should result from the transference of a node, (iii) the label of the relation this node must have to one specific focused cell; (iv) features to be removed from or added to each focused cell (possibly a newly-inserted one); (v) whether to delete or replicate one specific focused cell; and, if the rule is not trying to insert or delete a cell, (vi) whether the heads should jointly slide one position to the left or right on completion. Naturally, a rule is applicable iff the preconditions (i) to (iii) are satisfied, and the actions of a rule are put into effect only if it is applicable and is actually selected for application.

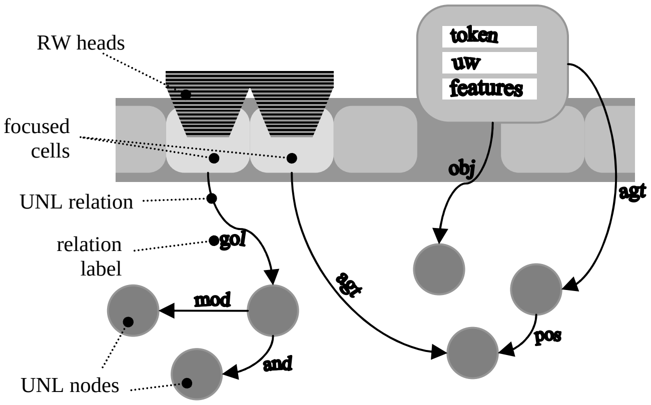


Fig. 1. Cells on DeCo's output tape

Two types of deconversion rules are of special interest to our discussion, namely:

- **feature-modifying rules**, which simply add or remove features to focused cells and optionally move the heads one position right or left on the tape. Features not only encode linguistic and conceptual information of the corresponding target language tokens, but also function as symbols in Turing Machines and much too often are used to simulate global states, implementing the rudiments of subroutines. For example, suppose there is the following top-priority set of rules (listed in order of priority. Notice this is not DeCo's actual notation):

```
if read(right, REWIND) and read(left, LEFT_DELIMITER)
   then erase(right, REWIND);
if read(right, REWIND)
   and not(read(left, LEFT_DELIMITER))
```

[4] This feature is only available to node-inserting rules and is explained later, when we tackle node-transferring rules, a specialization thereof.

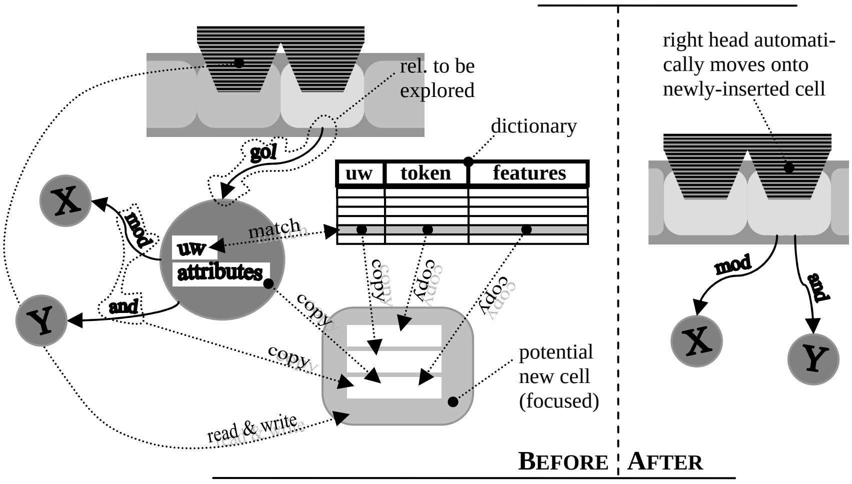


Fig. 2. Before and after node transference

```
then erase(right, REWIND),
     write(left, REWIND), move(left);
if read(left, REWIND) then move(left);
```

where `read(H,F)` is true iff head `H` can read `F` among the features of its respective focused cell, `write/erase(H,F)` adds/removes feature `F` to/from the cell focused by head `H`, and `move(D)` makes the heads move jointly one cell towards direction `D`. In this situation, a lower-priority rule writing REWIND roughly corresponds to calling a subroutine that will take the heads to the left end of the tape;

- **node-transferring rules**, which can also change the features of one focused cell, but whose most relevant effect is transferring one node from the UNL graph onto the tape and consequently expanding it. This is an all-important moment during deconversion as it is when the transferred node is "amalgamated" with one of its possible translations in the dictionary according to its UW, creating a new cell whose UW and output token are directly copied from the corresponding dictionary entry and whose feature set is initialized with (i) the features found in the entry, (ii) features homonymous to the UNL attributes found in the transferred node and (iii) features informative of the relations of the transferred node (e.g. a feature >*R* or <*R* indicates that the node plays respectively the left or right role in a relation labeled *R*). This process is depicted in Fig. 2. As an insertion rule may place preconditions on the feature set, token and UW of potential new cells, which are computed as just described, it follows that it is possible not only to state preconditions on the UNL attributes and relations of candidate nodes but also filter acceptable dictionary entries at all times.

One particularity of node-transferring rules, as well as node-inserting rules in general, is that the actual landing site of the new cell is not necessarily next to the **anchor cell** (i.e. the preexisting focused one, which provides the relation to be explored).

These rules may also specify a sequence of cells that should be present between the anchor cell and the cell to be inserted. Any such sequence is said to be *between the heads*, which is a very ephemeral state, as the heads again become contiguous right after insertion. It is exactly the mentioned particularity that allows e.g. the generation of discontinuous constituents, like the underlined subject in "A law was enacted during the previous administration that would require factories to reduce their emission of air pollutants by 70% over the next 3 years."

2.3 Nondeterministic Execution

DeCo executes nondeterministically in that it often reaches **choice points**, at each of which it has to choose from a priority-ordered set of alternative execution branches. DeCo then selects the highest-priority one to continue, but keeps track of the alternatives so that it may **backtrack** on failure. Backtracking consists of rolling execution back to the latest non-exhausted choice point, taking the highest-priority alternative branch not yet attempted and thus resuming execution. Unrecoverable failure, i.e. output consisting of an error message, arises from the exhaustion of all combinations of choices or usually timeout, due to combinatorial explosion. Success is restricted to the first good guess in depth-first search. Choice points are created whenever (i) there is more than one applicable rule or, during application of an insertion rule, (ii) there is more than one acceptable dictionary entry and/or eligible node for transference. It is worth mentioning that, whether implicitly or explicitly, developers statically stipulate the priority of every dictionary entry and deconversion rule.

Given one UNL graph as input, a configuration of DeCo tries to produce text in the target language of choice as follows: (i) DeCo starts with a tape containing only two predefined delimiter cells; (ii) it regularly transfers the entry node to the input graph onto the tape, between the delimiters; (iii) the right head is placed over the newly-inserted cell; (ii) DeCo iteratively applies rules until the right head tries to trespass the right end of the tape, the only sign of success; (iii) whenever DeCo gets stuck for lack of applicable rules, it tries to backtrack, unrecoverable failure arising from lack of non-exhausted choice points.

3 LIBRAS Testifies

This paper is one by-product of a very first attempt at Portuguese-LIBRAS[5] machine translation. This project is still under development but has provided enough opportunity to experiment with DeCo and put forth and implement the first draft of Manati, our alternative deconversion model. Naturally, neither DeCo nor Manati is ever intended to produce actual LIBRAS speech[6], but a script thereof – LIST (LIBRAS

[5] LIBRAS is an acronym for "LÍngua BRAsileira de Sinais", which is Portuguese for "Brazilian Sign Language".

[6] The words "spoken", "speech", etc. are employed here especially as opposed to "written", "writing", etc. Specifically, those words should not be regarded as necessarily implying *oral-*

Script for Translation) – to feed an eventual speech synthesizer. LIST is still in its infancy and shall be the result of a compromise between simplification of the translation apparatus and sufficiency for final synthesis.

To avoid some frequent misconceptions, it is worth reminding that sign languages are full-fledged languages on their own and usually rather dissimilar to their national oral counterparts. In fact, oral languages usually regarded as very different and thus translation–hard become in many respects closely related when sign languages come into scope. For the specific pair at issue, Portuguese and LIBRAS, we find that translation can be at times much harder than between Portuguese and English, for example. We shall present some evidence of this later, but plenty can be found elsewhere, as in Speers [15] and Brito [2]. Another point worth mentioning is that Linguistics have lately dedicated more and more attention to the subject, and most concepts and terms originally coined for the analysis of oral languages – such as "word", "phonetics", "phonology", "morphology", "syntax", and "prosody" – naturally apply and have been applied to sign languages as well.

LIBRAS is profligate in especially challenging problems for translation/deconversion and thus compelling examples, as several of Manati's features are thereby motivated. Consequently, an informal introduction to LIST is in order, so that LIBRAS examples can be presented.

First of all, LIST should not be expected to be readable by end-users. It is an interface protocol between two software modules: a translator and a speech synthesizer. If humans are ever to understand it, those must be the developers of those modules. Currently, LIST is biased towards ease of translation and, as much as possible, tries to approximate LIBRAS sentences with lists of **logograms**[7]. For the sake of readability, each logogram will herein be represented by a blank-delimited English string suggestive of its meaning in LIBRAS. For example, we show three LIST logograms below:

```
cut-with-a-knife old-man closed
```

LIST allows for tree-like structuring by means of **prosodic groups**, which are square-bracketed lists of logograms or other prosodic groups. This construct is to be used wisely and is just meant to include annotations strictly required by synthesis. The current prescriptions for prosodic group usage are beyond the scope of this paper, it sufficing to mention that every LIBRAS sentence is itself a prosodic group. Therefore, the following are examples of LISTified LIBRAS sentences: `[water still mosquito be-born grow]` ("Mosquitoes are born and grow in still water.") and `[I say water dangerous]` ("I told you water is dangerous.").

Finally, both logograms and prosodic groups as a whole may have associated **attribute-value matrices (AVMs)**, which allow e.g. (i) adding inflectional data to the logograms of the few inflected words of LIBRAS and (ii) annotating prosodic groups with the relevant prosodic information. An AVM is a curly-bracketed list of *Attrib-*

ity. In the case of sign languages, for example, speech synthesis involves actually moving an artificial communication actor, either by means of computer graphics or robotics.

[7] Each logogram is an atomic (i.e. non-analysable), strictly non-phonologically motivated symbol standing for a word. Chinese characters are examples of logograms; and, much though one can identify smaller component logograms inside a bigger Chinese character, the meaning of the latter can never be deduced from the former.

ute:Value pairs and is attached to the adjacent logogram or group to its left. When an *Attribute* is given without a value, *Attribute:true* is implied. For example,

```
[ ask{subjpers:2ps objpers:3pp} ]{imperative}
```

represents a one-word LIBRAS sentence meaning "Ask them!" and implies that LIBRAS `ask` agrees in person with its subject and object simultaneously, which are lexically absent in this sentence. Again, there are strict specifications ruling AVM usage, but they do not need to be covered here. It suffices to mention that, in addition to `subjpers` and `objpers`, attributes `subjgend`, `objgend` and `objlidgend` shall be used in examples and imply gender agreement of a verb with its subject, object and lid of its object (!), respectively.

4 DeCo Exposed

The craft of programming DeCo requires clockwork precision. Correct deconversion can be summed up as scheduling the transference of nodes with accuracy and handling cell features at the right times, since all things are global, flat, transient and public. Precise prioritizing of rules is the key. For example, supposing a verb must be preceded by its subject and object in that order, it follows that:

1. *subject insertion rules* must have priority over those for *object insertion*, as subject and object source nodes usually have direct UNL relations to verb cells;
2. the moment just after the insertion of a subject, in which it is adjacent to its verb, is the opportunity to solve whatever matters of agreement between them by means of rules that add specific features to the cell of the verb according to features they read in the cell of the subject. These *agreement rules* must thus have priority over object insertion;
3. agreement rules must read the right subject features; therefore, just in case e.g. we are dealing with a compound subject, there are likely to be rules computing the *sum* of agreement features (e.g. 1PS + 3PS = 1PP). These *agreement sum rules* must thus have priority over agreement;
4. subject insertion as a whole cannot simply have priority over agreement sum, as the insertion of nested modifying noun phrases may hinder the sum. Agreement sum must thus be *interleaved* with subject insertion;
5. sometimes and especially for languages with more than two number features (e.g. LIBRAS, the Brazilian Sign Language), the exact number of a noun phrase is not given by the source node of its nucleus (e.g. "those two girls"). Thus, at least some noun modifier insertion rules must have priority over agreement sum.

Obviously, the tasks above are error-prone, since each of these rule subsets (subject/object insertion, agreement, etc.) usually contains numerous low-level, hardly readable rules involving various artificial control features to keep DeCo's abstract machine on track. Furthermore, priority can be implemented not only by rule ordering but also – and often – implicitly, by careful positioning of the heads, which is always a must, anyway. Frequently a whole process is triggered by one single rule waiting on a certain feature/command under e.g. the left head only. The developer then choreographs the heads ingeniously so that the left head will only pass over the trigger at the right time. This is known to be a major source of unmanageability but is hard to avoid in real DeCo programming.

In the following sections, we discuss DeCo's limitations from two perspectives: firstly, as a formal object and, finally, as a software product.

4.1 Formal Limitations

In this section, we demonstrate some undesirable consequences of DeCo's formal specifications, which may hinder if not preclude operations necessary to quality deconversion.

Precondition Language

As routine a phenomenon as verb agreement suffices to demonstrate maybe the greatest among DeCo's limitations, namely the absolute simplicity of the precondition specification language. Cells do not hold attribute-value matrices, just plain feature sets; each feature is literally atomic; and the precondition language is equivalent to predicate logic. This means that, even though features like (i) *pers=1ps* and (ii) *pers=3pp* are possible, they appear to preconditions as if unrelated. Therefore, for each and every possible person feature a subject may assume, there must be a distinct rule to generate the corresponding information in the verb. It is simply *not possible* to express something like:

```
if read(left,SUJ) and read(right,V) and
   read(left,pers=$X) and not(read(right,pers=$_))
then write(right,subjpers=$X)
```

with `$X` and `$_` as variables. Neither would `read($X,SUJ)` be possible, implying that, if subjects were to be generated now to the right, now to the left of verbs, then there would have to be distinct rules to deal with each side, doubling the number of agreement rules.

In general, implementing any n-ary function (i.e. one head writing a specific value whenever *n* features of the form $Param_i=Val_i$ are read) takes as many rules as the cardinality of its domain, or rather, the *product* of the cardinalities of the domains of its parameters. If exactly the same function should yield its result now in the left, now in the right focused cell, taking its parameters from the other focused cell, then that number of rules doubles.

Linear, Non-Structured Output

Much of the awkwardness exemplified in the previous topic is due to the fact that DeCo gradually builds a linear tape of otherwise formally unrelated cells, which ultimately – and implicitly – stands for a highly structured entity, i.e. some sentence/text in a target language. Surely deconversion rules are designed to impose e.g. agreement and positioning constraints between syntactic constituents. The nuisance is that these constraints can never be expressed in terms of real syntactic structure. Configuration would be more natural if deconversion could be analyzed as if including two decoupled steps as follows:

- **syntactic mapping,** in which real syntactic nodes were created and explicitly related to each other as if, given e.g. a verb node and a noun phrase root node, the developer could simply say "Verb, this is your subject!"; and
- **governor-governee constraining**, in which one could simply state "this class of verbs agrees in person with its subject and object" or "this other class agrees in gender with its subject" and rely on the deconversion engine to impose these constraints during syntactic mapping implicitly.

Subgraph Matching

Deconversion would simply not be possible if rules were not able to sense the input UNL graph, even if only from the limited point of view of a focused cell. In fact, DeCo allows node-transferring rules to inspect no further than exactly one node directly linked to one focused cell (the other head is over the inspected node, so to speak). It is left to the other rules at most to sense relation-related features, i.e. those of the form >*R* or <*R* indicating that the inspected cell plays the left or right role in some relation labeled *R*.

Therefore, if ever a higher-level translation step requires inspecting/matching a less limited subgraph as a purely *semantic* precondition, then a cumbersome routine is in order of transferring the whole subgraph onto the tape and next deleting the undesired cells. This has an extreme side-effect: if the precondition succeeds, then the relations of the subgraph will have been explored and thus can never be traversed again by other translation steps. Unless the referred step coincidentally consumes the whole subgraph, that side-effect is unacceptable. In short, DeCo does not support general subgraph matching, which represents a heavy constraint on the expression of semantic preconditions.

Even if such a transfer-inspect-delete routine happens to be acceptable in a particular case, not only will it be difficult to choreograph, but also it will entail the creation of several undesired lexical bindings and related choice points, as the dictionary is necessarily accessed. Moreover, for reasons explained in the previous topic, the implementation of one such routine can seldom be reused by similar semantic preconditions on subgraphs.

Graph Editing and Nontrivial Maneuvers

If graph sensing is as restrained as described in the previous topic, graph editing follows closely, receiving the status of a mere side-effect. In fact, a relation can be considered erased once it has been explored in node transference, as it can never be explored again. However, more sophisticated operations are usually most welcome. Consider, for example, a real LIBRAS generation case in which informers seemed to neutralize the difference between English (i) "to keep something Xed" and (ii) "to X something", producing one single LIBRAS version reflecting (ii) more closely. One actual translation pair was the following:[8]

[8] All real-case source sentences are originally in Portuguese. However, whenever the differences between English and Portuguese are not relevant, we show only English translations to improve readability.

source: We must keep water tanks closed!
target: `[water tank must close-with-lid{objlidgend:flat}]{excl}`

The problem here is that, in order to produce the same target sentence, the source could as well be "We must close water tanks!", which would actually bear a more direct structural relation to the target. We managed to tackle this neutralization with DeCo at the expense of the following rather awkward ad-hoc strategy:

1. add as many entries under UW "closed" as to replicate all the possible mappings of UW "close";
2. make sure that the features of these new entries would avoid their application on trivial deconversion of UW "closed";
3. make the entry under UW "keep" map into an empty token and contain some common verbal features that would trigger subject insertion and aggreement;
4. add one special feature to that entry triggering a complex procedure as follows:
5. explore the obj relation to insert the root cell of the object;
6. copy subject agreement information from the "keep" cell into the object cell;
7. prior to full object development, use the object cell as an anchor to transfer the node accessible through the aoj relation, requiring the inserted cell to be a verb. This new verb cell will be inserted either to the left or right according to its own feature set, which should inform the required relative position of an object;
8. copy subject agreement information from the object cell into the verb cell;
9. develop the object fully;
10.copy object agreement information into the verb cell and so on.

Complex though it may seem, the description above is much simpler than the actual implementation, which involves subtle rule prioritizing and choreographing. It is worth noticing that we needed to implement and activate a whole rule set alternative to regular verb insertion rules. Were graph editing operations available in a decoupled form, it would have been much neater to perform a purely semantic transformation as depicted in Fig. 3 and next apply regular rules to the new root node. No ad-hoc entries would have to be added to the dictionary then; on the other hand, an additional semantic resource would be needed to map UW "closed" into "close".

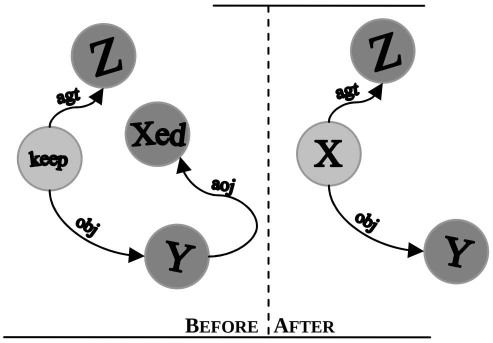


Fig. 3. Graph editing operation implementing the neutralization between "to keep something Xed" and "to X something"

Optimality

DeCo is strict in that (i) a solution is depth-first searched for and (ii) the first solution found is the only one to be outputted. This is built in DeCo's design and is acknowledged to rule out any chance of defining a quality measure to optimize. Even if it were possible to relax (ii) and have some external device eventually rank all the solutions, DeCo would probably timeout when searching for alternatives, since failure by timeout should always be enabled even for toy configurations, so many the created choice points usually are.

Therefore, all such concepts as models of conciseness/readability/etc. (see e.g. Eddy [8]) or Optimality-Theoretical soft constraints are rendered inapplicable. However, it is possible to implement simple rules of thumb locally to decide whether a constituent (e.g. a relative clause) should be generated now in a position, now in another based e.g. on its length. Unfortunately, this requires some nontrivial programming and risks so much backtracking as to lead to timeout. DeCo would have to be instructed to (i) generate the whole constituent first in one position, suitably delimited by special markers; (ii) check the size constraint on the basis of those markers; and, if the constraint were not satisfied, (iii) force failure in order eventually to try alternative rules starting generation in a different position.

4.2 Architectural Limitations

As a piece of software design, DeCo's architecture is perfectly closed in that at no time can user-defined modules aid deconversion. In other words, neither dictionary entries nor rules can refer or resort to any entity whatsoever outside the standard system, which can be a serious limitation to some applications. For a start, even if graph editing facilities were available, the semantic neutralization scheme exemplified in the previous section could not be implemented in this scenario, as it requires a special semantic resource to map e.g. UW "closed" into "close".

However, the need for interoperability becomes patent when one take into account that at times the source UNL graph may lack pieces of information essential to quality or even grammatical deconversion. This situation is frequent when LIBRAS is the target language. Sometimes a UW is just too general, like "cut", which simply has no direct translation. LIBRAS "cut" signs necessarily incorporate an instrument; therefore, there are only specific signs such as `cut-with-a-knife`, `cut-with-scissors`, `cut-with-a-saw` and so on. It is worth noticing that mistranslation would lead then to ungrammatical, unintelligible or at least seriously misleading sentences. Hence the need for a semantic resource to answer such queries as "With which instrument is X usually cut?"

In some cases, even a knowledge base, which could suffice for answering most such queries, is not enough. Take grammatical number in LIBRAS for example, which may assume as many as five values, namely singular, dual, trial, quadral and plural. According to current UNL codification standards, there is rather likely to be a simple node with attributes `@def` (definite) and `@pl` (plural) actually referring to a group of entities already mentioned by some preceding UNL graph in UNL-encoded text. Moreover, it may well be the case that the actual cardinality of the group has been made explicit in that or yet another previous reference. If so and the current sentence presents any kind of person agreement, then this cardinality – in the form of a corresponding number value – will be paramount to producing a sound LIBRAS version. Due to DeCo's closed architecture, there is no chance it would do anything but blindly guess in such a situation.

It is actually a non-issue here what the exact nature of such external devices should be. What is required is that a deconversion engine should be able to query them, and developers should be free to develop and employ any number of devices necessary in a specific application. To mention one less usual example, in our Portuguese-LIBRAS translation project, named PUL∅ (Portuguese-UNL-LIST DeOralizer), we intend to overcome such accuracy-demanding challenges as mentioned above by resorting to **human aid**, though strictly non-specialized in that only required to be proficient in the source-language (Portuguese). This means PUL∅ withdraws from real-time translation and resigns its operation to what we call edition (i.e. pre-publishing) time.[9] Therefore, in addition to a minimized knowledge base, PUL∅'s deconversion engine shall query an interactive device which, whenever necessary, should reply on the basis of a human editor's answers to questions elaborated on the fly.

5 Meet Manati

Manati is the linguistic realization engine all UNL-LIST conversion in PUL∅ is based on. In other words, PUL∅ includes a configuration of Manati, i.e. a module obtained by fixing Manati's parameters. It should be clarified at once that Manati, much unlike DeCo, is not an application, but a **software framework** or simply a **library** to serve as a foundation for UNL deconversion modules/systems. This should not be regarded as a disadvantage, actually being particularly favorable to **interoperability**. For example, auxiliary devices external to Manati, such as user prompts or knowledge bases, can be directly built in the final application; and the power of a full-fledged programming language is available to help handle complex translation procedures. The framework is fully implemented in Oz (www.mozart-oz.org [14, 17]) and heavily draws upon the expressiveness and elegant, seamless multiparadigm integration of this language to meet its requirements. The following description assumes some familiarity with the terminology of higher-order, constraint and especially object-oriented programming.

Manati[10] is undoubtedly DeCo's child. The parentage is not only historical – as it was only after experimenting with DeCo that Manati could be conceived, and it is in DeCo's shortcomings that one can find much of Manati's rationale – but also conceptual. Several features embryonic in DeCo have been generalized and above all *de-*

[9] Taking into account how rare and costly bilingual human translators are in this case, one can easily understand how reasonable this tradeoff is.

[10] Manati is named in honor of its idol and definitive evolution-perfected form, the legendary Babel Fish [1], which feeds upon mixed-up brain-wave energy and absorbs all but intentional linguistic thought. Just as manatis are not really fish, Manati is not a Babel Fish and strives to digest the UNL into one target language at a time.

coupled in Manati. Manati's lexicon-driven **delegation model** is perhaps the most outstanding of its resemblances to DeCo, although each lexicon entry now states not simply a mapping, but rather a **translation rule** triggered by a UW. Each rule covers an arbitrary subgraph, inspects and changes the input graph at will, builds an arbitrary portion of the output and eventually delegates the translation of other adjacent subgraphs by invoking translation rules for boundary nodes via the lexicon.

Manati takes **decoupling** seriously. The very concept of a translation rule is not atomic, being the crossing of four orthogonal concepts – semantic precondition, syntactic mapping, governor-governee constraints and linear precedence constraints – each of which are implemented separately by four distinct class hierarchies. Rules are obtained by combining classes from these hierarchies interchangeably. At all times, class definition is supported by high-level constructs, e.g. syntactic dependency trees [7] and morphosyntactic feature structures in syntactic nodes. It follows that Manati produces **highly-structured output**, which, nonetheless, can straightforwardly be printed out as a sentence.

Efficiency and **optimality** are also major concerns. The ultimate goal of configuring Manati is instruct it to derive a low-level constraint satisfaction problem (CSP) description, effectively exploiting propagation, when it is fed with input. Naturally, a quality measure should integrate the derived CSP; and search for an optimum solution is carried out as usual in constraint programming. This programming paradigm was chosen due to its potential to reduce search dramatically. Its application in conjunction with the dependency tree formalism follows work by Duchier [4][5] & Debusmann [7], which focused on parsing. Their research also inspired Koller & Striegnitz's generation work [9], which is, however, fundamentally distinct from ours in that it strictly focuses on taming flat semantics, a non-issue here.

5.1 Parameters

Manati currently allows the rationalized configuration of ten *orthogonal* parameters in that independently and modularly defined, namely:

1. **input formalism**, which, even though restricted to hypergraph types, is free to accept any open set of UWs (node labels) and closed set of relations (edge labels);

2. **morphosyntax:** each part of speech (POS) in the target language must be defined as a record with arity *{avm, constr}*, where feature *avm* is an attribute-value matrix (AVM) type, and *constr*, a constraint on instances of *avm*. Whenever a morpheme *M* is generated with part of speech *P*, then *M.feats* denotes a unique morphosyntactic feature structure for which:

   $$M.feats \in P.avm \wedge P.constr(M.feats)$$

   holds. Furthermore, given that *M.roles* denotes the actual label set of all syntactic relations having *M* as a governor, the invariants:

   $$M.feats.reqComps \subseteq M.roles$$
   $$M.roles \subseteq M.feats.reqComps \cup M.feats.optComps$$

also hold, meaning that POSs must necessarily define at least features *reqComps*, specifying required syntactic relations (as to complements), and *optComps*, specifying optional ones (as to adjuncts).

POS declaration in Manati is extremely user-friendly, allowing inheritance hierarchies and expressiveness in defining AVM types building on work by Duchier et al. [6]. Attribute types can be any of (i) atom from a finite domain, (ii) set of atoms from a finite domain, (iii) the cartesian product of such sets or (iv) nested AVM. The *contsr* features of POSs have intuitive notational support (also due to Duchier et al.) and are useful when stating e.g. that a given set of nouns/verbs imply specific gender/tense;

3. **syntactic mapping:** a specific mapper class hierarchy must be provided in order exclusively to specify the mapping of UNL (hyper)graphs onto syntactic dependency trees in the target language. Roughly speaking, mappers simply convert (i) semantic nodes into lexemes (classes of lexical items) and (ii) semantic relations into syntactic roles; or, in Natural Language Generation jargon, they are responsible for *lexical choice* and *aggregation* [13]. It is worth noticing that mappers are not interested either in morphosyntatic constraints, such as agreement, or in final linear ordering of morphemes.

   During generation, according to information in the lexicon (see below), a set of *mutually exclusive* mappers are instantiated for the global UNL entry node *Src*. Each mapper (i) tries to recognize a specific subgraph of its own starting at *Src*, performing whatever necessary semantic checking on candidate nodes, (ii) creates a set of syntactic nodes (usually corresponding to target language words) and (iii) establishes binary syntactic relations between them. Some of the nodes in (ii) may be created by means of recursively applying the same process to some of the source nodes in the subgraph recognized in (i), as mappers always yield exactly one syntactic root node. In time, the root *received* by a mapper as a result of recursion might actually be a *selector node* choosing from the root set of several mutually exclusive subtrees produced by alternative mappers.

   If the mapper class hierarchy is well-defined and correctly employed in the lexicon, the process sketched above will traverse the source UNL graph tree-wise from its global entry. For more complex operations such as generating relative clauses and dealing with coordination – or sometimes simply to avoid infinite cycling – some input formalisms (and UNL flavours) require that mappers be able to edit source hypergraphs. The edit operations available are node insertion and edge deletion and insertion. In any case, however, changes by a mapper are only visible to itself and the mappers it creates recursively;

4. **mapping preconditions:** in order to optimize resource usage during search, part of if not all precondition checking in mappers can optionally be delegated to a specific class hierarchy. Such so-called *precond* classes are associated with mappers by lexicon data. See “lexicon” below for details;

5. **governor-governee constraints:** a specific class hierarchy must be provided in order exclusively to tell morphosyntactic constraints on each pair of syntactically related target nodes (i.e. words or morphemes). Such so-called *gamma* classes are associated with mappers by lexicon data and have methods of the signature

Role(Parent.feats Child.feats) invoked for each syntactic relation *Role* their corresponding mappers establish between any target nodes *Parent* (governor) and *Child* (governee);

6. **linear precedence:** a specific class hierarchy must be provided in order exclusively to determine the final ordering of target nodes and carry out whatever further tasks that might occasionally be required on mapper completion, when all direct child nodes are accessible – though not as yet fully determined – for e.g. telling further constraints. Such so-called *finishUp* classes tackle linear precedence by telling constraints relating target nodes to each of their children and siblings to each other. Order constraints, though definable at various levels of abstraction, ultimately operate on features *roots* and *yield* of nodes or *role bundles* – a simplified interface to all siblings filling one same syntactic role. Feature *roots* denotes a set containing either the absolute position of a node within the generated text or the union of all *roots* features of the siblings in a role bundle. Feature *yield* denotes the union of either all *roots* in the subtree rooted at a node or all *yield* features of the siblings in a role bundle. If needed, role bundles also give access to each "bundled" sibling individually.

   Following Duchier & Debusmann [7], ultimate control over linear precedence is provided by constraints operating also on *topological fields*. The concept involves axiomatizing a *topology* of the *yield* of a syntactic tree, i.e. a *partition P[i]* such that:

$$\forall x,y,i,j\ (x \in P[i] \wedge y \in P[j] \wedge i < j \rightarrow x < y).$$

   Each partition element *P[i]* is said a *topological field*. Manati allows absolute flexibility in axiomatizing topologies, including the possibility of nesting, one topology holding for an entire tree unless a mapper overrides it for subtrees. Nested topological fields can be unified with those in an overridden topology granting finer-grained overriding control.

   Topologies provide for long-distance "movements", topicalization, nested clauses and the generation of multiple sentences from a single source graph[11], which is essential for Libras generation;

7. any number of **oracles** – e.g. user prompts, knowledge bases, etc. – to resort to at virtually any generation stage. Oracles are services running concurrently and accepting asynchronous requests. The sole requirements on oracles are (i) requests must be ground, i.e. involving no unbound variables, and (ii) responses must be either ground or finite domain variables (in some commonly agreed protocol, e.g. 0/1 meaning true/false) eventually to be determined by oracles themselves;

8. **lexicon:** Manati's lexicon is more of a **translation rule base**, each of whose entries is a tuple *(UW, TransList, POS, Precond, Mapper, Gamma, FinishUp)*, where *UW* is a source node label; *TransList*, a character string list of possible target lan-

[11] Notice that it is always possible to define a rightmost/leftmost topological field to contain trailing/preceding text.

guage translations; *POS*, the part of speech of the elements of *TransList*; and *Precond*, *Mapper*, *Gamma* and *FinishUp*, classes of the homonymous types.

When the translation of a source node is required, its label is used to search the lexicon for eligible transfer rules. For each rule, *Precond* is activated, performs specified checking and, iff *TransList* has more than one element, must select exactly one of its elements to be the translation word of the rule. *TransList* may as well be empty, but then the definition of a translation word is left to *Mapper*, which hinders code reuse and search efficiency. The *null* word is also possible, creating an invisible syntactic node and enabling null categories.

From this point on, Manati attempts to optimize the application of constraint programming by instantiating one single mapper for each set of so far successful rules sharing the same *Mapper* class. Each mapper receives a default syntactic target node constructed from the data remaining in its originating set of rules, i.e. translation words, *POS*s, *Gamma*s and *FinishUp*s. This is actually a complex two-level selector node built with the powerful Oz selection constraints. It is up to each mapper to decide what to do with its default target node: (i) simply ignore it (not wasteful due to lazy evaluation) or (ii) use it as a final target to receive children or likewise (iii) as part of any arbitrary syntactic structure it may build;

9. **output formalism,** i.e. how the resulting syntactic trees are to be printed out. This is highly configurable ranging smoothly from raw lists of target language words to fully structured trees by means of user-defined bracketing. Words and bracketed groups may be associated with arbitrary *Output AVMs* (OAVMs) created by *FinishUp* classes. OAVMs may be useful to add syntactic and prosodic annotations (as required by PUL∅) or even to output morphologic features, leaving full inflection of words to dedicated modules and thus downsizing the lexicon. These are the facilities that allow the generation of prosodic groups and AVMs in LIST;

10. **quality measure**, in the form of a binary constraint *Q* that, during search, is iteratively imposed on pairs *(CurBest, Wannabe)*, where *CurBest* is the best fully determined solution so far, and *Wannabe* is a partially determined solution which will attempt to be even better than *CurBest*. In fact, it is exactly *Q* that should give *Wannabe* a drive to supersede by strictly constraining it to be better. As *CurBest* and *Wannabe* are given as the roots of their respective syntactic tree solutions, and *Wannabe* is not yet fully determined, which rules out direct access to its subtrees, constraint *Q* should relate both solutions solely on the basis of their roots. Therefore, for complex quality measures, developers are expected to include quality-related features in POSs (see “morphosyntax” above) and have them propagate up the tree by means of special constraints in *gamma* and *finishUp* classes.

    So far we have only experimented with minimizing output length in words. This is especially interesting to LIBRAS generation because it is rather often the case that two or more distinct words can actually be combined into a single preferred one. As Manati provides every node with feature **`yieldCard`**, denoting the cardinality of its *word yield*,[12] this quality measure is optimized by a simple procedure `LengthOrder` as follows:

[12] The *word yield* of a node is the subset of its *yield* that actually corresponds to target language words, which excludes null categories and bracketed block nodes.

```
proc {LengthOrder CurBest Wannabe}
    Wannabe.yieldCard <: CurBest.yieldCard
end
```

where `X <: Y` is not the usual comparison operation, but rather a constraint, telling that `X < Y` should always hold. Other measures are usually of interest to systems relying on e.g. heavy content selection and advanced referring expression generation, which are almost absent in PUL∅ since these tasks are satisfactorily performed by source text authors to current standards. Alternatively, if one is interested in the very first solution only, it suffices to provide the following even simpler constraint:

```
proc {FirstWillDo _ _}
    fail
end
```

5.4 Searching for a Global Optimum

Deconversion starts by applying translation rules to the global UNL entry node. In spite of involving some pattern matching and search, this corresponds to model creation only and yields a complex partially determined syntactic tree, whose distinct potentialities are modelled by occasional **higher-order selector nodes** choosing from (the roots of) a set of subtrees. In addition to `yieldCard`, every node has two further important features *affected by constraint propagation*, namely: **`id`**, denoting its absolute position in the generated text, and **`active`**, denoting an encoded boolean telling whether the node actually takes part in the current solution or is discarded. Every higher-order selector node has at most one active selectable root at a time and is active iff it has exactly one such root. If so, that root becomes selected, which makes its features (`id`, `active`, `yieldCard`, etc.) and those of the selector coincide. Finally, all other nodes are actually **first-order selectors** choosing from a list of alternative target language words, for which reason they have an additional **`lexI`** feature, denoting the index of the word of choice.

Manati's search script is just like any ordinary Oz script and is executed in cycles of constraint propagation followed by domain distribution until a solution is found. It reads as follows:

1. distribute over the vector of all *active* features, prioritizing (i) activation over deactivation and (ii) elements in order of appearance, which roughly corresponds to the order in which translation rules appear in the lexicon;
2. *ActiveNds* ← list of all active first-order selector nodes. The notation *List.Feature* used in subsequent steps denote the vector obtained by selecting *Feature* for each item in *List*;
3. tell $\forall Id \in ActiveNds.id: dom(Id) \subseteq \{1 \ldots length(ActiveNds)\}$;
4. distribute naïvely over *ActiveNds.lexI*, i.e. trying lower values first;
5. distribute over *ActiveNds.id* using a first-fail strategy, i.e. prioritizing the distribution of the most constrained `ids` as an heuristic to rule out failed choices first and thus minimize their impact on search;

6. if a fully determined solution *CurBest* is found, try to improve it by starting over with a fresh model *Wannabe* and ensuring that *Q(CurBest,Wannabe)* should hold, for a given quality constraint *Q*.

6 Conclusions and Future Work

We have scrutinized DeCo and demonstrated that some of its features are likely to hinder productivity and quality in deconverter development. These features can be summarized as strong coupling of concepts, lack of generality, low level of abstraction and no support for modularity, abstraction, optimality or interoperability.

As a first attempt to overcome these shortcomings, we have presented the first draft of Manati, an alternative linguistic realization/UNL deconversion engine. Manati heavily draws on constraint programming as a means to reduce search; while object-oriented and higher-order programming provides a basis for defining friendly primitives with which (i) to fill the blanks (i.e. parameters) of a configuration at appropriate levels of abstraction and (ii) automatically to derive a low-level constraint satisfaction problem (CSP) description, effectively exploiting propagation, when a configuration is eventually fed with input.

Manati is currently being configured to generate the Brazilian Sign Language and shall be evaluated against other linguistic realization engines in the near future. Scheduled further work on Manati includes full coverage of generation tasks [13] – e.g. content selection – and experiments with different quality measures. Additionally, as our experience of applying Manati in real-case scenarios increases, we expect to produce even higher-level abstractions building on Manati’s current facilities.

Acknowledgements This project has been partially funded by MEC (the Brazilian Ministry of Education). We would like to thank Prof. Dr. Tanya Amara Felipe (UPE – Universidade de Pernambuco) and her team for providing the LIBRAS versions of this article.

References

1. Adams, D. *The Hitchhiker's Guide to the Galaxy*. Ballantine Books, 1995 (reissue edition).
2. Brito, L. F. Por uma Gramática de Línguas de Sinais. Tempo Brasileiro Ed., Departamento de Lingüística e Filologia, Universidade Federal do Rio de Janeiro, 1995.
3. Cahill, L. and Reape, M. *Component tasks in applied NLG systems*. Technical Report ITRI-99-05, Information Technology Research Institute (ITRI), University of Brighton, 1998. http://www.itri.brighton.ac.uk/projects/rags.
4. Duchier, D. Configuration of labeled trees under lexicalized constraints and principles, *Journal of Language and Computation*, 2002.
5. Duchier, D. Axiomatizing dependency parsing using set constraints. In *Proceedings of the 6th Meeting on the Mathematics of Language*, USA, 1999.
6. Duchier, D., Gardent C., and Niehren J. *Concurrent Constraint Programming in Oz for Natural Language Generation*. http://www.ps.uni-sb.de/~niehren/Web/Vorlesungen/Oz-NL-SS01/vorlesung (accessed on Aug. 2004).

7. Duchier, D. and Debusmann, R. Topological dependency trees: A constraint-based account of linear precedence. In *Proceedings of the Association for Computational Linguistics (ACL)*, France, 2001.
8. Eddy, B. Toward balancing conciseness, readability and salience: an integrated architecture. In *Proceedings of the International Natural Language Generation Conference (INLG'02)*, 2002.
9. Koller, A. and Striegnitz, K. Generation as Dependency Parsing. In *Proceedings of the Association for Computational Linguistics (ACL)*, 2002, 17-24.
10. Martins, R. T., Rino, L. H. M., Nunes, M. G. V., and Oliveira Jr., O. N. The UNL distinctive features: evidences through a NL-UNL encoding task. In *Proceedings of the First International Workshop on the UNL, other Interlinguas and their Applications*, 2002, 08-13.
11. Martins, R. T. *A Língua Nova do Imperador*. Ph.D. Thesis, Instituto de Estudos da Linguagem, Universidade Estadual de Campinas (UNICAMP), 2004.
12. Paiva, D. *A survey of applied natural language generation systems*. Technical Report ITRI-98-03, Information Technology Research Institute (ITRI), University of Brighton, 1998. http://www.itri.brighton.ac.uk/techreports.
13. Reiter, E. and Dale, R. *Building Natural Language Generation Systems*. Cambridge University Press, 2000.
14. Schulte, C. *Programming Constraint Services: High-Level Programming of Standard and New Constraint Services*, Lecture Notes in Computer Science Series, Springer-Verlag, 2002.
15. Speers, d'A. L. *Representation of American Sign Language for Machine Translation.* PhD. dissertation, Graduate School of Arts and Sciences, Georgetown University, 2001.
16. Stone, M. and Doran, C. Sentence planning as description using tree-adjoining grammar. In *Proceedings of the Association for Computational Linguistics*, 1997, 198-205
17. Van Roy, P. and Haridi, S. *Concepts, Techniques, and Models of Computer Programming*, MIT Press, 2004.
18. Uchida, H., Zhu, M. and Della Santa, T. *UNL: A Gift for a Millennium*. Institute of Advanced Studies, University of the United Nations, 1999. http://www.undl.org/publications.

Arabic Generation in the Framework of the Universal Networking Language

Daoud Maher Daoud

ALzaytoonah University, Amman, Jordan
daoud_m@yahoo.com , daoud@maherinfo.com
&
GETA, CLIPS, IMAG
daoud.daoud@imag.fr

Abstract. This paper describes the work done on the developing of Arabic Deconversion within the framework of the Universal Networking Language (UNL). In this paper, the architecture of the system is explained along with the strategy used for the development. We also discuss issues and problems related to the UNL representation that affect the quality of generation. Additionally, the lingware engineering is introduced as a technique to enhance the quality and increase the development efficiency.

1 Introduction

Arabic is one of the world's main languages. It is the official language for over 289 million people. It is also the sacred language of nearly 1.48 billion Muslims throughout the world.

The alphabet consists of twenty-eight consonants but three of these are used as long vowels. Arabic also contains short vowel signs being indicated by marks above or below the letters. Like other Semitic languages, Arabic is written from right to left. It is a language characterized by rich morphology: most of the words are built from consonantal roots in which inflections and derivations are generated by vowel changes, insertions, and deletions.

The Universal Networking Language is a specification for the exchange of information. It is a formal language for symbolizing the sense of natural language sentences.

Currently, the UNL includes 16 languages. These include the six official languages of the United Nations (Arabic, Chinese, English, French, Russian and Spanish), in addition to ten other widely spoken languages (German, Hindi, Italian, Indonesian, Japanese, Latvian, Mongol, Portuguese, Swahili and Thai). In its second phase (1999–2005) the project will seek to further extend UNL access.

This paper presents the work completed on the generation of Arabic from UNL during the author's employment with Royal Scientific Society (RSS) in Jordan and his work on the UNL project. It described the work done on the generation of Arabic from UNL between 1996 till 1999. Since then, we think that the generation system maintained its main architecture.

Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 195–209.

Section 2 gives a description of the system that generates Arabic sentences from UNL representations. In section 3, we show how the system in implemented. In this perspective, issues such as mapping of relations, word ordering, and morphological generation are addressed. Results, future works, and conclusions are presented in Sections 4, 5 and 6 respectively.

2 The Architecture of Arabic Generation (Deconversion) System

Universal networking language (UNL) is a semantic, language independent representation of a sentence that mediates between the enconversion (analysis) and deconversion (generation). It is a computer language aiming at removing language barriers from the Internet. The pivot paradigm is used: the representation of an utterance in the UNL interlingua is a hypergraph where normal nodes bear UWs ("Universal Words", or interlingual acceptions) with semantic attributes, and arcs bear semantic relations [13].

The sentence "Khaled bought a new car" can be expressed in UNL as:

agt*(buy(icl>do(obj>thing),icl>purchase).@past.@entry, Khaled)*
obj*(buy(icl>do(obj>thing),icl>purchase).@past.@entry, car(icl>automobile))*
mod(car(icl>automobile),new)

Figure 1 shows the graph representation of this UNL expression. The node represents the Universal Word (UW). Arcs represent binary relations such as "agt", "obj" and "mod". Attributes are attached to UW to include information about time, aspect, number, modality, etc. In the previous sentence, the attribute "@past" was attached to the event "buy" to indicate that the event happened in the past. The "@entry" attribute is used to indicate the entry point or main node for the whole expression.

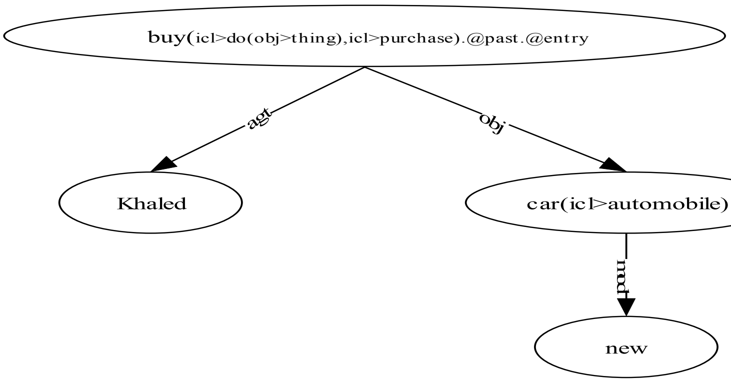


Fig. 1. The graph of the UNL expression

Generation of the target sentence is the process of converting the UNL expression or the hyper-graph to one dimensional character string. We use the DeCo tool, which is provided by UNDL foundation to work the Arabic Deconversion. On the other hand, enconversion is the process of generating UNL from a natural language text. A software for enconversion called "EnCo" which constitutes an enconverter together with a word dictionary, UNL knowledge base, and conversion rules for a language [4].

2.1 Deconverter

DeCo is a language independent system capable of traversing any UNL graph and constructing morphemes based on each node visited. As shown on Figure 2 the main inputs of the Deco tool are a UNL expression, a dictionary, and generation rules. First, DeCor transforms the sentence represented by an UNL expression - that is, a set of binary relations - into the directed hyper graph structure called **Node-net**. Then, it applies generation rules to every node in the node-net respectively, and generates the word list in the target language (Node-list) [3].

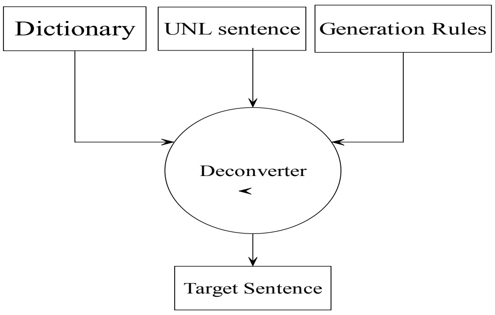


Fig. 2. The deconversion Process

The order of traversal is specified by the generation rules, which also systematize the preference of the target lexis.

The DeCo engine employs the generation rules to map the UNL expression into the appropriate syntactic and morphological structure of the target sentence.

The DeCo tool implements an abstract transducer model with multi-heads (or windows). The DeCo tool uses two types of windows: Generation Window (GW) and Condition Window (CW). There are 2 GWs bordered from both sides by several CWs (Figure 3).

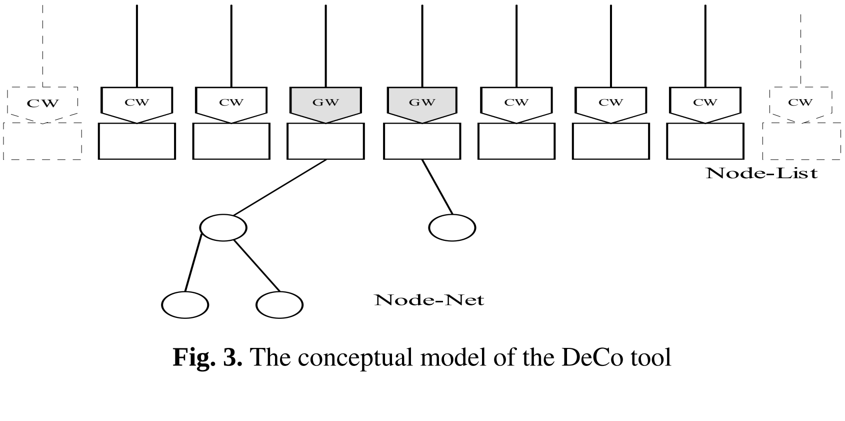


Fig. 3. The conceptual model of the DeCo tool

The Condition Windows are used to test out the conditions in the neighboring nodes. Alternatively, the Generation Window is able to test a condition, modify attributes, and insert nodes from the graph to the Node-List.

2.2 The Generation Rules

A generation rule is a finite collection of instructions, each calling for a certain operation to be performed if certain conditions are met.
Every rule is of the form:

```
<TYPE>
["("<PRE>")" ["*"]]...
"{" | """" [ <COND1> ] ":" [ <ACTION1> ] ":" [ <RELATION1> ] ":" [ <ROLE1>]
"}" | """"
["("<MID>")" ["*"]]...
"{" | """" [ <COND2> ] ":" [ <ACTION2> ] ":" [ <RELATION2> ] ":" [ <ROLE2> ]
"}" | """"
["("<SUF>")" ["*"]]...
"P(" <PRIORITY> ");"
```

As an example of inserting a new node to the right, the following rule layout is used:

```
:{<COND1>:<ACTION1>:<RELATION1>:<ROLE1>}
"<COND2>:<ACTION2>:<RELATION2>:<ROLE2>"
```

When a node in the node-list satisfies the conditions expressed in <COND1>, AND a node in the node-net which is linked to by the relation of <RELATION1> or <RELATION2> is found, AND IF the node satisfies the conditions expressed in <COND2>, THEN the system inserts a new node to the right of node in the node-list, and executes <ACTION1> AND <ACTION2> [3].

Structuring the corresponding target sentence (node-list) is directed by the rules (Figure 4).

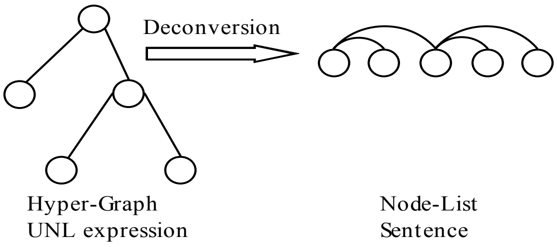


Fig. 4. Building Node-list.

Rules specify conditions and possibly semantic relations needed to trigger actions. Conditions concern the lexical, semantic, and morphological attributes of the node under processing which are specified in the dictionary and/or through the conversion process. Semantic relations are the relations linking two nodes such as agt, obj, etc.

Actions (such as insertion of a node in the node-list) are executed when conditions are met.

For example the following rule:
:{V, >obj: ->obj, +obj_ad, +RBL::} " N,^#N, <obj, ^@pl, ^PLUR:-<obj, +is_obj,+ACC:obj:"P110;

Inserts a node from the UNL graph into the node-list right of the verb node, which is already in the node list. The inserted node is an object of the verb. Besides the insertion action, the rule modifies certain attributes to be used by other rules to add morphological features to the generated word.

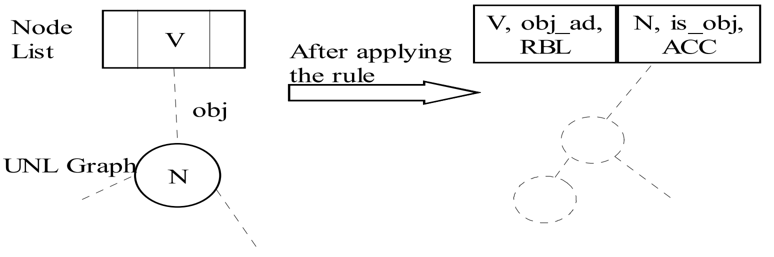


Fig. 5. Applying the rule

Figure 5 shows this process. The insertion of the node is followed by changing attributes: obj_ad (object is added to the verb) and RBL (to add a blank right of the verb) are added to the verb node, is_obj (to mark that this node is an object) and ACC (to mark that the case of this noun is accusative according to Arabic grammar) are added to the object node.

2.3 The Dictionary

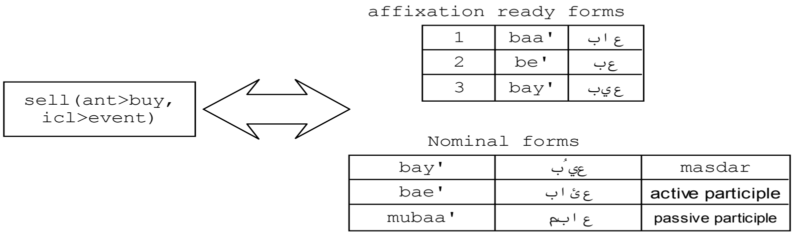
Dictionary stores word entities for each language. The data format of each entity consists of three main components: Head Word (HW) of each local language, Universal Word (UW) and Grammatical Attributes of HW.

Attributes are used by the generation rules to control the selection of the target word in addition to the surface form of the target sentence.

Although Arabic has many inflectional and derivational forms which increase the need to do morphological synthesis rather than full-form listing during the generation process, we preferred full-form listing. This comes from the fact that the DeCo tool lacks the functions to perform infixation.

The generation of lexical entries, (Head word) is based on syntactic and morphological features of each lemma. As a result, each UW can be mapped to different forms that share the same meaning.

For example the UW "sell(ant>buy, icl>event)" is mapped to the HW "باع" [baa'] which is weak middle radical (hollow) verb. As shown in figure 6, different forms are added to dictionary with inflection for gender (masculine, feminine), number (singular, plural, dual), person (first, second, third) , tense (past, present, future), mood (indicative, subjunctive, jussive) and voice.(passive, active) .



affixation ready forms

1	baa'	باع
2	be'	بع
3	bay'	بيع

Nominal forms

bay'	بَيع	masdar
bae'	بائع	active participle
mubaa'	مباع	passive participle

Fig. 6. Entries for verb baa'

In addition to the "affixation ready forms", “nominal forms" are also linked to the same UW which is derived directly from the verb. The nominal inflection of verbs is used to generate accurate sentences. Grammatically these forms act as nouns or adjectives.

inflections	Selected form	Prefix	suffix
masculine, singular, first, perfect, indicative, active	2	-	ت
masculine, plural, third, imperfect, indicative, active	3	ي	ون
feminine,plural, third, imperfect, indicative, passive	1	ت	-

Fig. 7. Examples of inflection and selected forms

The generation rules should select the right form and add the necessary prefixes and suffixes (figure 7).

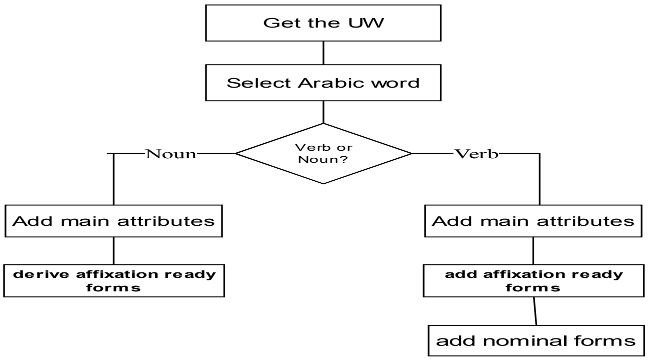


Fig. 8. adding entries for verb and nouns

This approach proved to be feasible also for handling nouns. The linguistic attributes of nouns that have been used in the dictionary are basically: gender, number, case and definiteness.

The issue of broken plural is solved by linking this form to the same UW. Finally the variations in the written forms (such as hamza and nouns ending with long vowel) of Arabic is also handled by making entry for each of these forms in the dictionary.

A database system has been developed for the classification and features adding for each entry in the dictionary. As shown in figure 8 the system gets the UW and tries to get the equivalent Arabic word from an English-Arabic dictionary. The selected Arabic word is then classified to Noun or Verb or Particle. As an example : If the word is denoted as having a broken plural, the system will ask the user to add this entry and both forms are linked to the same UW.

3 The Deconversion of Arabic

Deconversion is the process of producing a grammatically correct sentence from the UNL graph. This process involves mapping of relations, Lexical transfer, word ordering, and morphological generations.

3.1 Mapping of relations

Each UNL relation has been mapped to the corresponding Arabic grammar structure or syntactic relation. It is not a one-to-one mapping as one relation can be mapped to different target grammatical relations. As an example, the "obj" relation can acquire the syntactic role of subject or object or to Idafa construct depending in the UWs involved and the adjacent relations in the UNL graph.

As an example the sentence:

1- The mouse died

Could have the following UNL expression:

obj(die(icl>event).@past.@entry, mouse)

In this sentence, the mouse is not responsible for the event and it undergoes a change of state, so semantically it is the object of the verb die. However, when the sentence is deconverted into Arabic the mouse is the grammatical subject and should have its inflections (nominative).

2- The flour becomes bread.
obj(become(icl>event).@entry, flour)
gol(become(icl>event).@entry, bread)

Since flour experiences a change of state, it is the semantic object of the verb become. Flour is the final state then it is linked with become by the gol relation. Syntactically flour is the nominative actor and bread is the accusative object.

3- He told me a joke.
agt(tell(icl>event).@past.@entry, he)
ben(tell(icl>event).@past.@entry, me)
obj(tell(icl>event).@past.@entry, joke)

In this sentence: *he* is the nominative actor, *me* is the first accusative object and *joke* is the second accusative object.

4- Khaled was killed.
obj(kill(icl>event).@past.@entry, Khaled)

The verb kill is transitive and the agent is deleted from the verb argument. Khaled who is the object of the killing takes the role of "substitute of the doer of the verb" in the generated Arabic sentence. The verb kill is inflected by the passive voice form.

5- Khaled appreciated Ali's learning of French.
agt(appreciate (icl<event).@past.@entry, Khaled)
obj(appreciate(icl<event).@past.@entry, learn(icl<event))
agt(learn(icl<event), Ali)
obj(learn(icl<event), French)

The IDAFA construction is an important grammatical structure in Arabic. It is a genitive construction in which two nouns are linked up in such a way that the second (second particle of the construction) qualifies or specifies the application of the first (first particle of the construction). The usage that concerns us here involves the nominalization of processes, in which the first prticle is typically a *masdar* representing a nominalized process, and the second particle represents either the ‘agent’ or ‘object’ of that process. When the above UNL expression is deconverted to Arabic, “learn” becomes the accusative object of the verb “appreciate” which is converted to the *masdar* form. The *masdar* form of the verb “learns” is also the first particle of the IDAFA construction. The second particle is “Ali” who is the actual actor of the verb “learn” but in the genitive case. “French” is the accusative object of the *masdar* as shown in Figure 9.

The relation mapping is implemented in the deconversion rules. The following rule shows how the relation "obj" is mapped. The inserted node becomes the object and marked by "is_obj" attribute and takes the accusative case "ACC”.
:{V,>obj:->obj, +obj_ad, +RBL::} "NDE,^MASDAR,^#N,<obj, ^@pl, ^PLUR: -<obj,+is_obj,+ACC:obj:" P110;

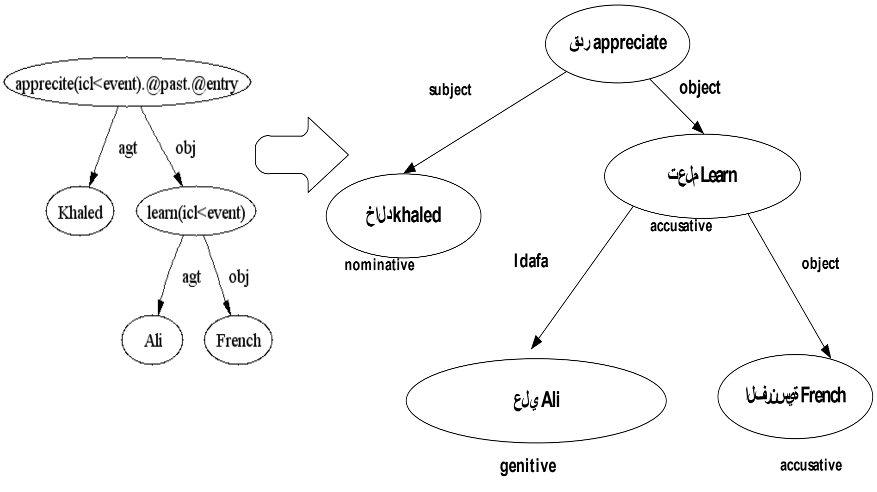


Fig. 9. Mapping of relations

3.2 Word Ordering

Although Arabic shows a flexible word order. It can be said that the dominant or preferred ordering is VSO, the subject and object follow the verb. We also find that specifiers, adjectives, genitives, and relative clauses usually follow the nouns they modify, that adverbs and adjectival arguments usually follow the adjectives they modify, and that noun phrases usually follow the prepositions that govern them. In other words, with very few exceptions, modifiers and arguments usually follow the words they modify or what govern them.

Figure 10 shows the process of insertion and the direction of the UNL graph (5). This is governed by the deconversion rules during the insertion of a new node from the graph to the node-list.

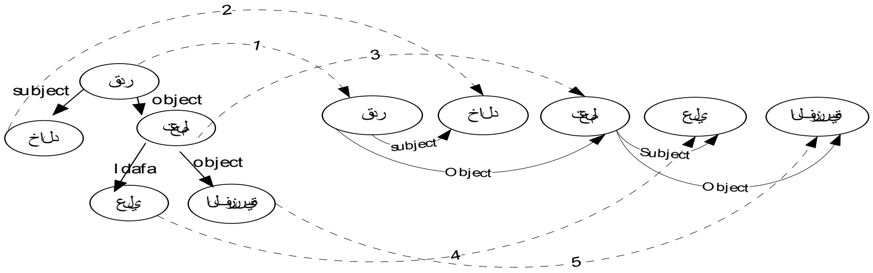


Fig. 10. Insertion sequence and direction

3.3 Morphological Generation

Arabic is a Semitic language, and its basic characteristic is the rich morphology in which most of its words are derived from roots. Inflections and derivations are generated by changing vowels and insertion of consonants.

Arabic sentences are characterized by a strong tendency for agreement between its constituents, between verb and noun, noun and objective, in matters of numbers, gender, definitiveness, case, person etc. These properties are expressed by a comprehensive system of affixation. To satisfy these grammatical properties, generation rules are expected to be complex, to handle the processing of generating grammatically correct Arabic sentences from UNL expression and structure.

In our system, we managed to handle this rich and complicated morphology by implementing a modular approach to coding the rules (figure11).

Our implemented process of morphological generation starts by choosing the right stem which is set to accept prefixes or suffixes depending on its position and role in the sentence.

As an example, both rules below insert a plural subject to a verb already in the Node-list. If the corresponding Arabic word has a regular plural form (by adding the right suffix), then the first rule is executed. Otherwise, the system looks for the other form (broken plural) in the dictionary, and the second rule must be triggered.

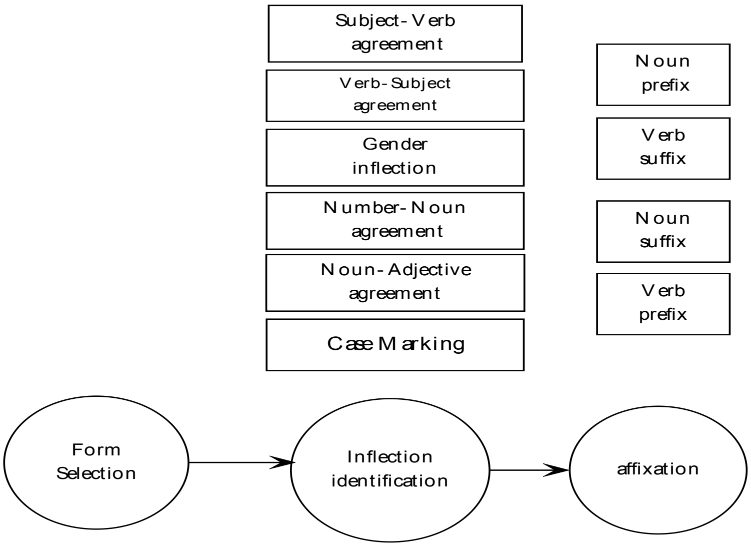


Fig. 11. Morhological generation

Rule 1
:{V,>agt:->agt,RBL,has_subj, V_subj}"^#N,N,@pl,^IRGPL,<agt:-<agt,is_subj,NOM:agt"P100;

Rule 2
:{V,>agt:->agt,RBL,has_subj, V_subj}"^#N,@pl,PLUR,<agt:-<agt,is_subj,NOM:agt"P100;

This approach of word selection is dependent on the syntactic conditions at the time of insertion. However if new facts or conditions become true later, which the selected form does not comply with, the system should backtrack and select a new form. A good example for this observable fact is the implementation of the number-noun agreement that is controlled by complicated set of rules in Arabic. As an illustration, in numbers above ten the noun must be singular, indefinite and accusative and the number takes the grammatical role of the noun. The difficulty becomes very apparent if this noun is attributed by "@pl" in the original UNL expression.

The second phase is marked by identifying types of inflections required to generate quality Arabic sentence such as agreements.

All types of agreements are implemented in our system. For example, Arabic has incomplete agreement in verb-subject sentences. In this case, the agreement will be in the gender but not in the number. Rule 3 (figure 12) shows the implementation of verb-subject agreement. When this rule is executed the verb is marked by the attribute "male_infl". This information is then passed to other rules to add the necessary suffix depending on the type of verb as shown in the rules (4-5) listed below.

The last phase of morphological generation is implemented by prefixing and suffixing rules to mark the inflections identified in the previous process.

```
Rule3
:{V,has_subj:male_infl,V_SUB_GenderAgr}{is_subj,male,^V_SUB_GenderAgr:V_SUB_GenderAgr}P200;

Rule4 :{V,V_subj,QDAA,3person_infl,^add_past,^@not,^suff,male_infl:suff}"[ي]"P2;

Rule5
:{V,V_subj,SAA,3person_infl,^add_past,^@not,^suff,male_infl:suff}"[ى]"P2;

rule6
:{V,V_subj,DAA,3person_infl,^add_past,^@not,^suff,male_infl:suff}"[و]"P2;

Rule7
:{V,V_subj,QDAA,3person_infl,add_past,^suff,male_infl:suff}"[ى]"P2;
```

Fig. 12. Verb-subject agreement rules

In our system, three main groups of rule are designed: insertion rules, inflection identification rules and affixation rules.

4 Results

During the development period of Arabic Module the number of lexical items added to UNL-Arabic dictionary reached 120,000 entries. This covers the UWs provided by UNL center and the most frequent Arabic lexicon. More sophisticated features are added to each entry to cover morphological, syntactic and semantics aspects. In designing those features, we took into consideration the analysis and generation processes.

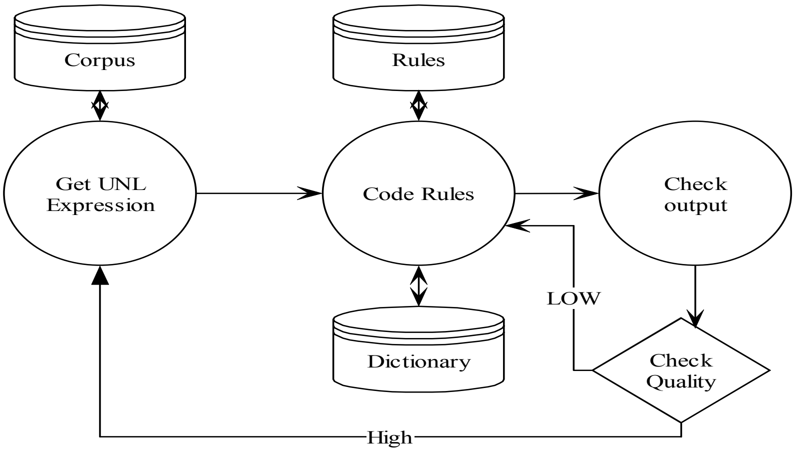


Fig. 13. Current methodology of coding generation rule

Arabic Corpus has been built for Soccer and other topics in order to specify accurately the word usage and to extract the most frequent Arabic words. Functional words are also added to the dictionary along with all prefixes and suffixes needed for Arabic morphology.

The Arabic Deconversion system managed to handle the following situation and sentences:

- Agreement and Morphological generation
- Scope
- All type of relations and attributes
- Loop structure
- Embedded and relative sentences
- Nominal and verbal sentences

However, a variety of problems emerged during the implementation of our system. Some of these problems are related to the nature of UNL others are due to wrong UNL representations of the source language. While Other problems are caused by the dictionary and the limitation of the DeCo tool.

In reality, UNL expressions are not always language-independent. They are influenced by the source language. In the same context, using a separate UWs and relations rather than using attributes to describe subjectivity of the sentences is also a demonstration of the source language influence. For example:

A possible UNL expression of "People no longer have to go." is:
agt(go(icl<event).@obligation.@entry, people)
man(go(icl<event).@obligation.@entry, no longer)
The generated Arabic sentence is not acceptable from the above UNL expression. In contrast, the following UNL expression has produced a good result:
agt(go(icl<event). @obligation-not.@entry, people)

Additionally, multiple identical relations connected to the same UW are problematic since the DeCo tool is incapable of the right word ordering. As examples: "honest (aoj) Jordanian (aoj) citizen" or "I prefer orange (obj) over apple (obj)".

As for dictionary related problems, they are mainly caused by using unrestricted UWs. Leading to imprecise selection of Arabic corresponding words. Besides, the Arabic compound words that correspond to UWs in the dictionary are also a significant cause of low quality generated sentence.

In most cases, the quality of Arabic is highly dependent on the UNL expressions. The need for common consensus and standards among the producers of UNL expressions is important. A grammatically appropriate input sentence is a prerequisite for parsing. Likewise, generation requires correct UNL expressions to produce satisfactory results.

More than 2000 rules have been written to generate the Arabic language.

Figure 13 shows the current methodology of coding the generation rules. It is an evolutionary process, which demands many activities: writing rules, testing, and validation of rules, maintenance of rules, updating, and maintenance of the dictionary. Controlling these activities requires many resources, is very time consuming and the results is not always accurate.

5 Future Work

DeCo is a true SLLPS (specialized languages for linguistic programming), but still of quite "low level" in the hierarchy of programming models. The iterative methodology of writing rules shown in Figure 13 proved to be inadequate. In this methodology of development, the rules are written for each sentence in the corpus and there is no guarantee that any modification will not have harmful side effects. Therefore, a systematic development methodology is necessary to transform the natural language representation of a sentence into rules based representation. The main function of this methodology is to specify diagrammatically the language grammar using language components, which are entities that embed syntactic and semantic information that can be identified in the source language from its unique function in the sentence. These diagrams can be integrated into one development environment enabling systematic development of rules and ease of maintenance.

A Computer Aided rules Engineering (CARE) is needed at this point. It is an integrated environment, which provides a set of tools for the production and maintenance of the rules and dictionary.

Basically the proposed system consists of the following main components:

- Language Modeling
- Repository
- Rules Generator
- Dictionary Maintenance

Language modeling module facilitates the description and representation of linguistic knowledge using language components. This module is capable to describe natural language structure. This description should specify the words ordering, relationships, and dependencies among the constituents of the sentence. Additionally, it provides the proper description of UNL and the structure for mapping it to Arabic.

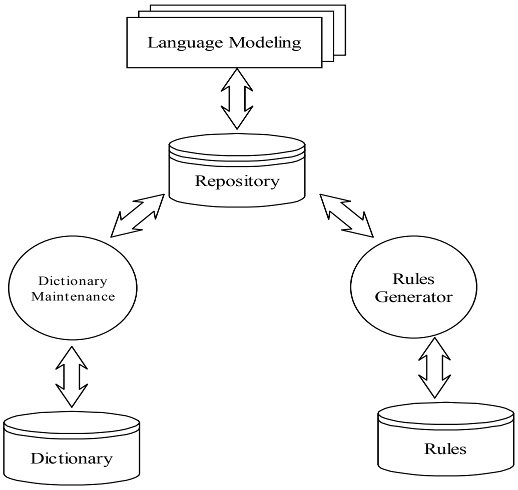


Fig. 14. Basic structure of CARE

As shown in figure 14, linguistics knowledge, UNL, and mapping rules are stored in the repository.

The repository is then interfaced with rules generation component that will facilitate the automatic production of the DeCo rules.

The repository is also interfaced with the dictionary to enable handling and maintenance of the dictionary.

6 Conclusion

In this paper, we have described the development of our first version of an UNL-Arabic Deconversion system. All information for the generation of Arabic from UNL has been addressed in all levels (i.e. morphological and syntactic). We also presented some complexities and issues related to the generation of Arabic. We have tried to introduce some systematicity in using the available DeCo tool, in order to compensate for its lack of high level programming constructs and modularity features. Our future work will concentrate on the development of an adequate CARE environment.

References

1. Uchida, H. (1989). “ATLAS-II: A machine translation system using conceptual structure as an Interlingua”. Proceedings of the Second Machine Translation Summit. Tokyo, Japan.
2. UNL center UNDL Foundation (2003). “The Universal Networking Language Specifications”. http://www.undl.org.
3. UNU/IAS (1999). "DeConverter Specifications. UNU/IAS UNL Center". www.undl. org.
4. Uchida H., Zhu M. (2001), "The Universal Networking Language Beyond Machine Translation". http://www.undl.org.
5. Soudi, A., Cavalli-Sforza, V., & Jamari, A., "Prototype English-to-Arabic Interlingua-based MT System," Proceedings of the Workshop on Arabic Language Resources and Evaluation - Status and Prospects, 3rd International Conference on Language Resources and Evaluation (LREC 2002), Jun 1, 2002, Las Palmas de Canaria, Spain.
6. Abuleil, S. and Evens M. (1998). Discovering Lexical Information by Tagging Arabic Newspaper Text., Workshop on Semitic Language Processing. COLING-ACL.98,
7. G. Sérasset and C. Boitet (1999), “Unl-french deconversion as transfer and generation from an interlingua with possible quality enhancement through offline human interaction,” in MT Summit VII, J.-I. Tsujii, Ed. Singapore: Asia Pacific Ass. For MT, pp. 220–228.
8. G. Sérasset and E. Blanc (2003) , “Remaining issues that could prevent UNL to be accepted as a standard,” in Convergences 3, Library of Alexandria, Egypt.
9. Boguslavsky I. (2001a). UNL from the linguistic point of view. Proceedings of the First International Workshop on MultiMedia Annotation. Electrotechnical Laboratory SigMatics, Tokyo, 1-6.
10. Mel’čuk I. (1988), Dependency Syntax: Theory and Practice, State University of New York Press.

11. Beesley, K.R. (2001). Finite-State Morphological Analysis and Generation of Arabic at Xerox Re-search: Status and Plans in 2001, in Arabic Language Processing: Status and Prospects -39th Annual Meeting of the Association for Computational Linguistics, pp. 1–8.
12. Dichy, J. (2001),"On Lemmatization of the Arabic Entries of Multilingual Lexical Databases, in Arabic Language Processing: Status and Prospects - 39th Annual Meeting of the Association for Computational Linguistics, pp. 23–30.
13. G. Sérasset and C. Boitet (2000), "On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French Deconverter", COLING 2000.
14. Uchida H., Zhu M. (1993), "Interlingua for Multilingual Machine Translation", CICC.

Development of the User Interface Tools for Creation of National Language Modules

Tigran Grigoryan, Vahan Avetisyan

Institute for Informatics and Automation Problems
P. Sevak Str. 1,
375014 Yerevan, Armenia
{va@ipia.sci.am; tigrangr@ipia.sci.am }
http://ipia.sci.am; http://www.unl.am

Abstract. The paper describes the UNL Toolbox, software for development of national language modules of UNL, designed at the Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia. The software provides tools for creating dictionaries, enconversion and deconversion rules. There are also enconversion and deconversion modules which output the converted text and the list of occurred errors and their descriptions (if any). This software also can be used as an educational tool to learn creating UNL dictionaries and conversion rules.

1 System Overview

The UNL Toolbox is an integrated environment for UNL development. It contains tools for performing the most common tasks arising during UNL development such as dictionary creation and conversion rules creation. The Toolbox makes routine tasks like compilation of dictionary and conversion rules transparent to the end user. It allows setting options for an individual tool as well as for a whole system (for example the common output directory).

The main window of the Toolbox is divided into two parts (Fig. 1). On the left side of the window the toolbar is located. Pressing the buttons on the toolbar brings up appropriate tool in the right side of the window. Currently four tools are available – Dictionary Editor, Enconversion Rules Editor, Deconversion Rules Editor and Converter.

2 Dictionary Editor

Dictionary Editor provides a user friendly interface for creating UNL dictionaries and editing the existing ones. It uses XML to store the dictionary. When needed it is possible to export the dictionary in a standard plain text UNL dictionary format.

The dictionary in the dictionary editor has a tree-like structure. Each word is represented as a node of a tree with its stems (if any) represented as child nodes (Fig. 2).

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 210–214.

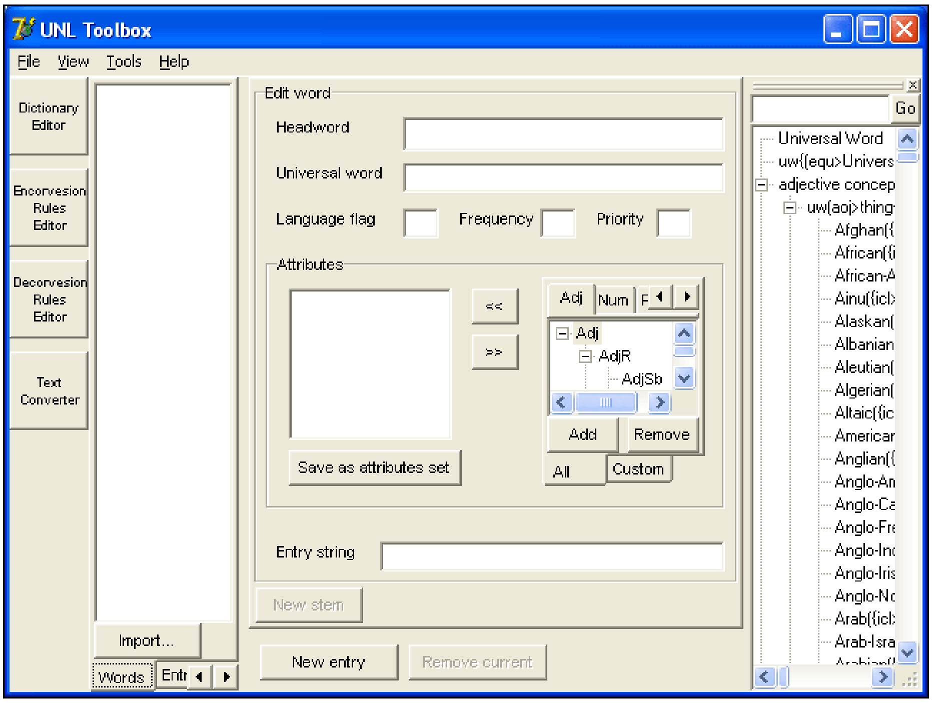


Fig. 1. Main window of the Toolbox.

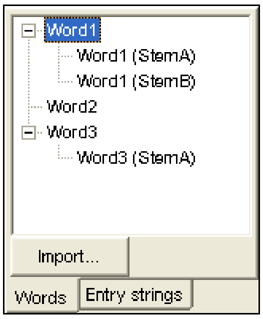


Fig. 2. Representation of a word.

In the UNL dictionary words and their stems are considered as separate dictionary entries, and when exporting the dictionary into plain text format the tree-like structure of the dictionary is lost. The tree-like structure of the dictionary is maintained only for ease of use for the end user. Before saving the dictionary the words are ordered alphabetically, preserving the tree structure.

The dictionary editor window is divided into three areas – dictionary words (on the left), attributes and properties of the word (in the center), and the UNL knowledge base tree (on the right), as shown in Fig. 3. The tree-like structure of the dictionary is shown on the left side of the dictionary editor. It has two views, which are switched using the tabs bellow the tree view. One of the views is called “Words” and it shows only the native language words, another view is called “Entry strings” and it shows entries of the dictionary in the UNL dictionary format (word, UW, attributes, etc.). The ‘Import’ button makes it possible to paste a set of words into a dictionary. When clicking on a word in the dictionary tree, the word becomes active and all its properties are shown in the central part of the window and can be edited. The attributes from the editable language-specific attribute list are assigned to the word and removed from the word using ‘>>’ and ‘<<’ buttons. It is also possible to define custom sets of attributes that are frequently used together (by clicking on the ‘Save as attributes set’ button) and assign an attribute set to the word. Attribute sets appear on the “Custom” tab of attributes tree.

The knowledge base tree (KB) on the right is the UNL knowledge base downloaded from the undl.org. On the top of the KB tree there is a text field for typing in a keyword to be searched in the KB. When the keyword is typed and the ‘Go’ button is clicked the search is performed over the KB tree and the matching (or partially matching) entry is highlighted. Click on the universal word (UW) in the KB will assign that UW to the current word in the dictionary.

If an adequate UW is not found in the KB, a new (made up) UW corresponding to the word can be typed in as a UW property of the word (in the UW textbox in the center). These new UWs are *not* added to the KB.

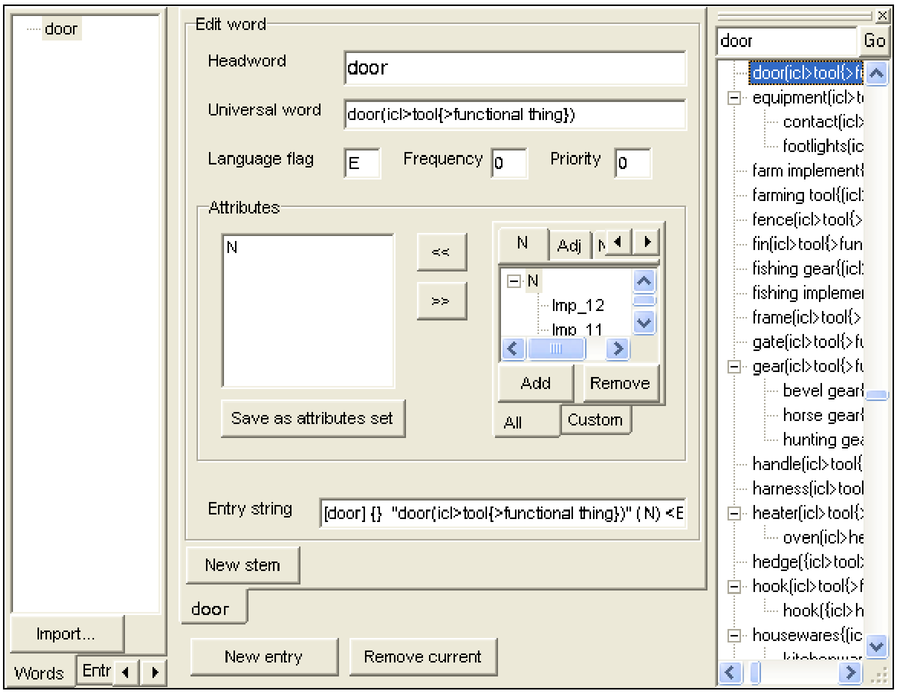
There are two approaches in adding new words to the dictionary: adding a single word and adding more than one word. A single word is added by clicking on the ‘New entry’ button after which a new entry with headword “New word” is added to the dictionary and becomes active so its properties can be set. Multiple words are added by clicking on the ‘Import’ button and pasting their headwords in the appeared popup window. After importing a set of entries each of them must be activated and the properties must be set.

3 Enconversion and Deconversion Rules Editors

Important tools in the UNL Toolbox are enconversion and deconversion rules editors. Although they are separate tools but their functionality is similar. Below the enconversion rules editor will be described (deconversion rules editor is mostly the same). As the structure of enconversion/deconversion rules is pretty complicated for novice users this tool will be very helpful for them. The tool provides an interface to the user for setting conditions and actions on the *condition windows* and *analysis windows*. It is also enables the user to assign labels (names) to the rules making it easier to manage them.

On the left side of the enconversion rules editor there is a list of the defined rules, represented by names (Fig. 4). Clicking on the name of the rule in the list will show the content of the rule in the right side of the window making it available also for editing.

To add a new rule the ‘Add’ button is used. When clicking on it the new rule will be created and will become active for editing.



Headword: door
Universal word: door(icl>tool{>functional thing})
Language flag: E
Frequency: 0
Priority: 0
Entry string: [door] {} "door(icl>tool{>functional thing})" (N) <E

Fig. 3. Dictionary editor.

An existing rule can be removed by clicking on it in the list view and then on the ‘Remove’ button.

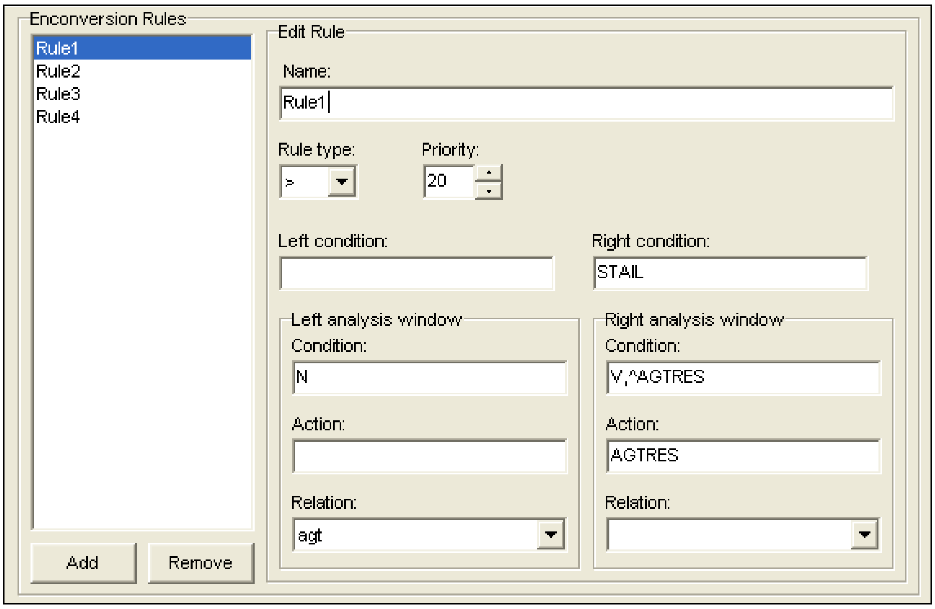
The enconversion and deconversion rules can be exported in the plain text UNL format files which can be used with EnCo and DeCo.

4 Converter

The last tool in the UNL Toolbox is the Converter. The Converter uses external enconversion and deconversion tools, such as EnCo and DeCo (from UNL Development Set) to do native language – UNL and UNL – native language translations and returns the result of the translation. An important feature of the Converter is that it also performs an analysis of the errors occurring during the translation process and represents them in a user friendly style.

5 Use and Future Development of the System

Currently the UNL Toolbox is used for creating the UNL Armenian module. Its user friendly interface and automation of routine tasks greatly increases the speed of dic-



Rule1
Rule2
Rule3
Rule4

Name: Rule1

Rule type: >

Priority: 20

Left condition:

Right condition: STAIL

Left analysis window
Condition: N
Action:
Relation: agt

Right analysis window
Condition: V,^AGTRES
Action: AGTRES
Relation:

Fig. 4. Ruses Editor.

tionary and conversion rules creation. The users of this system also benefit from its error handling, which makes the process of finding errors and typos easier and faster.

The system can also be used as an educational tool for those who are new to UNL. It is easy to understand and to use for novice users. And again the error handling functionality has a great educational role because the user can see where he did the mistake and after correcting it try it again.

Having integrated system that contains dictionary editing, conversion rules editing and converting tools is also beneficiary as it allows performing centralized management of the dictionary and the rules that are used with that particular dictionary. When developing a dictionary or conversion rules it is possible to check the current state of the dictionary and the set of rules using the converter to see if the change we just made brings up any anomalies or not.

The UNL Toolbox is a fully functional system but it is still being improved. The additions to the system will include a support for plug-in modules so that the other tools may be designed for the system and plugged to it as plug-in modules.

References

1. Uchida H., Zhu M., The Universal Networking Language (UNL) specifications version 3.0,*1998*. Technical Report, United Nations University, Tokyo, 1998.
2. EnConverter Specification Version 2.1, UNU/IAS/UNL Centre, Tokyo 150-8304, Japan; http://www.unl.ias.unu.edu/unlsys/enco/index.html, 2000.

Universal Networking Language Based Analysis and Generation for Bengali Case Structure Constructs

Kuntal Dey[1] and Pushpak Bhattacharyya[2]

[1] Veritas Software, Pune, India.
u2ckuntal@yahoo.com
[2] Computer Science and Engineering Department
Indian Institute of Technology, Bombay, India.
pb@cse.iitb.ac.in

Abstract. Case structure analysis forms the foundation for any natural language processing task. In this paper we present the computational analysis of the complex case structure of Bengali- a member of the Indo Aryan family of languages- with a view toward interlingua based MT. Bengali is ranked 4^{th} in the list of languages ordered according to the size of the population that speaks the language. Extremely interesting language phenomena involving morphology, case structure, word order and word senses makes the processing of Bengali a worthwhile and challenging proposition. A recently proposed scheme called the *Universal Networking Language* has been used as the interlingua. The approach is adaptable to other members of the vast Indo Aryan language family. The parallel development of both the analyzer and the generator system leads to an insightful intra-system verification process in place. Our approach is *rule based* and makes use of authoritative treatises on Bengali grammar.

1 Introduction

Bengali is spoken by about 189 million people and is ranked 4^{th} in the world in terms of the number of people speaking the language (ref: *http://www.harpercollege.edu/˜mhealy/g101ilec/intro/clt/cltclt/top100.html*). Like most languages in the Indo Aryan family, descended from Sanskrit, Bengali has the SOV structure with some typical characteristics. A motivating factor for creating a system for processing Bengali is the possibility of laying the framework for processing many other Indian languages too.

Work on Indian language processing abounds. *Project Anubaad* [1] for machine translation from English to Bengali in the newspaper domain uses the *direct translation approach. Angalabharati* [2] system for English Hindi machine translation is based on pattern directed rules for English, which generates a *pseudo-target-language* applicable to a group of Indian Languages. In MATRA [3], a web based MT system for English to Hindi in the newspaper domain, the input text is transformed into case-frame like structures and the the target

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 215–229.

language is generated by parameterized templates. The *MANTRA* MT system for official documents uses Tree Adjoining Grammar (TAG) to achieve English Hindi MT (ref: *http://www.cdacindia.com/html/about/success/mantra.asp*). Project *Anusaaraka* [4] is a language accessor system rather than an MT system and addresses multiple Indian languages. Interlingua based MT for English, Hindi and Marathi [5] [6], that uses the UNL, transforms the source text into the *UNL representation* and generates target text from this intermediate representation. References to most of these works can also be found at *http://www.tdil.mit.gov.in/mat/ach-mat.htm.* Other famous MT systems are *Pivot [7], Atlas [8], Kant [9], Aries [10], Geta [11], SysTran [12]* etc.

The Universal Networking Language (UNL) (*http://www.unl.ias.unu.edu*) has been defined as a digital meta language for describing, summarizing, refining, storing and disseminating information in a machine independent and human language neutral form. The information in a document is represented sentence by sentence. Each sentence is converted into a directed hyper graph having concepts as nodes and relations as arcs. Knowledge within a document is expressed in three dimensions:

1. Word Knowledge is expressed by Universal Words (UWs) which are language independent. These UWs are tagged using restrictions describing the sense of the word in the current context. For example, $drink(icl > liquor)$ denotes the noun sense of *drink* restricting the sense to a type of *liquor.* Here, *icl* stands for inclusion and forms an *is-a* relationship like in semantic nets [13].
2. Conceptual Knowledge is captured by relating UWs through a set of UNL relations [14]. For example,

   Humans affect the environment

   is described in the UNL as

```
agt(affect(icl>do).@present.@entry, human(icl>animal).@pl)
obj(affect(icl>do).@present.@entry, environment(icl>abstract thing).@pl)
```

   agt means the *agent* and *obj* the *object.* $affect(icl > do)$, $human(icl > animal)$ and $environment(icl > abstract\ thing)$ are the UWs denoting concepts.
3. Speaker's view, aspect, time of event, etc. are captured by UNL attributes. For instance, in the above example, the attribute *@entry* denotes the main predicate of the sentence, *@present* the present tense and *@pl* the plural number.

The above discussion can be summarized using the example below

John, who is the chairman of the company, has arranged a meeting at his residence

The UNL for the sentence is

```
  ;======================== UNL ======================
  mod(chairman(icl>post).@present.@def,company(icl>institution).@def)
  aoj(chairman(icl>post).@present.@def, John(icl>person))
  agt(arrange(icl>do).@entry.@present.@complete, John(icl>person))
  pos(residence(icl>shelter), John(icl>person))
obj(arrange(icl>do).@entry.@present.@complete, meeting(icl>event).@indef)
  plc(arrange(icl>do).@entry.@present.@complete, residence(icl>shelter))
  [/S]
  ;==================================================
```

In the expressions above, *agt* denotes the *agent* relation, *obj* the *object* relation, *plc* the *place* relation, *pos* is the *possessor* relation, *mod* is the *modifier relation* and *aoj* is the *attribute-of-the-object* (used to express constructs like *A is B*) relation. The detailed specification of the Universal Networking Language can be found at *http://www.unl.ias.unu.edu/unlsys*.

Our work is based on an authoritative treatise on Bengali grammar [15]. The strategies of analysis and generation of linguistic phenomena have been guided by rigorous grammatical principles.

2 EnConverter and DeConverter machines

The EnConverter (henceforth called *EnCo*) [16] is a language-independent parser, a multi-headed Turing machine [17] providing a framework for morphological, syntactic and semantic analysis synchronously using the UW dictionary and analysis rules. The structure of the machine is shown in the figure 1.

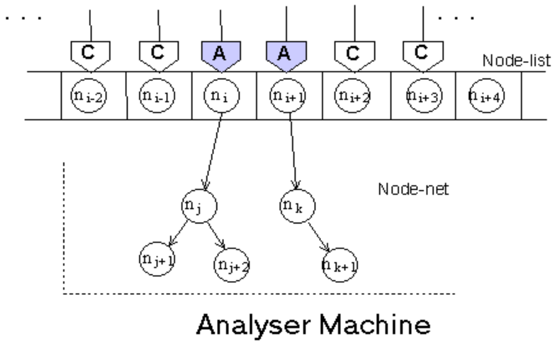


Fig. 1. The EnCo machine

The machine has two types of *heads- processing heads* and *context heads.* The processing heads (2 nos.) are called *Analysis Windows (AW)* and the context heads are called *Condition Windows (CW)*. The machine traverses the sentence back and forth, retrieves the relevant universal words from the lexicon and, depending on the *attributes* of the nodes under the AWs and those under the surrounding CWs, generates semantic relations between the UWs and/or attaches speech act attributes to them. The final output is a set of UNL expressions equivalent to a UNL graph.

The DeConverter (henceforth called the *DeCo*) [18] is a language-independent generator that produces sentences from UNL graphs (figure 2).

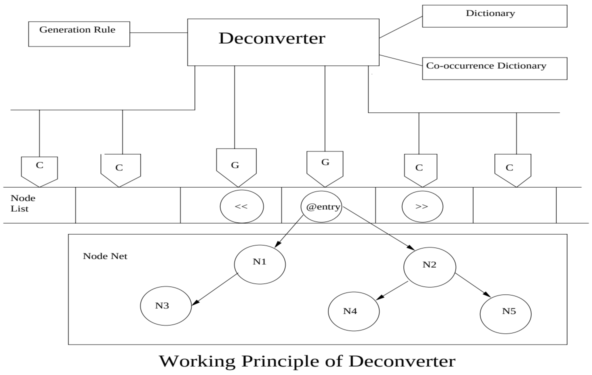


Fig. 2. The DeCo machine

Like EnCo, DeCo too is a multi-headed Turing Machine. It does syntactic and morphological generation synchronously using the lexicon and the set of generation rules.

3 Rule theory

EnCo and DeCo are driven by *analysis rules* and *generation rules* respectively. These rules are *condition-action structures* that can be looked upon as *program* written in a specialized language to process various complex phenomena of a natural language, both for analysis and generation. They have the following format:

```
< TYPE >
["(" < PRE > ")"["*"]]...
"{"||""""[< COND1 >]":"[< ACTION1 >]":"[< RELATION1 >]":"[< ROLE1 >]"}"||""""
["(" < MID > ")"["*"]]...
"{"||""""[< COND2 >]":"[< ACTION2 >]":"[< RELATION2 >]":"[< ROLE2 >]"}"||""""
["(" < SUF > ")"["*"]]...
"P("< PRIORITY >");"
```

Characters between double quotes are the predefined delimiters of the rule. The rules mean that

- **IF**
  under the *left processing window* there is a node satisfying <COND1> and under the *right processing window* a node satisfying <COND2> attributes, and there are nodes that fulfill the conditions in <PRE>, <MID> and <SUF> in the order of left, middle and right sides of processing windows respectively,
  THEN
  the lexical attributes in processing windows are rewritten according to the <ACTION1> and <ACTION2> as specified in rule, and new attributes added if necessary. (By *processing window*, *analysis window* is meant for the enconversion process and *generation window* for the deconversion process).
- The operations are done on the node-list depending on the <TYPE> of the rule. <RELATION1> describes the semantic relation of the node on right processing window to the node on left processing window and <RELATION2> describes the reverse [6].
- <PRIORITY> describes the interpretation order of the rules, whose value lies between 0-255. Larger number indicates higher priority. Matching rule with the highest priority is selected for multiple matching rules.

A sequence of such rules get activated depending on the sentence situation (the conditions of the nodes under the analysis/generation windows). These are the lexico-morpho-grammatical-semantic attributes of the words under processing. For example, for a sentence like *John laughs*, the *animate* attribute of *John*, the *verb* attribute of *laugh* and the *adjacency* of these two words under the analysis windows dictate with high probability establishing the *agt (agent)* relation between the corresponding two nodes in the UNL graph.

In order to adapt the UNL engines to enconvert the Bengali sentences into the UNL interlingua and to deconvert the UNL interlingua/graph into Bengali sentences, an enconverter rule-base and a deconverter rule-base have been written. The rules within the rule-base are compliant with the corresponding UNL engines and are focused to deal with the Bengali language structure.

4 Case Structure in Bengali: *Kaarak*s

In the Indian linguistic system- descended from Sanskrit- the *case constructs* are called *kaarak*s [19]. As in the traditional understanding, they denote the relationship of the nominals with the main verb of the clause except in the *genitive case* where two nominals are related to each other. The case structure in Bengali is complex. The *kaarak*s are broadly classified into 6 types [15], each having a finer categorization into sub-types. The correspondence between the Bengali *kaarak* system and the traditional linguistic concept of case [20] is shown by means of table 1. The *Bibhakti signs* are the case markers. An exhaustive study of the *kaarak* system with a view to analyzing Bengali into UNL has been carried out. The foundation of this work is the *kaarak* theory [15]. Due to the word limitation, we exemplify the work with only the first *kaarak, viz., the kartri kaarak.*

Table 1. Case-*kaarak* correspondence

Classical Case	**Corresponding Bengali kaarak**	**Bibhakti signs** (Case Marker)
Nominative case	*Kartri kaarak*	None
Accusative case	*Karma kaarak*	*ke, re, ere*
Instrumental case	*Karan kaarak*	*dwaaraa, diye, diya, kartrik*
Dative case	*Sampradaan kaarak*	*janya, nimitta, ke*
Ablative case	*Apaadaan kaarak*	*theke, haite*
Genitive case	*Sambandha pad*	*r, er*
Case of time and place	*Adhikaran kaarak*	*e, te, ete*

4.1 *Kartri* kaarak

Kartri kaarak denotes the *agent* of the action stated by the verb. The *kaarak* is divided into the following classes:

1. **Projojak karta** (প্রযোজক কর্তা): Here the agent *causes* some event to take place, with an inclination towards compelling the event to happen. The morphology of the verb is exploited and the extracted knowledge has the *causative* feature marked.
   Example:
   টম জনকে খেলাবে
   tama janake khelaabe.
   Tom John-to will-make-play.
   Tom will make John play.
2. **Nirapekkha karta** (নিরপেক্ষকর্তা): Here there are more than one verb in the sentence with at least one অসমাপিকা (finite) verb and one সমাপিকা (non-finite) verb, and the *karta*s, *i.e., agents* for these verbs are different or not related. The *karta* associated with the non-finite verb is called the *nirapekkha karta* (*nominative absolute* in English). As there is an অসমাপিকা verb involved, a *con* or *seq etc.* relation is generated, also there is a possible generation of compound UW.
   Example:
   টম খেলে জন খাবে
   tama khele jana khaabe.
   Tom if-eats John will-eat.
   If Tom eats John will eat.

3. **Karmakartribachchyer karta** (কর্মকর্তৃবাচ্যের কর্তা): Here, the actual *karta* is not present, and hence the *karma, i.e., the object* acts as the *karta.* As a result, there is no *agt* or equivalent relation generated for conceptualizing an agent of the sentence, instead, an *obj* relation is realized.
   Example:
   বালতি ভরেছে
   baalti bhareche.
   Bucket has-filled-up.
   The bucket has filled up.
4. **Anukta karta** (অনুক্ত কর্তা): In cases of কর্মবাচ্য (*karma bachya*) and ভাববাচ্য (*bhaab baachya*) (which are variants of the passive voice), the karta is not emphasized on.
   Example:
   টমের আজ খাওয়া হয়নি
   tamer aaj khaaoyaa hay ni.
   Tom-of today eating not-happened.
   Eating has not happened to Tom today.
5. **Sahajogi karta** (সহযোগী): Two *karta*s are present in the same sentence, co-acting with each other to perform the action specified by the verb.
   Example:
   বাঘে গোরুতে খাচ্ছে
   baaghe gorute khaacche.
   Tiger cow eating.
   Tiger is eating with cow.
6. **Bakyangsha karta** (বাক্যাংশ কর্তা): Here the noun phrase as a unit acts as the *karta.* A noticeable fact is that this noun phrase does not have any সমাপিকা (finite) verb.
   Example:
   সৎপথে জীবনযাপন করা কঠিন কাজ
   satpathe jiibanjaapan karaa kathin kaaj.
   Honest-way-in leading-life hard work.
   Leading a life in an honest way is hard work.
   (Note: Here *hard work* means *difficult.*)
7. **Upabakyiya karta** (উপবাক্যীয় কর্তা): Here there is a noun clause in the sentence. This noun clause conceptually acts as the *karta.* However, in order to retain the *person* information present in the verb, a different term causing *agt* relation has to be introduced in the sentence during enconversion. The conceptual *karta* actually does not get identified as a *karta*, instead it is identified as something different (for example, *karma*).
   Example:
   ভয় কাকে বলে জানি
   bhay kaake bale jaani.
   Fear to-whom call I-know.
   I know what is called fear.
8. **Karta with 'e' bibhakti** (কর্তায় এবিভক্তি): In spite of the presence of the *e* (এ) bibhakti, the *karta* has to be identified as an *agt* or equivalent relation. A

salient point to note is that the *e* bibhakti can be used with all other *kaarak*s as well, so appropriate analysis has to be done to identify its functionality. Often the context of occurrence of the word and the grammatical attributes available with the word from the lexical dictionary guide in identifying the *kaarak* in case of *e bibhakti.*
Example:

ছাগলে	ঘাস	খায়
chaagale	ghaash	khaay.
Goat	grass	eat.

Goat eats grass.

(UNL relations generated for *kartri kaarak*: *agent (agt), co-agent (cag), partner (ptn) etc.*).

4.2 Other *kaarak*s

Five other *kaarak*s have been analyzed exhaustively as above.

1. *Karma* kaarak (6 subcategories): *Karma kaarak* is the person or thing on which the *kartri kaarak* executes the action stated by the sentence.
   (UNL relations for *karma kaarak*: *object (obj), beneficiary (ben), co-object (cob)*).
2. *Karan* kaarak (5 subcategories): *Karan kaarak* is the thing or tool or method by which the *kartri kaarak* of the sentence executes the specified action.
   (UNL relations for *karan kaarak*: *instrument (ins), method (met)*).
3. *Sampradaan* kaarak (2 subcategories): *Sampradaan kaarak*s are cases where the agent (*kartri kaarak*) does something for someone or gives away something to someone.
   (UNL relations for *sampradaan kaarak*: *beneficiary (ben), goal (gol), purpose (pur), reason (rsn)*).
4. *Apaadaan* kaarak (6 subcategories): This stands for the concept of sources of creation, location, position *etc.* All types of relations bearing the concept of *source* in some sense are eligible to come into this category.
   (UNL relations for *apaadaan kaarak*: *place-from (plf), time-from (tmf), from (frm), source (src).*).
5. *Sambandha pad* (4 subcategories): If related to the next noun or pronoun, then the term having a *r* (র) or *er* (এর) bibhakti is called a *sambandha pad. Sambandha pad* always has some *bibhakti* with it (never *sunya bibhakti*).
   (UNL relations for *sambandha pad*: *modifier (mod), possession (pos), part-of (pof).*)
6. *Adhikaran* kaarak (8 subcategories): *Adhikaran kaarak*s are the ones that describe the place, time and topic of the action performed by the sentence.
   (UNL relations for *adhikaran kaarak*: *place (plc), time (tim), place-to (plt), time-to (tmt), to (to), goal (gol), virtual-place (scn), objectified-place (opl).*)
7. *Sambodhan* (3 subcategories): *Sambodhan* (সম্বোধন) is the case where someone hails some other person and says something to this person. This act of hailing

is captured by what is called সম্বোধন. This generates a *@vocative* attribute against the called person's appearance in the UNL graph.

Table 2 summarizes the correspondence between Bengali *kaaraks* and the *UNL relations.*

Table 2. Correspondence beteween *kaarak* and UNL relations

Kaarak	**Corresponding UNL Relations**
Kartri kaarak	agt, cag, ptn, aoj, cao
Karma kaarak	obj, ben, cob
Karan kaarak	ins, met
Sampradaan kaarak	ben, gol, pur, rsn
Apaadaan kaarak	frm, src, plf, tmf
Sambandha pad	mod, pos, pof
Adhikaran kaarak	plc, plt, tim, tmt, to, gol, scn, opl

The UNL relations that are not covered by the *kaarak*s in Bengali are: *and (and), or (or), quantity (qua), proportion, rate or distribution (per), content (cnt), via (via), condition (con), sequence (seq), co-occurrence (coo), basis for expressing degree (bas), duration (dur), range: from-to (fmt)* and *manner (man).*

5 *Kaarak* enconversion strategy

The basic idea is as follows. The non-verb primary (non-case [21]) words appearing in the sentences are one of the two types: (i) A word denoting a concept, which is a *kaarak* or *sambandha pad* or *sambodhan*, (ii) A word or *bibhakti* causing a conceptual relation to link two concepts.
The *kaarak*s, *sambandha pad*s and *sambodhan*s get mapped to the UNL word concepts (UWs) after the analysis and appear in the UNL graph as **node**s. The *bibhakti*s or conceptually relating words result in forming the **edge**s of the graph which embed the logical relation between the two word-concepts. Also, there are lexical, morphological and semantic attributes in the dictionary entries of the word-concepts, which too are used to analyze the input. We illustrate the approach with an example:
কীর্তনে এবং বাউল গানে আমি মাতিয়ে রাখবো (Input to enconverter)
kiirtane ebang baaul gaane aami maatiye raakhbo
Kiirtan-by and baaul song-by I enchant-will
I will enchant with Kirtan and baaul song
Strategy:

- When the *e* (এ) *bibhakti* is added to and abstract noun, it becomes a candidate for the *met* relation, and hence, a *+MET* is added to it.

– Finally, a *met* relation gets resolved when the node having the *MET* attribute and the verb becomes juxtaposed.

Salient rules:

– +{N,Na,ABS,ˆPLACE,ˆCONCRETE,ˆSCN,ˆRSN,ˆTIME,
ˆBLKINSERT:+MET,+MORADD,+eADD,+BLKINSERT::}
{[[*e*]],NMOR,BLKINSERT:::}P30;
– >{N,MET,ABS,ˆV::met:}{V,ˆMETRES,:+METRES::}P20;

UNL:
met(enchant(icl>do):0T.@entry.@future,:01)
agt(enchant(icl>do):0T.@entry.@future,I(icl>person):0P)
and:01(song(icl>song):0K.@entry,kirwana(icl>song):00)
mod:01(song(icl>song):0K.@entry,bAula(icl>song):0E)

This example gives a flavor of the procedure involved. Similar procedure has been applied all the various categories and subcategories. (Note: *Kirtan* and *baaul* are two Indian blends of songs.)

6 Verification

An exhaustive verification of the system has been carried out by writing a **UNL to Bengali Deconverter** (*i.e.* generator). This uses the same lexicon as the *Bengali enconversion* system and a set of *Bengali generation rules*. The enconverted input sentences have been re-generated from the UNL graphs and manually matched for conceptual equivalence. This is a form of intra-platform verification, which verifies both the preservation of information and meaning during enconversion and its wholesome retrieval during deconversion using the appropriate rule-bases. Some examples follow. Many of the output sentences map back exactly to the same set of words and sentence structure as the input, without any divergences. However, to provide a more interesting delineation (within this short span of space) of the challenges faced, we mainly give the instances of input output divergence.

1. **Projojak karta** (প্রযোজক কর্তা):
Input to enco: tama janake khelaabe
Equivalent: টম জনকে খেলাবে
Gloss: Tom John-to will-make-play

Meaning: Tom will make John play.

Output of deco: tama janake khelaabe
Equivalent: টম জনকে খেলাবে
Gloss: Tom John-to will-make-play

Remark: Exact match between input and output sentences.

2. **Nirapekkha karta** (নিরপেক্ষকর্তা):
   Input to enco: tama khele jana khaabe
   Equivalent: টম খেলে জন খাবে
   Gloss: Tom if-eats John will-eat

   Meaning: John will eat if Tom eats.

   Output of deco: jadi tama khaay jana khaabe
   Equivalent: যদি টম খায় জন খাবে
   Gloss: If Tom eats John will-eat

   Remark: This is an interesting case where the *jadi* (*if*) clause has got introduced into the output of the deconverter while it was not explicitly present in the input to the enconverter. However, it is correct as these sentences have the same sense conceptually.

3. **Upabakyiya karta** (উপবাক্যীয় কর্তা):
   Input to enco: bhay kaake bale jaani
   Equivalent: ভয় কাকে বলে জানি
   Gloss: Fear to-whom call I-know

   Meaning: (I) know what is called fear.

   Output of deco: aami jaani bhay kaake bale
   Equivalent: আমি জানি ভয় কাকে বলে
   Gloss: I know fear to-whom call

   Remark: An explicit *aami (I)* has been introduced in the generated sentence.

4. **Bakyangsha karma (noun phrase as an object)** (বাক্যাংশ কর্ম):
   Input to enco: aamtaa aamtaa kathaa balte bhaalobaasi naa
   Equivalent: আমতা আমতা কথা বলতে ভালোবাসি না
   Gloss: Soft soft to-talk I-like not

   Meaning: (I) don't like to talk softly.

   Output of deco: aami bhaalobaasi naa aamtaa aamtaa kathaa balte
   Equivalent: আমি ভালোবাসি না আমতা আমতা কথা বলতে
   Gloss: I like not soft soft to-talk

   Remark: Conceptually these are the same, although the structures differ and order in the generated sentence is not normal in Bengali prose.

5. **Karmer bipsaa** (কর্মের বীপ্সা) (Repetition in Karma):
   Input to enco: kii kii caao bali

Equivalent: কী কী চাও বলি
Gloss: What what you-want I-say

Meaning: (I)/(Let me) say what (you) want.

Output of deco: aami bali tomraa kii kii caao
Equivalent: আমি বলি তোমরা কী কী চাও
Gloss: I say you what what want

Remark: The input to enco has no default number information associated with the person, so the output generates (by default implementation as per the rule base) a singular number output for the first person and a plural number output for the second person. As it can be seen, an *aami*, which means *I* (first person singular number) and a *tomraa*, which means *you* (second person plural number), have been explicitly added to the output.

6. **Karaner bipsaa** (করণের বীপ্সা) (Repetition in Karan):
Input to enco: taaraay taaraay bharaa raater aakaash
Equivalent: তারায় তারায় ভরা রাতের আকাশ
Gloss: Star-with star-with filled night's sky

Meaning: (The) night's sky is filled with stars.

Output of deco: raater aakaash taaraader diye bhareche
Equivalent: রাতের আকাশ তারাদের দিয়ে ভরেছে
Gloss: Night's sky stars-with has-filled

Remark: The structural and morphological differences in the input and output here is noticeable, although the conceptual meanings are the same in both the cases.

7. **Sunya bibhakti (no bibhakti) in karan** (করণে শূন্যবিভক্তি):
Input to enco: gaadhaake haajaar caabuk maarleo se ghozDaa hay naa
Equivalent: গাধাকে হাজার চাবুক মারলেও সে ঘোড়া হয় না
Gloss: Donkey-to thousand whiplash in-spite-of-beating-with it horse become not

Meaning: In spite of thousand beatings with whiplashes a donkey does not become a horse.

Output of deco: jadi tomraa haajaar caabuk diye gaadhaake maaro tabuo se ghozDaa hay naa
Equivalent: যদি তোমরা হাজার চাবুক দিয়ে গাধাকে মারো তবুও সে ঘোড়া হয় না

Gloss: If you thousand whiplash with donkey-to beat yet it horse become not

Remark: Here the output is a complex sentence while the input is not, yet they mean the same in terms of concept.

8. **Asamaapikaa kriyaa baachak (infinite verb-related) apaadaan kaarak** (অসমাপিকা ক্রিয়াবাচক অপাদান কারক):
Input to enco: aami marte bhiita nai
Equivalent: আমি মরতে ভীত নই
Gloss: I to-die afraid not

Meaning: I am not afraid to die.

Output of deco: maraar janya aami bhiita nai
Equivalent: মরার জন্য আমি ভীত নই
Gloss: To-die I afraid not

Remark: These, again, differ in the *anusarga* (*janya* in the output), but the input means the same in Bengali as the output in spite of this difference in construction.

9. **Saamipya suchak (proximity-denoting) adhikaran kaarak** (সামীপ্যসূচক অধিকরণ কারক):
Input to enco: tama darajaay daazDiye brishti dekhche
Equivalent: টম দরজায় দাড়িয়ে বৃষ্টি দেখছে
Gloss: Tom at-door standing rainfall seeing

Meaning: Tom is seeing rainfall standing at the door.

Output of deco: tama darajaay daazDiye daazDiye brishti dekhche
Equivalent: টম দরজায় দাড়িয়ে দাড়িয়ে বৃষ্টি দেখছে
Gloss: Tom at-door standing standing rainfall seeing

Remark: These two mean the same, although the word *daazDiye* has come in twice in the deconverter output (to ensure the *coo* concept) in spite of the fact that it was present only once in the input to the enconverter.

10. **Bishayaadhikaran (topic denoting adhikaran) kaarak** (বিষয়াধিকরণ কারক):
Input to enco: se taase pokta ebang futbale ostaad
Equivalent: সে তাসে পোক্ত এবং ফুটবলে ওস্তাদ
Gloss: He in-cards solid and in-football expert

Meaning: He is solid in cards and expert in football.

Output of deco:	futbale	ostaad	ebang	se	taase	pokta
Equivalent:	ফুটবলে	ওস্তাদ	এবং	সে	তাসে	পোক্ত
Gloss:	In-football	expert	and	he	in-cards	solid

Remark: This is an instance of free-format input natural language, where the output structure has significantly varied from the input structure, in spite of having the same meaning and hence being correct.

7 Conclusion

Systematic analysis of the case structure forms the foundation for any natural language processing system. In this paper, we have described a system for the computational analysis of the Bengali case structure for the purpose of interlingua based MT using UNL. The complementary generator system too has been implemented, which provides the platform for intra system verification. Verification via cross system generation is being done using the Hindi generation system (also under development.) Apart from the case structure, computational analysis based on authoritative grammatical treatise, addressing complex phenomena involving verbs, adjectives and adverbs is under way.

References

1. Dey, K.: Project Anubaad: an English-Bengali MT system. Jadavpur University, Kolkata (2001)
2. Sinha, R.: Machine translation: The Indian context. AKSHARA'94, New Delhi (1994)
3. Rao, D., Mohanraj, K., Hedge, J., Mehta, V., Mahadane, P.: A practical framework for syntactic transfer of compound-complex sentences for English-Hindi machine translation. (2000)
4. Bharati, A., Chaitanya, V., Sanyal, R.: Natural Language Processing: A Paninian Perspective. Prentice Hall India Private Limited (1996)
5. Dave, S., Bhattacharya, P., Girishbhai, P.J.: Interlingua based English-Hindi machine translation and language divergence. Journal of Machine Translation, Volume 17 (2002)
6. Monju, M., Sachi, D., Bhattacharyya, P.: Knowledge extraction from Hindi texts. Knowledge Based Computer Systems, Proceedings of the International Conference KBCS2000 (2000)
7. Muraki, K.: Pivot: Two-phase machine translation system. MT Summit Manuscripts and Program, pp. 81-83 (1987)
8. Uchida, H.: Atlas. MT Summit II, pp. 152-157 (1989)
9. Lonsdale, D.W., Franz, A.M., Leavitt, J.R.R.: Large-scale machine translation: An interlingua approach. (www.lti.cs.cmu.edu/Research/Kant/PDF/aei94.pdf)
10. Gonzlez, J.C., Go, J.M., Nieto, A.F.: Aries: A ready for use platform for engineering Spanish-processing tools. Digest of the Second Language Engineering Convention, pages 219-226 (1995)
11. Vauquois, B., Boitet, C.: Automated translation at Grenoble University. (acl.ldc.upenn.edu/J/J85/J85-1003.pdf)

12. W.John, H., L., S.H.: An introduction to Machine Translation. London: Acamedic Press (1992)
13. Woods, W.: What's in a link: Foundations for semantic networks. (Brachman RJ and Levesque HJ, editors, Readings in Knowledge Representation, pp 218-241)
14. Foundation, U.C.: The Universal Networking Language (UNL) Specifications. (2001)
15. Chakrabarti, B.: Uchchatara Bangla Byakaran. Akshay Malancha (1963)
16. UNU/IAS: EnConverter. UNL Centre/UNDL Foundation (2001)
17. E., H.J., D., U.J.: Introduction to Automata Theory, Languages and Computation. Addision-Wiseley Publishing Company (1989)
18. UNU/IAS: DeConverter Specifications. UNU/IAS UNL Center (1998)
19. Shastri, C.D.: Panini Re-interpreted. Motilal Banarasidass, New Delhi (1990)
20. Fillmore, C.J.: The case for case. E Bach and R Harms (eds.), New York: Holt, Rinehart and Winston (1968)
21. Dey, K., Dubey, S.K., Bhattacharyya, P.: Knowledge extraction from Indo-Aryan family of languages using a rule based approach. ICON-2002 (2002)

Interactive Enconversion by Means of the Etap-3 System

Igor M. Boguslavsky, Leonid L. Iomdin and Victor G. Sizov

Institute for Information Transmission Problems RAS, 19, Bolshoj Karetnyj, GSP-4
Moscow, Russia,
{bogus,leonid,sizov}@iitp.ru

Abstract. A module for enconversion of NL texts into Universal networking Language (UNL) graphs is considered. This module is designed for the system of multi-lingual communication in the Internet that is being developed by research centers of about 15 countries under the aegis of UN. The enconversion of NL texts into UNL is carried out by means of a multi-functional linguistic processor ETAP-3, developed in the Computational linguistics laboratory of the Institute for Information Transmission Problems of the Russian Academy of Sciences. One of the major problems in the automatic text analysis is high degree of ambiguity of linguistic units. The resolution of this ambiguity (morphological, syntactic, lexical, translational) is partly ensured by the linguistic knowledge base of ETAP-3, but complete algorithmic solution of this problem is unfeasible. We describe an interactive system that helps resolve difficult cases of linguistic ambiguity by means of a dialogue with the human.

1 Introductory Remarks

ETAP-3 is a multipurpose NLP environment that was conceived in the 1980s and has been worked out in the Institute for Information Transmission Problems, Russian Academy of Sciences ([1], [2], [7]). The theoretical foundation of ETAP-3 is the Meaning ⇔ Text linguistic model by Igor' Mel'čuk and the Integral Theory of Language by Jurij Apresjan. ETAP-3 is a non-commercial environment primarily oriented at linguistic research rather than creating a marketable software product. The main focus of the research carried out with ETAP-3 is computational modelling of natural languages. All NLP applications in ETAP-3 are largely based on a three-value logic and use an original formal language of linguistic descriptions, FORET.

2 Briefly on ETAP-3

The major NLP modules of ETAP-3 are as follows:

- Machine Translation System
- Natural Language Interface to SQL Type Databases
- System of Synonymous Paraphrasing of Sentences
- Syntactic Error Correction Tool

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 230–240.

- Computer-Aided Language Learning Tool
- Tree Bank Workbench
- UNL Deconverter and Enconverter.

The following are the most important features of the whole ETAP-3 environment and its modules:

- Rule-Based Approach
- Stratificational Approach
- Transfer Approach
- Syntactic Dependencies
- Lexicalistic Approach
- Multiple Translation
- Maximum Reusabilty of Linguistic Resources

In the current version of ETAP-3, its modules that process NL sentences are strictly rule-based. ETAP-3 shares its stratificational feature with many other NLP systems. It is at the level of the normalized, or deep syntactic, structure that the transfer from the source to the target language takes place in MT. ETAP-3 makes use of syntactic dependency trees for sentence structure representation instead of constituent, or phrase, structure. The ETAP-3 system takes a lexicalistic stand in the sense that lexical data are considered as important as grammar information. A dictionary entry contains, in addition to the lemma name, information on syntactic and semantic features of the word, its subcategorization frame, a default translation, rules of various types, and values of lexical functions for which the lemma is the keyword. The word's **syntactic features** characterize its ability/non-ability to participate in specific syntactic constructions. A word can have several syntactic features selected from a total of more than 200 items. **Semantic features** are needed to check the semantic agreement between the words in a sentence. The **subcategorization frame** shows the surface marking of the word's arguments (in terms of case, prepositions, conjunctions, etc.). **Rules** are an essential part of the dictionary entry. All rules operating in ETAP-3 are distributed between the grammar and the dictionary. Grammar rules are more general and apply to large classes of words, whereas the rules listed or simply referred to in the dictionary are restricted in their scope and only apply to small classes of words or even individual words. This organization of the rules ensures self-tuning of the system to the processing of each particular sentence. In processing a sentence, only those dictionary rules are activated that are explicitly referred to in the dictionary entries of the words making up the sentence.

3 ETAP-3 and UNL

It would be out of place to present here the whole UNL system, its underlying philosophy, language design, and the current state of system development. We refer the readers first of all to the publications by the author of UNL Hiroshi Uchida and other data that can be found at the UNL official site *http://www.undl.org*. Our purpose is to describe the UNL module of ETAP-3, and, in particular, the UNL enconverter, i.e. the system that receives a natural language sentence at the input and produces a UNL graph at the output.

Since ETAP-3 is an advanced NLP system based on rich linguistic knowledge, it is natural to maximally re-use its linguistic knowledge base and the whole architecture of the system in this new application. Our approach (described in detail in [8]) is to build a bridge between UNL and one of the internal representations of ETAP, namely Normalized Syntactic Structure (NormSS), and in this way link UNL with all other levels of text representation, including the conventional orthographic form of the text.

The level of NormSS is best suited for establishing correspondence with UNL, as UNL expressions and NormSS show striking similarities. The most important of them are as follows:

- Both UNL expressions and NormSSs occupy an intermediate position between the surface and the semantic levels of representation. They roughly correspond to the so-called deep-syntactic level. At this level the meaning of lexical items is not decomposed into the primitives, and the relations between lexical items are language independent.
- The nodes of both UNL expressions and NormSSs are terminal elements (lexical items) and not syntactic categories.
- The nodes carry additional characteristics (attributes).
- The arcs of both structures are non-symmetrical dependencies.

At the same time, UNL expressions and NormSSs differ in several important respects:

- All the nodes of NormSSs are lexical items, while a node of a UNL expression can be a sub-graph.
- Nodes of a NormSS always correspond to one word sense, while UWs may either be broader or narrower than the corresponding English words.
- A NormSS is the simplest of all connected graphs - a tree, while a UNL expression is a hyper-graph. Its arcs may form a loop and connect sub-graphs.
- The relations between the nodes in a NormSS are purely syntactic and are not supposed to convey a meaning of their own, while the UNL relations denote semantic roles.
- Attributes of a NormSS mostly correspond to grammatical elements, while UNL attributes often convey a meaning that is expressed both in English and in Russian by means of lexical items (e.g. modals).
- A NormSS contains information on the word order, while a UNL expression does not say anything to this effect.

These differences and similarities make the task of establishing a bridge between UNL and NormSS far from trivial but feasible.

The architecture of the UNL module within ETAP-3 is represented in Fig. 1.

As shown in Fig. 1, the interface between UNL and Russian is established at the level of the English NormSS. In the generation task, at this point, ETAP's English-to-Russian machine translation facility can be switched which carries through the phases of transfer and Russian generation. This architecture allows obtaining English generation for relatively cheap, as ETAP has a Russian-to-English mode of operation as well. Some experiments in this direction have been carried out which proved quite promising. Below, we will consider this scheme in the opposite direction – from the NL sentence to the UNL graph.

4 Interactive enconversion.

One of the most difficult problems in the automatic analysis of NL texts is the ambiguity of linguistic units. In ETAP-3, there is no single stage of processing expressly dedicated to disambiguation. The sentence is gradually disambiguated at different stages of processing on the basis of the restrictions imposed by the linguistic knowledge of the system. Examples:

1. lexical meanings with different grammatical properties: We have tea in the garden - We were having tea in the garden, but: I have a pack of tea - *I was having a pack of tea.
2. lexical meanings with different syntactic properties: Children grow fast – Children grow vegetables in the garden.
3. grammatical meanings with different syntactic properties: *represented* – past participle (*countries represented in the UN discuss the resolution*) vs. past indefinite (*he represented his country*)
4. different syntactic structures: *the accusation of the minister* – 'the minister accused somebody' vs. 'somebody accused the minister' (the type of ambiguity not sufficiently accounted for in UNL!). In the sentence *The accusation of the minister by the parliament*, syntactic context provides a clue for disambiguation.
5. different translations of the same lexical meaning: *Wash your hands* – Rus. *Vymoj ruki*, but *Wash the tablecloth* – Rus. *Postiraj skatert'.*

All these and many other cases are successfully disambiguated by ETAP-3 thanks to the linguistic knowledge it is supplied with. However, in many cases linguistic knowledge of the system is insufficient for disambiguation. Of course, this may be due to the incompleteness of grammar and the dictionaries of the system. Should it be the case, this obstacle could in principle be overcome. In the long run, the linguistic knowledge base could be made virtually complete. Unfortunately, however, incompleteness of linguistic data is not the main obstacle. It is well-known that in very many cases the disambiguation performed by humans is not based on their linguistic knowledge alone. To a large extent, humans heavily employ their extra-linguistic competence in the outer world

(1) *AIDS threatens economic collapse.*

It is very likely that the sentence will be wrongly understood as 'AIDS poses a threat to economic collapse' rather than 'AIDS threatens (some countries) with economic collapse', and, consequently, yield a wrong translation, for the simple reason that the system may lack the resources needed to distinguish the syntactic structure of this sentence from that of the sentence

(2) *AIDS threatens economic prosperity.*

Indeed, in order to make sure that the original sentence is parsed correctly, the system must know that the noun *collapse* instantiates the instrumental slot of the verb *to threaten* and not its object slot as in the second sentence. However, to provide adequate word lists for different slots of particular verbs is hardly possible because such lists will inevitably intersect in multiple ways; cf. ambiguous phrases like *threaten changes, threaten a revolution,* or *threaten the reduction.* On the other hand, any human who happens to read the BBC article will at once know what the original sentence (1) means.

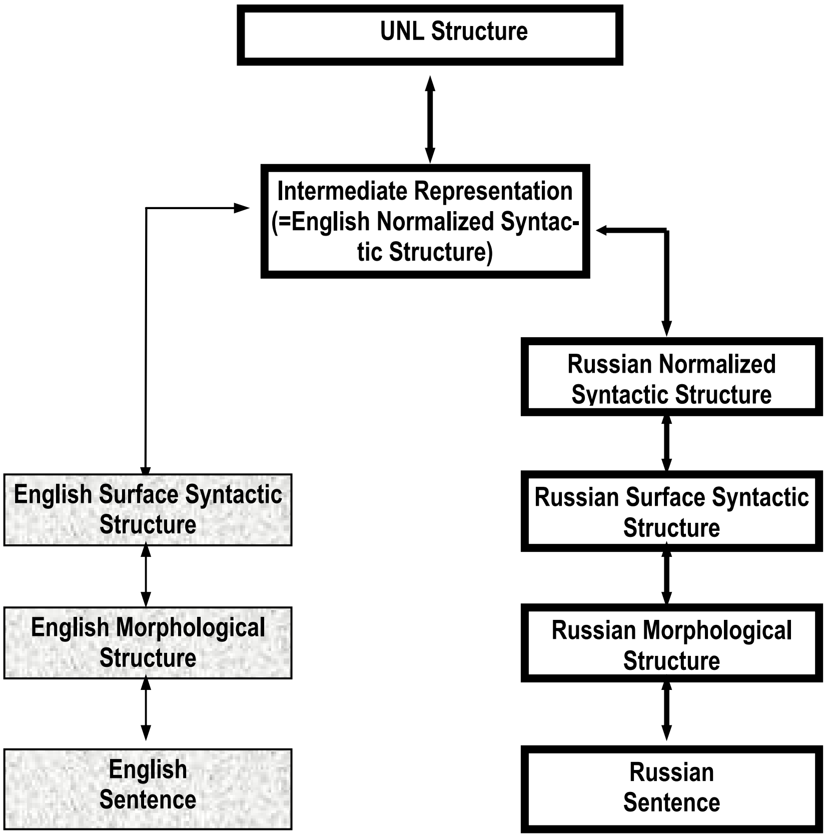

UNL Structure
Intermediate Representation (=English Normalized Syntactic Structure)
Russian Normalized Syntactic Structure
English Surface Syntactic Structure
Russian Surface Syntactic Structure
English Morphological Structure
Russian Morphological Structure
English Sentence
Russian Sentence

Fig. 1. To give a simple example, suppose that a machine translation system has to translate a title from a recent article on the BBC site.

It is therefore highly desirable that a rule-based NLP system be supplemented with an interactive tool that could, at certain pivotal points of language parsing, ask for human intervention and use this assistance to disambiguate some, or all of the ambiguous elements of the text being processed. Much work in this direction has already been done, first of all by the GETA group in the ARIANE environment [3–6, 10].

It is exactly this interactive tool that we present in this paper. It should be stressed that the interactive tool will only be activated for the cases of ambiguity that cannot be resolved automatically and therefore require human intervention.

We will illustrate our approach with one English example. The sentence

(3) We made the general remark that everything was fine

is ambiguous between (at least) two interpretations:

(3a) ‘we made the general observation that everything was fine’
(3b) ‘we made the general say that everything was fine’

Obviously, meanings (3a) and (3b) are translated differently into other languages and should receive two different UNL-representations.

As mentioned above, one of the salient features of ETAP-3 is the fact that it has a MULTIPLE TRANSLATION option that can produce multiple (ideally, all possible) translations of each sentence. This option allows obtaining two different lexico-syntactic structures of sentence (3) and consequently two different translations into UNL. These structures, given in Fig. 2 and 3 below, conveniently visualize lexical and syntactic differences between (3a) and (3b). Note that syntactic links are represented as labeled dependency relations between the words of the sentence. The lexico-syntactic structure of a sentence is a tree in which every word (except for the root node) is connected by an incoming dependency relation with some other word. The root has no incoming relations but only outgoing ones.

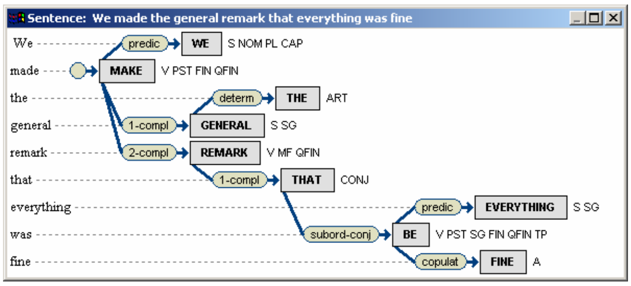


Fig. 2. Representation of the lexical and syntactic structure for the reading of (3a)

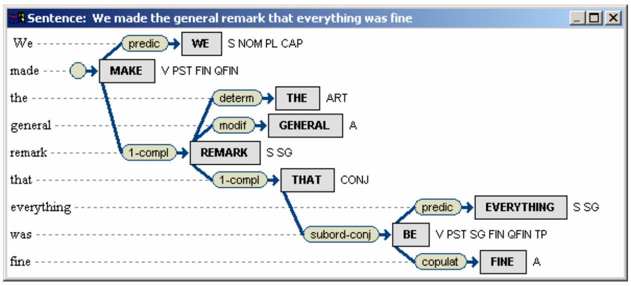
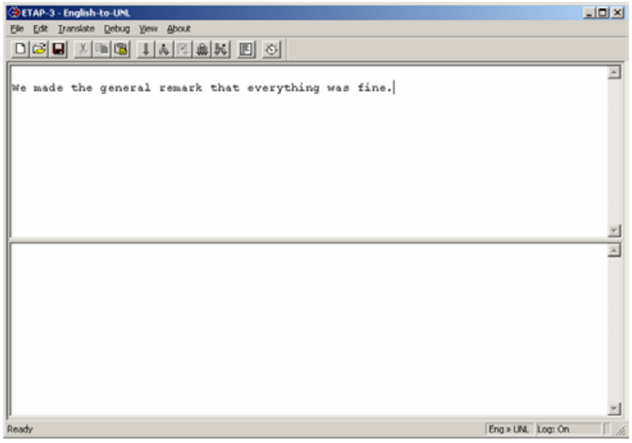


Fig. 3. Representation of the lexical and syntactic structure for the reading of (3a)

In Fig. 2 *general* is an adjective (cf. label A to the right of the gray rectangle with the name of the word) and *remark* is a noun (cf. label S). In Fig. 3, *general* is a noun (cf. label S) and *remark* is a verb (cf. label V). Accordingly, in Fig. 2 the adjective *general* serves as a modifier of the noun *remark* (cf. label *modif* on the link that connects *general* to *remark*) and article *the* is also attached to *remark*. In Fig. 3 the noun *general* attaches article *the* and serves as the first complement of the verb *make* while the verb *remark* is its second complement (cf. labels *1-compl* and *2-compl* on the corresponding links). Besides that, there is a purely lexical ambiguity not shown in these structures. The word *fine* is ambiguous between an adjectival meaning (as in *fine weather*), a nominal one (as in *to pay a fine*) and a verbal one (as in *You will be fined*).

ETAP-3 is able to identify these ambiguities but in a general case cannot automatically decide which of the options is appropriate in a particular context. As mentioned above, this task can be reliably solved only in co-operation with the human. Let us switch on the Interactive disambiguation mode of ETAP-3 and participate in the dialogue proposed by the system. Fig. 4 shows the initial state of the English-to-UNL option with the English sentence input in the upper window.

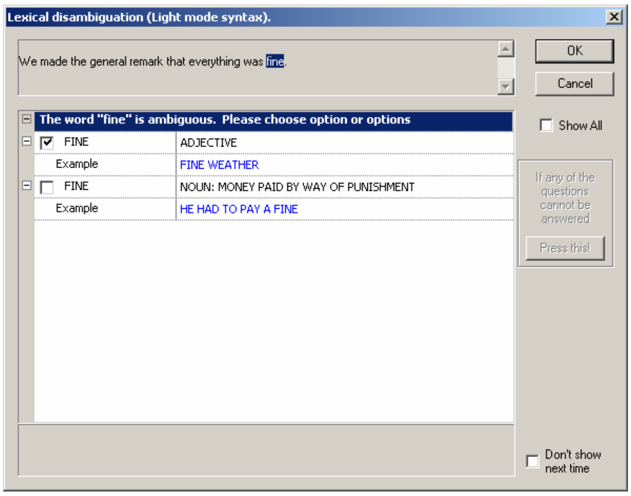


We made the general remark that everything was fine.

Fig. 4. Initial state of the English-to-UNL option with the English sentence input in the upper window

The first occasion for the system to ask a question is the moment in the parsing when the root node of the structure should be selected. If a word that a system has chosen as a root node is ambiguous and the system cannot resolve this ambiguity, the user is asked for assistance. In our example, the only candidate for the root node (*made*) is unambiguous and no need for human intervention arises.

The word that activates a dialogue on the lexical ambiguity is *fine*. Of the three options mentioned above, the verbal one is incompatible with the syntactic context, but the other two can perfectly fit into it. Therefore, the first option is automatically rejected by the system and the other two are offered to the user. Fig. 5 shows the dialogue window that appears when the user is asked for assistance in lexical disambiguation. In the upper part of the dialogue window the sentence is reproduced with the word at issue highlighted. Below, the user is shown the options not yet resolved by the system, among which he/she is asked to choose. Each option is provided with a short but clear and informative comment and/or a simple example. What the user should do is identify and click the appropriate option. Comments and examples are formulated in such a way that no special linguistic knowledge is required to choose among the options.


Lexical disambiguation (Light mode syntax).

We made the general remark that everything was fine.

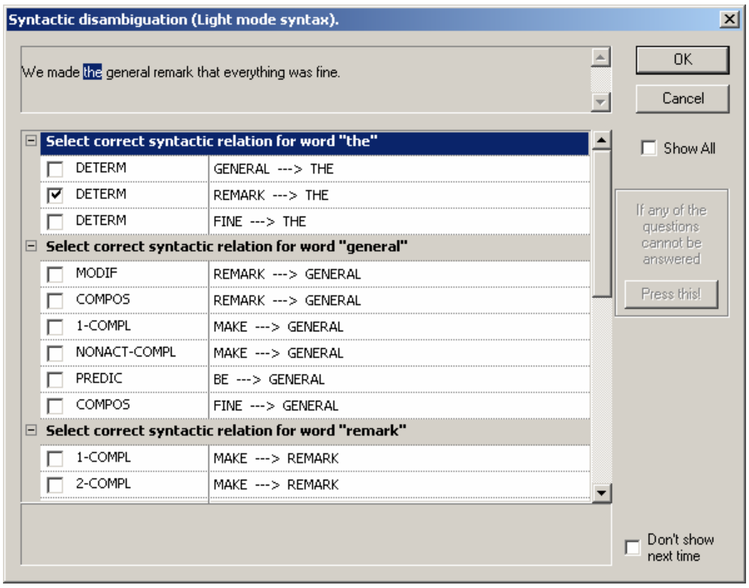
The word "fine" is ambiguous. Please choose option or options	
☑ FINE	ADJECTIVE
Example	FINE WEATHER
□ FINE	NOUN: MONEY PAID BY WAY OF PUNISHMENT
Example	HE HAD TO PAY A FINE

Fig. 5. Dialogue window for interactive lexical disambiguation

After having dealt with purely lexical ambiguity, the system passes to syntactic ambiguity or to complex cases when lexical and syntactic ambiguities come together. The syntactic ambiguity dialogue window represents words that have more than one alternative governors (while in a tree only one governor is allowed) and the parser has no means to make a choice. In Fig. 6, we see three situations of this type: article *the* can determine either *general* or *remark,* the word *general* can be subordinated by means of different syntactic relations either by *remark, make* or *general,* and *remark* can be linked either to *make* or to *be.* Note that *make* can subordinate *remark* by two different syntactic relations. The latter can be either the first complement of *remark* (as in *make the remark*), or the second complement (as in *make (the general) remark*). Obviously, some of these options rely on different part-of-speech characteristics of ambiguous words. For example, *remark* is a noun in *make the remark* and a verb in *make (the general) remark.*

For each word with alternative links, the user should choose one option and click the corresponding square. In Fig. 6 the phrase *the remark* is given priority over the phrase *the general.*

Often enough, we need not resolve all the ambiguities identified by the system. It may be the case that one choice made by the user is sufficient for the system to resolve the remaining ambiguities on its own. In our example, the resolution of any one of the ambiguities shown in Fig. 6 directly leads to automatic disambiguation of the remaining ones and to the construction of a UNL graph.

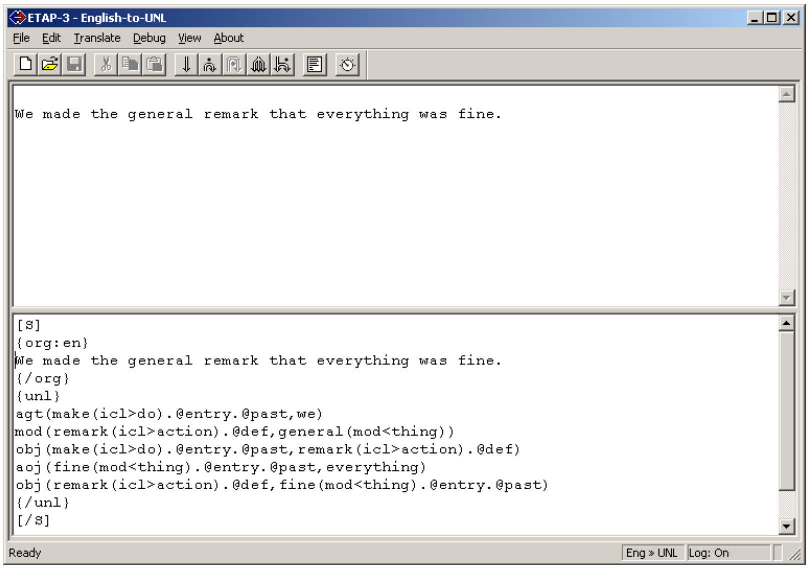

Syntactic disambiguation (Light mode syntax).
OK
Cancel
Show All
If any of the questions cannot be answered
Press this!
Don't show next time

We made the general remark that everything was fine.

Select correct syntactic relation for word "the"		
☐	DETERM	GENERAL ---> THE
☑	DETERM	REMARK ---> THE
☐	DETERM	FINE ---> THE
Select correct syntactic relation for word "general"		
☐	MODIF	REMARK ---> GENERAL
☐	COMPOS	REMARK ---> GENERAL
☐	1-COMPL	MAKE ---> GENERAL
☐	NONACT-COMPL	MAKE ---> GENERAL
☐	PREDIC	BE ---> GENERAL
☐	COMPOS	FINE ---> GENERAL
Select correct syntactic relation for word "remark"		
☐	1-COMPL	MAKE ---> REMARK
☐	2-COMPL	MAKE ---> REMARK

Fig. 6. Resolution of ambiguities

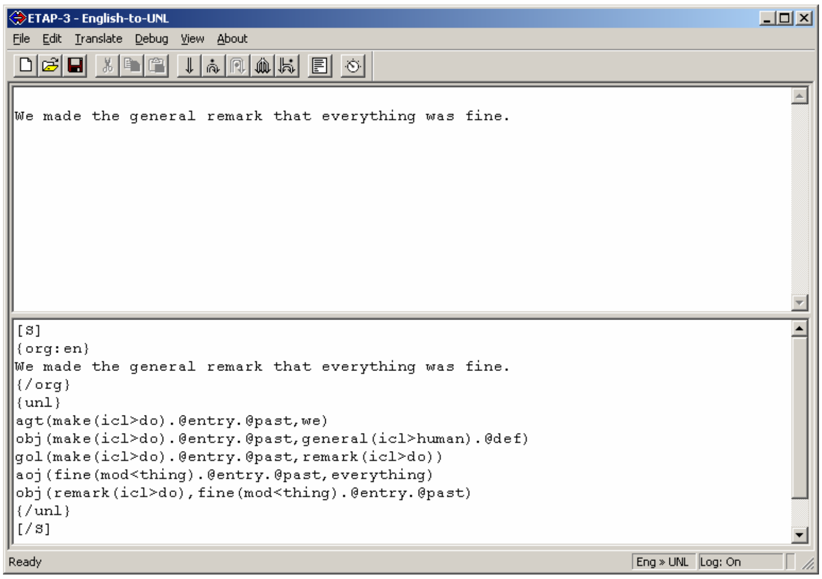
After the one choice made in Fig. 6, the enconvertor comes up with the UNL graph shown in Fig. 7. If, instead of selecting the phrase *the remark,* we had opted in Fig. 6 for the phrase *the general,* the result would have been different – see Fig. 8.



```
We made the general remark that everything was fine.
```

```
[S]
{org:en}
We made the general remark that everything was fine.
{/org}
{unl}
agt(make(icl>do).@entry.@past,we)
mod(remark(icl>action).@def,general(mod<thing))
obj(make(icl>do).@entry.@past,remark(icl>action).@def)
aoj(fine(mod<thing).@entry.@past,everything)
obj(remark(icl>action).@def,fine(mod<thing).@entry.@past)
{/unl}
[/S]
```

Fig. 7. UNL graph after interactive disambiguation



```
We made the general remark that everything was fine.
```

```
[S]
{org:en}
We made the general remark that everything was fine.
{/org}
{unl}
agt(make(icl>do).@entry.@past,we)
obj(make(icl>do).@entry.@past,general(icl>human).@def)
gol(make(icl>do).@entry.@past,remark(icl>do))
aoj(fine(mod<thing).@entry.@past,everything)
obj(remark(icl>do),fine(mod<thing).@entry.@past)
{/unl}
[/S]
```

Fig. 8. UNL graph for the second option in the process of interactive desambiguation

5 Future work

The interactive enconverter described above needs further improvement in the following directions.

First, the questions on the syntactic links should be supplied with clear and simple comments similar to the ones generated in the lexical ambiguity dialogue.

Second, the dialogue should be extended to the cases of UNL-related ambiguity. We mean here situations in which an unambiguous Russian or English word corresponds to more than one Universal Word.

Third, we are planning to supply a facility that allows to graphically visualize the output of the enconverter as a UNL graph and manually revise it by the human expert.

References

1. Apresjan Ju.D., I.M.Boguslavsky, L.L.Iomdin *et al.* Lingvisticheskij processor dlja slozhnyx informacionnyx sistem. (A linguistic processor for advanced information systems.) Nauka, 1992, 256 p. Moscow.
2. Apresjan Ju.D., I.M.Boguslavsky, L.L.Iomdin *et al*. ETAP-2: The Linguistics of a Machine Translation System. // META, Vol. 37, No 1, 1992, pp. 97-112.
3. Blanchon, H. *Interagir pour traduire : la TAO personnelle pour rédacteur monolingue.* in La Tribune des Industries de la Langues. Vol. 17-18-19, 1995, pp. 28-34.

4. Blanchon, H. *A Customizable Interactive Disambiguation Methodology and Two Implementations to Disambiguate French and English Input.* Proc. MIDDIM'96. Le col de porte, Isère, France. 12-14 Août 1996. Vol. 1/1, 1996, pp. 190-200.
5. Blanchon, H. Interactive Disambguation of Natural Language Input: a Methodology and Two Implementations for French and English. Proc. IJCAI-97. Nagoya, Japan. August 23-29, 1997. Vol. 2/2, 1997, pp. 1042-1047
6. Boitet, C. & Blanchon, H. Multilingual Dialogue-Based MT for monolingual authors: the LIDIA project and a first mockup. in Machine Translation. Vol. 9(2), 1995, pp 99-132.
7. Boguslavsky I. A bi-directional Russian-to-English machine translation system (ETAP-3). Proceedings of the Machine Translation Summit V. Luxembourg, 1995.
8. Boguslavsky I., N. Frid, L. Iomdin, L. Kreidlin, I. Sagalova, V. Sizov. Creating a Universal Networking Language Module within an Advanced NLP System // Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), 2000, p. 83-89.
9. Boguslavsky I. UNL from the linguistic point of view. // Proceedings of The First International Workshop on MultiMedia Annotation. Electrotechnical Laboratory SigMatics, Tokyo, 2001, 1-6.
10. Proceedings of the MIDDIM-96 seminar on interactive disambiguation. Le Col de Porte, 1996.

Prepositional Phrase Attachment and Interlingua

Rajat Kumar Mohanty, Ashish Francis Almeida, and Pushpak Bhattacharyya

Department of Computer Science and Engineering
Indian Institute of Technology Bombay
Mumbai - 400076, India
Emails: {rkm, ashishfa, pb}@cse.iitb.ac.in

Abstract. In this paper, we present our work on the classical problem of *prepositional phrase attachment.* This forms part of an interlingua based machine translation system, in which the semantics of the source language sentences is captured in the form of *Universal Networking Language (UNL)* expressions. We begin with a thorough linguistic analysis of six common prepositions in English, namely*, for, from, in, on, to* and *with.* The insights obtained are used to enrich a *lexicon* and a *rule base*, which guide the search for the correct attachment site for the prepositional phrase and the subsequent generation of accurate semantic relations. The system has been tested on British National Corpus, and the accuracy of the results establishes the effectiveness of our approach.

1 Introduction

No natural language processing system can do a meaningful job of analyzing the text, without resolving the prepositional phrase (PP) attachment. There are two fundamental questions related to this problem:

(1) *Given a sentence containing the frame*
[V-NP$_1$-P- NP$_2$]
does NP$_2$ attach to V or to NP$_1$?
(2) *What should be the semantic relation that*
links the PP with the rest of the concept graph of the sentence?

Our work is motivated by seeking answers to these questions. We focus our attention on six most common prepositions of English, *viz., for, from, in, on, to* and *with* (for the motivation, please see Table 5 in section 5).

In order to resolve these issues, we have taken linguistic insights from the following works [1–4]. Other related and motivating works specific to the PP-attachment problem are [5–9].

The roadmap of the paper is as follows: Section 2 provides a linguistic analysis of the six prepositions in question. The UNL system is introduced in Section 3. Section 4 discusses the design and implementation of the system. Evaluation results are given in Section 5. Section 6 concludes the paper and is followed by the references.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 241–253.

2 Linguistic Analysis

Prepositions are often termed as syntactic connecting words. However, they have syntactic as well as semantic specifications that are unique to them. The *selection* of a preposition is decided by the meaning of the syntactic elements that determine it, and the *meaning* depends partly on the preceding syntactic elements and partly on the ones that follow. We now provide a detailed linguistic study of six prepositions in English.

2.1 Syntactic Environments

A preposition can occur in different syntactic environments. For instance, the preposition *for* participates in eight different sequential environments. In each environment, it refers to a specific *thematic role*[1] depending on the semantics of the preceding and the immediately following lexical heads. Table 1 illustrates these environments.

Possible Frames	Examples
[NP–for-NP-V]	The search for the policy is going on.
[NP–for–V_{-ing}–NP-V]	The main channel for breaking the deadlock is the Airport Committee.
[V-for–NP]	He applied for a certificate.
[V-NP_1-for-NP_2]	He is reading this book for his exam.
[V-NP-for-V_{-ing}]	The Court jailed him for possessing a loaded gun.
[V-AP-for-NP]	She is famous for her painting.
[V-AP-for-V_{-ing}]	They are responsible for providing services in such fields.
[V_{-pass}-for-V_{-ing}]	They have been prosecuted for allowing underage children into the theatre.

Table 1: Syntactic environments of *for*

In this table, the first column gives the environments (henceforth, *frames*), and the second column gives the relevant examples. In fact, for each frame a preposition can have different senses depending on the *thematic role* of the NP which the preposition licenses[2] to.

The assumption is that thematic roles are closely related to the argument structure of particular lexical items (*viz*., verbs and complex event nominals). Each argument is assigned one and only one theta role. Each theta role is assigned one and only one argument. The relationship between the thematic properties of lexical items and their syntactic representations is mediated by a syntactic principle called the *theta-criterion* [1]. On the basis of the above assumption, Table 2 provides a brief analysis of six prepositions and the related verb types [4]. The first column provides the thematic

[1] In linguistic theory, *thematic roles* are broad classes of participants in events.

[2] By *licensing* we mean that in a PP the preposition governs and assigns *case* to the NP. (cf. *Governing Theory* and *Case Theory* [1])

roles. The rest of the columns show the *verb types* [4] that assign the thematic roles to the *P-NP*$_2$.

Thematic Roles	**For**	**from**	**In**	**On**	**To**	**With**
Benefactive	Build, Create, Prepare Verbs	–	–	–	–	–
Goal	Spend Verbs	–	Put Verbs	Put, Spend Verbs	Send Verbs	–
Instrumental	–	Build, Create, Prepare Verbs	–	–	–	Spray Verbs
Source	–	Send Verbs	–	–	–	–

Table 2: Thematic roles for [V-N$_1$–P-N$_2$] (*not exhaustive*)

2.2 Conditions for Attachment Sites

We focus our attention on the particular frame [V-NP$_1$–P-NP$_2$], for which the prepositional phrase attachment sites under various conditions are enumerated, as shown in Table 3. The descriptions are self explanatory.

Conditions	**Sub-conditions**	**Attachment Point**
[NP$_2$] is subcategorized by the verb [V]	[NP$_2$] is licensed by a preposition [P]	[NP$_2$] is attached to the verb [V] *(e.g., He forwarded the mail to John)*
[NP$_2$] is subcategorized by the noun in [NP$_1$]	[NP$_2$] is licensed by a preposition [P]	[NP$_2$] is attached to the noun in [NP$_1$] *(e.g. She had no answer to the accusations)*
[NP$_2$] is neither subcategorized by the verb [V] nor by the noun in [NP$_1$]	[NP$_2$] refers to [PLACE] *feature*	[NP$_2$] is attached to the verb [V] *(e.g., I met him in his office; The girls met him on different days)*
	[NP$_2$] refers to [TIME] *feature*	

Table 3: PP-attachment conditions for the *frame* [V-NP$_1$-P-NP$_2$]

3 The UNL System

UNL is an electronic language for computers to express and exchange information [10]. UNL consists of *Universal words (UW)*, *relations*, *attributes*, and the *UNL knowledge base (KB)*. The UWs constitute the vocabulary of UNL, relations and attributes the syntax and the UNL KB the semantics of the framework. UNL represents information sentence by sentence as a hyper-graph with concepts as nodes and relations as arcs. Figure 1 represents the UNL graph for the sentence (4).

```
(4) The boy went to school.
```

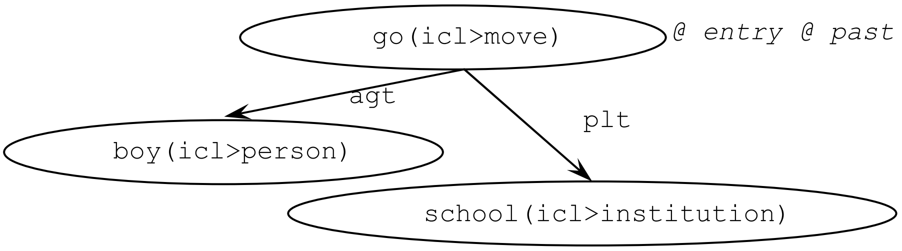


Figure 1: UNL graph for the sentence ‘*The boy went to school*’.

In *figure 1*, the arcs labeled with *agt* (agent) and *plt* (destination) are the relation labels. The nodes *go(icl>move), boy(icl>person), school(icl>institution)* are the *Universal Words* (*UW*). These are words with *restrictions* in parentheses for denoting unique sense. UWs can be annotated with attributes like *number*, *tense etc.*, which provide further information about how the concept is being used in the specific sentence. Any of the three restriction labels- *icl(inclusion of)*, *iof (instance of)* and *equ* (*used for abbreviations*)- can be attached to an UW for restricting its sense. For (4), the UNL expressions are as follows:

```
(5)   agt(go(icl>move).@entry.@past, boy(icl>person))
      plt(go(icl>move).@entry.@past, school(icl>institution))
```

The most recent specfication of the UNL contains 41 relation labels and 67 attribute labels [11].

3.1 The Analyzer Machine

The analysis of the source language sentences into UNL is carried out using a language independent analyzer called *EnConverter* [12], which does morphological, syntactic and semantic analysis sentence by sentence, accessing a knowledge rich **Lexicon** and interpreting the **Analysis Rules**. The *EnConverter (henceforth, EnCo)* is essentially a multi headed Turing Machine which has two kinds of heads: *processing heads* and *context heads*. The processing heads are also called *Analysis Windows* and are *two* in number: the *left analysis window (LAW)* and the *right analysis window (RAW).* The context heads are also called *condition windows* of which there can be many.

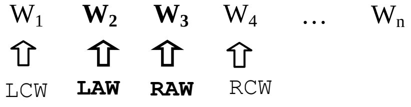


Figure 2. EnCo analyses a sentence by placing *windows* on the constituent words.

The nodes under the analysis windows (Figure 2) are processed for linking by a UNL relation label and/or for attaching UNL attributes to. The contents of a node are the Head Words (HWs), the Universal Words (UWs), and the lexical and the UNL attributes. The context heads are located on either side of the processing heads and are used for look-ahead and look-back. The machine has functions like *shifting the windows right or left by one node*, *adding a node to the node-list* (tape of the machine), *deleting a node*, *exchange of nodes under processing heads*, *copying a node* and *changing the attributes of the nodes*. During the analysis, whenever a UNL relation is produced between two nodes, one of these nodes is deleted from the tape and is added as a child of another node in the tree. Forming the analysis rules for EnCo is equivalent to programming a sophisticated symbol processing machine.

3.2 The English Analyzer

The English Analyzer makes use of the *EnCo*, the *English-UW dictionary* and the *rule base* for English Analysis. At every step of the analysis, the rule base drives the EnCo to perform tasks like

a. completing the morphological analysis (*e.g.*, combine *Boy and 's*),
b. combining two grammatical entities (*e.g., is* and *working*) and
c. generating a UNL relation (*e.g., agt* relation between *he* and *is working).*

Many rules are formed using Context Free (CFG)-like grammar segments, the productions of which help in clause delimitation, prepositional phrase attachment, part of speech (POS) disambiguation and so on. This is illustrated with the example of noun clause handling:

```
(6) The boy who works here went to school.
```

The processing proceeds as follows:

a. The clause *who works here* starts with a relative pronoun and its end is decided by the system using the grammar. The system does not include *went* in the subordinate clause, since there is no rule like
   *CL*AUSE-> WH-Word V ADV V

b. The system detects *here* as an adverb of place from the lexical attributes and generates *plc* (place relation) with the verb *work* of the subordinate clause. At this point *here* is deleted. After that, *work* is related with *boy* (which is modified by the relative clause and coindexed with the relative pronoun *who*) through the *agt* relation and gets deleted. At this point the analysis of the clause finishes.

c. *boy* is now linked with the main verb *went* of the main clause. Here too the *agt* relation is generated after deleting *boy*.

d. The main verb is then related with the prepositional phrase to generate *plt* (indicating *destination),* taking into consideration the preposition *to* and the noun *school* (which has *PLACE* as a semantic attribute in the lexicon). *to* and *school* again are deleted. From *went, go(icl>move)* is generated with the

@entry attribute- which indicates the main predicate of the sentence- and the analysis process ends.

The final set of UNL expressions for the sentence in (6) is given in (7)[3].

```
(7) agt(go(icl>move).@entry.@past, boy(icl>person))
    plt(go(icl>move).@entry.@past, school(icl>institution))
    agt(work(icl>do),boy(icl>person))
    plc(work(icl>do),here))
```

The English analysis system currently has close to 5000 analysis rules and approximately 70,000 entries in the lexicon.

4 Design and Implementation of the PP-Attachment System

The system is implemented using an enriched lexicon and a rule base that guide the operation of the English analyzer (*cf.* Section 3.2). We first describe the enrichment of the *lexicon*. This is followed by the *core strategy* of analysis, which is heavily lexicon dependent. The strategy is translated into the *rule base.*

4.1 The Lexicon

The lexicon is the heart of the UNL system. Lexical knowledge consists of lines of entries describing the *headword (HW)*, the *Universal Word (UW)* and the *properties* of *HW*. For example, the lexical entries for (8a) are given in (8b):

```
(8)a. John ate rice with a spoon
   b. [John] "John(iof>person)" (N,MALE,PROPER,ANIMATE)
       [eat] "eat(icl>do)" (V,VoI)
       [rice] "rice(icl>food)" (N,FOOD)
      [spoon] "spoon(icl>artifact)" (N,INSTR)
```

The HWs are enclosed in square brackets, the UWs in quotes and the properties of the *HWs* in parentheses. The properties are fairly obvious except possibly for *VoI* which means *verb of ingestion* and *INSTR* which means *instrument.*

As discussed in Section 2, the arguments of *V* and *N* are lexically specified. For example, consider the entry for *give* in the lexicon:

```
(9)[gave] "give(icl>do)"(VRB,VOA,VOA-PHSL,PAST)<E,0,0>;
```

The attributes are shown within parentheses. These attributes specify that *give* is a *verb (VRB), verb of action (VOA), physical-action verb (VOA-PHSL),* and is in *past* tense *(PAST).* Now, consider the sentence

```
(10) He gave a gift to her
```

in which *give* takes one NP as its first argument and a PP as its second argument. This is specified in the lexicon through the attribute #_TO_A2. Additionally, the UNL relation is specified (*#_TO_A2_GOL*). This leads to

[3] The adverb *here* does not need a disambiguating restriction.

```
(11)[gave] "give(icl>do)" (VRB,VOA,VOA-PHSL,#_TO_A2,
                          #_TO_A2_GOL,PAST) <E,0,0>;
```

The entries for nouns and adjectives are enriched in a similar manner.

4.2 Strategy of Analysis: Exploiting the Lexical Attributes

To determine the attachment site of NP_2, four cases of different attribute combinations are considered, as shown in Table 4. *#<P>* indicates that preposition *P* is part of the attribute list of *V* or N_1 and *Not#<P>* suggests the absence from the attribute list.

	Conditions in lexicon			**Action**	
	Attributes of *V*	**Attributes of NP_1**	**Attributes of NP_2**	**Attachment of NP_2**	**Examples**
1	#<P>	#<P>	–	N_1	…paid a visit to the museum. …imposed a law on food hygiene.
2	#<P>	Not#<P>	–	V	...passed the ball to Bill. …imposed heavy penalties on fuel dealers.
3	Not#<P>	Not#<P>	–	N_1	…saw the trap in question.
			#<PLACE>	V	…met him in his office.
			#<TIME>		…met him in the afternoon.
4	Not#<P>	#<P>		N_1	…supplied plans for projects.

Table 4: Lexical conditions for *P-NP_2 -attachment*

The explanation of Table 4 is as follows:

A. NP_2 is attached to *V*, only when
(*V* has *#P* attribute) **AND** (N_1 does not have it); see *row 2*,

Otherwise

B. NP_2 is attached to N_1 when
(both *V* and N_1 have *#<P>* attribute); see *row 1*

OR

(*V* does not have #<*P*>) **AND** (N_1 has it); see *row 4*

Otherwise

C. (Neither *V* nor N_1 has *#<P>*, in which case combinations of attributes of *V*, N_1 *or* N_2 determine the attachment site); see *row 3*

The strategy enumerated produces UNL relations corresponding to the six prepositions under consideration. These relations and the various attributes that are called into play appear in Appendix A.

4.3 The Rule Base

The strategy illustrated through Table 4 is converted into a set of rules which guides the analysis process. There are two types of rules, specific to PP-attachment:

Type I: Rules *using the argument structure information* provided in the lexicon.

Type II: Rules *identifying the noun with spatial/temporal feature* and attaching it to the verb or to the nearest complex event nominal.

Let us consider an example of a Type I rule. The rule r_1 in (12) decides when to shift right to take care of case 1 in Table 4.

```
(12) ;Right shift to affect noun attachment
     r1. R{VRB,#_FOR_AR2:::}{N,#_FOR:::}(PRE,#FOR)P60;
```

This states that

IF

the left analysis window is on a *verb* which takes a *for-pp*
as the *second* argument (indicated by *#_FOR_AR2*)

AND

the right analysis window is on a *noun* which takes a *for-pp*
as an argument (indicated by `#_FOR`)

AND

the preposition *for* follows the *noun* (indicated by `(PRE,#FOR)`)

THEN

Shift right (indicated by *R* at the start of the rule)
(anticipating *noun attachment for the pp*).

The priority of this rule is 60 which should be between 0 (lowest) and 255 (highest). The priority is used in case of *rule conflict*.

Taking another example, where a UNL relation is created, the rule r_2 in (13) sets up *rsn* (standing for *reason)* relation between *V* and NP_2 and deletes the node corresponding to NP_2

```
(13)  ; Create relation between V and N2, after resolving the
        preposition preceding N2
     r2. <{VRB,#_FOR_AR2,#_FOR_AR2_rsn:::}{N,FORRES,PRERES::rsn:}P25;
```

This states that

IF

the left analysis window is on a *verb* which takes a *for-pp*
as the *second* argument which should be linked with the
rsn relation (indicated by `#_FOR_AR2_rsn`)

AND

the right analysis window is on a *noun* for which the preceding preposition has been processed and deleted

THEN

set up the *rsn* relation between *V* and N_2.

The above is relation-setting rule as indicated by < at the start of the rule. The priority is 25.

Now we consider an example of Type II rules (r_3 and r_4), where *tim* relation is set up between *V* and N_2 with the help of the attributes of N_2.

```
(14) r3.DL(VRB,EVENT,VOA){PRE,#ON:::}
                              {N,UNIT,TIME,DAY:+ONRES,+PRERES::}P27;
     r4.<{VRB,EVENT,VOA:::}{N,TIME,UNIT,ONRES,PRERES::tim:}P20;
```

The rules are added to the existing rule base so that they can work in conjunction with the basic rules of the analyzer machine (*shifting*, *relation-setting*, *node-deleting*, *node-inserting*, *attribute-changing* and so on and so forth). The new rules use the new set of attributes to resolve the PP.

5 Evaluation

In this section, the preparation of the test data and the experiments conducted thereon, are reported.

5.1 Creation of Test Data

For the linguistic analysis, we relied on the data from Oxford genie [13], Web Concordancer [14], Wordnet 2.0 [15], and [16]. The obvious reason is the availability of a number of sentence structures with a variety of semantic information. The relevant sentences were collected, and segmented into sequential frames, each frame containing a preposition.

5.2 Experiments and Top Level Statistics

The experiment of generating UNL expressions has been performed on the British National Corpus [15]. We have chosen the BNC corpus mainly because of its wide domain coverage. The only hindrance to using it is that the sentences are too long to be easily processed. Hence a word limit of 12-15 words per sentence was imposed on the test sentences. The steps in the evaluation are as follows:

a. Sentences with various patterns are extracted. Care is taken to exclude frames with phrasal verbs and compound nouns (which are not in the scope of the current work).
b. These are processed by the EnCo to generate UNL expressions.
c. The correctness of the UNL expressions is manually ascertained. A correct UNL *entails that attachment problems have been already solved.*

One sentence from each *sentence type* for six prepositions was tested (*cf. Table 5* and *6*). The result shows 100% accuracy. The UNL expressions for six representative sentences for the six prepositions under study are given in Appendix B.

For	**From**	**In**	**on**	**To**	**With**
0.7 million	0.35 million	1.4 million	0.5 million	0.8 million	0.6 million

Table 5: statistics of the participation of six prepositions in BNC; these six account for about 45% of the total 11 millions PPs in the corpus.

Prepositions in the frame [V-NP_1-P-NP_2]	**Total no. of Sentence Types**	**Examples**
For	6	He carved a toy for the baby. The Court jailed him for 8 years. He is the Commissioner for Inland Revenue. He is reading this book for his exam. They selected him for his honesty. This is the train for Delhi.
From	3	This is a proposal from a group. They make a small income from fishing. They are starting their project from next Sunday.
In	8	I have confidence in him. I deposited the money in my bank account. He revealed this fact in a short statement. He delivered his speech in English. He lost his arm in an accident. I met him in his office. I meet him in the evening. The council recorded 12 complaints in two weeks.
On	5	I put the book on the table. He commissioned John on personal basis. I can picture a farmer on a picnic. I met him on the road. The girls met him on different days.
To	4	They served a wonderful meal to fifty delegates. He forwarded the mail to the minister. We received an invitation to the wedding. Ambulances rushed the injured to the hospital.
With	8	He cancelled a meeting with his students. She wore a green skirt with a blouse. They equated the railways with progress. He covered the baby with a blanket. He started the event with a hectic schedule. I bother her with my problems. That provides him with a living. He is playing chess with his friend.

Table 6: Statistics of sentence types for six prepositions in the frame [V-NP_1-P-NP_2]

We obtained correct UNL relations for all the sentence types (Table 6 above) involving the six prepositions under study.

6 Conclusion and Future Work

In this paper we have investigated the problem of PP-attachment in the context of interlingua based MT systems. Our work reinforces the belief that an in-depth linguistic analysis of sentence phenomena not only leads to the design of accurate systems, but also makes the task of evaluation simpler, in that only a set of *sentence types* need to be tested and not millions of sentences. The investigation also underlines the importance of designing rich and high-quality lexicons and integrating these with comprehensive rules of analysis. The future work consists in extending the approach to the complete set of English prepositions and the post positions for Indian languages.

References

1. Chomsky, Noam.: *Lectures on Government and Binding*. Foris, Dordrecht. (1981)
2. Grimshaw, Jane.: *Argument Structure*. The MIT Press, Cambridge, Mass. (1990)
3. Jackendoff, Ray.: *Semantic Structures*. The MIT Press, Cambridge.(1990)
4. Levin, Beth.: *English verb Classes and Alternation*. The University of Chicago Press, Chicago. (1993)
5. Brill, E. and Resnik, R.: A Rule based approach to Prepositional Phrase Attachment disambiguation. *Proc. of the fifteenth International conference on computational linguistics*. Kyoto. (1994)
6. Kordoni, Valia.: A Robust Deep Analysis of Indirect Prepositional Arguments. *Proc. of ACL-SIGSEM workshop on preposition* 2003, Toulouse.
7. Hindle, D and Rooth, M.: Structural Ambiguity and Lexical Relations. *Computational Linguistics,* 19(1). (1993)
8. Niemann, Michael.: Determining PP attachment through Semantic Associations and Preferences. Ms. (2003)
9. Ratnaparkhi, Adwait.: Statistical Models for Unsupervised Prepositional Phrase Attachment. Proc. of COLING-ACL 1998. http://www.cis.upenn.edu/~adwait /statnlp.html
10. Uchida, Hiroshi., Zhu, M., and Senta, T. Della.: UNL: A Gift for a Millennium.The United Nations University, Tokyo. (1999) http://www.undl.org/publications/gm/top.htm
11. UNDL Foundation: The Universal Networking Language (UNL) specifications version 3.2. (2003) http://www.unlc.undl.org
12. UNU/ IAS. 2003. EnConverter Specification Version 3.3. UNL Center, United Nations University/ Institute of Advanced Studies, Tokyo.
13. Hornby, A. S.: *Oxford Advanced Learner's Dictionary of Current English*. Oxford University Press, Oxford.(2000)
14. Greaves, Chris.: Web Concordancer. http://www.edict.com.hk
15. Miller, George. Wordnet 2.0. (2003) http://wordnet.princeton.edu/
16. BNC Consortium: *British National Corpus*. The Humanities Computing Unit of Oxford University. (2000) http://www.hcu.ox.ac.uk/BNC.

Appendix A

Prepositions and their UNL Relations

By applying the strategy specified in the previous section, we have generated the UNL relations for the six prepositions under consideration. Lexical attributes have been used in most cases. These facts are presented in the following Table.

	UNL Relations	**Attributes of [V]**	**Attributes of [N_1]**	**Attributes of [N_2]**
For	*ben*	[#_FOR_A2_ben]	-	[N,ANIMATE]
	dur	[VRB]	-	[TIME,UNIT, PL]
	mod	[BE]	-	[^
	pur	[#_FOR_A2_rsn]	-	-
	rsn	[#_FOR_A2_rsn]	-	[ABS]
	to	[BE]	-	[PLACE]
From	*frm*	[BE]or [HAVE]	[N]	[N]
	src	[#_FROM_A2_src]	-	-
	tmf	[VRB]	-	[TIME,UNIT]
In	*aoj*	[HAVE]	[ABS]	[ANIMATE]
	gol	[#_IN_A2_gol]	-	-
	man	[VOA-COMM]	-	[PSYFTR,ABS]
	met	[VOA-COMM]	-	[PSYFTR,ABS, LANG]
	scn	[VRB]	-	[EVENT,ABS]
	plc	[VRB]	[EVENT]	[PLACE]
	tim	[VRB]	[EVENT]	[TIME]
	dur	[VRB]	[EVENT]	[TIME,UNIT,PL]
On	*gol*	[#_IN_A2_gol]	-	-
	man	[VRB]	-	[PSYFTR,ABS]
	scn	[VRB]	-	[ABS]
	plc	[VRB]	[EVENT]	[PLACE]
	tim	[VRB]	[EVENT]	[TIME]
To	*ben*	[#_TO_A2_ben]	-	
	gol	[#_TO_A2_gol]	-	[PLACE]
	obj	[V,^VOA-MOTN]	-	[EVENT,ABS]
	plt	[V,VOA-MOTN, TO_plt]	-	[PLACE]
With	*cag*	[VOA]	-	[ANIMATE]
	cao	[BE]		[ABS]
	cob	[#_WITH_A2_cao]	-	-
	ins	[#_WITH_A2_ins]	-	[^ABS]
	man	[VOA]	-	[PSYFTR,ABS]
	met	[#_WITH_A2_met]	-	[ABS]
	obj	[#_WITH_A1_obj]	[ANIMT]	-
	Ptn	[#_WITH_A2_ptn]		[ANIMT]

Table 7: UNL Relation Inventory for Six Prepositions in the frame [V-NP_1-P_1-NP_2]

Appendix B

Out of 34 tested sentences, six sentences with UNL expressions are given in the following Table.

	Sentences with UNL Expressions

For	He carved a toy for the baby.
	{unl}
	ben(carve(icl>cut):03.@entry.@past,baby(icl>child):0O.@def)
	obj(carve(icl>cut):03.@entry.@past,
	toy(icl>plaything):0C.@indef)
	agt(carve(icl>cut):03.@entry.@past, he:00)
	{/unl}

From	They make a small income from fishing.
	{unl}
	src(make(icl>do):05.@entry.@present,fishing(icl>business):0U)
	obj(make(icl>do):05.@entry.@present,income(icl>gain):0I.@indef)
	agt(make(icl>do):05.@entry.@present, they(icl>persons):00)
	mod(income(icl>gain):0I.@indef,small(aoj>thing):0C)
	{/unl}

In	I deposited my money in my bank account.
	{unl}
	gol(deposit(icl>put):02.@entry.@past,account(icl>statement):0W)
	obj(deposit(icl>put):02.@entry.@past,money(icl>currency):0F)
	agt(deposit(icl>fasten):02.@entry.@past, I:0C)
	mod(money(icl>currency):0F, I:0C)
	mod(account(icl> statement):0W,bank(icl>possession):0R)
	mod(account(icl> statement):0W, I:0O)
	{/unl}

On	I put the book on the table.
	{unl}
	gol(put(icl>move):02.@present.@entry,table(icl>object):0M.@def)
	obj(put(icl>move):02.@present.@entry,
	book(pof>publication):0A.@def)
	agt(put(icl>move):02.@present.@entry,I:00)
	{/unl}

To	They served a wonderful meal to fifty delegates.
	{unl}
	gol(serve(icl>provide):05.@entry.@past,
	delegate(icl>person):12.@pl)
	obj(serve(icl>provide):05.@entry.@past,
	meal(icl>food):0O.@indef)
	agt(serve(icl>provide):05.@entry.@past, they(icl>thing):00)
	mod(meal(icl>food):0O.@indef, wonderful(mod<thing):0E)
	qua(delegate(icl>person):12.@pl, fifty(icl>number):0W)
	{/unl}

With	John covered the baby with a blanket.
	{unl}
	ins(cover(icl>do):05.@entry.@past,
	blanket(icl>object):0T.@indef)
	obj(cover(icl>do):05.@entry.@past,baby(icl>child):0H.@def)
	agt(cover(icl>do):05.@entry.@past,john(iof>person):00)
	{/unl}

Table 8: UNL Expressions for six representative sentences for the six prepositions under study

Hermeto: A NL-UNL Enconverting Environment

Ronaldo Martins, Ricardo Hasegawa and M. Graças V. Nunes

Núcleo Interinstitucional de Lingüística Computacional (NILC)
Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação
Av. do Trabalhador São-Carlense, 400. CEP 13560-970. São Carlos – SP – Brasil
ronaldo@nilc.icmc.usp.br; gracan@icmc.usp.br; rh@nilcicmc.usp.br
http://www.nilc.icmc.usp.br

Abstract. This paper aims at presenting and describing HERMETO, a computational environment for fully-automatic, both syntactic and semantic, natural language analysis. HERMETO converts a list structure into a network structure, and can be used to enconvert from any natural language into the Universal Networking Language (UNL). As a language-independent platform, HERMETO should be parameterized for each language, in a way very close to the one required by the UNL Center's EnConverter. However, HERMETO brings together three special distinctive features: 1) it takes rather high-level syntactic and semantic grammars; 2) its dictionaries support attribute-value pair assignments; and 3) its user-friendly interface comprises debug, compiling and editing facilities. In this sense, HERMETO is said to provide a better environment for the automatic production of UNL expressions.

1 Introduction

In the UNL System [1], natural language (automatic) analysis has been carried out either by the EnConverter (EnCo) [2] or, more recently, by the Universal Parser (UP) [3], both provided by the UNL Center. In the first case, enconverting from natural language (NL) to Universal Networking Language (UNL) is supposed to be conducted in a fully-automatic way, whereas in the second case a full-fledged human tagging of the input text should be carried out before NL analysis is triggered. In both cases, results have not been adequate. EnCo's grammar formalism, as well as UP's tagging needs, are rather low-level, and requires a human expertise seldom available. In what follows, we present an alternative analysis system, HERMETO, developed at the Interinstitutional Center for Computational Linguistics (NILC), in Sao Carlos, Brazil, which has been used for automatic enconverting from English and Brazilian Portuguese into UNL. Due to its interface debugging and editing facilities, along with its high-level syntactic and semantic grammar and its dictionary structure, it is claimed that HERMETO may provide a more user-friendly environment for the production of UNL expressions than EnCo and UP.

The structure of this paper is as follows. The second section, on motivation, addresses the context in which the HERMETO initiative was conceived and the goals ascribed to the system. The third section presents HERMETO's architecture. HERMETO's functioning is briefly detailed in section four (on resources) and five

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 254–260.

(on processes). Partial results, though rather preliminary, are reported in section six. Limitations and further work are addressed in section seven.

2 Motivation and Goals

HERMETO is a side product of two ongoing research and development projects carried out by NILC: POLICARPO and PUL∅. The former concerns the development of an English-to-Portuguese web translator, specialized in translating headlines and leads from the electronic edition of *The New York Times on the Web* into Brazilian Portuguese. PUL∅ concerns the development of a bimodal human-aided machine translation system for translating a Brazilian comics into LIST, a linearized version of Libras, the Brazilian Sign Language (for deaf people). Both systems are conceived as exclusively language-based, in the sense they are not supposed to require any extra-linguistic knowledge (as the one required in KBMT systems [4]) neither a corpus of already translated samples (as in the case for EBMT systems [5]). Additionally, both POLICARPO and PUL∅ were originally conceived as interlingua-based multilingual MT systems. Although the transfer approach might seem more suitable for each isolated task, our final goal is to provide a single system able to process, bidirectionally, both the oral-auditive (English and Portuguese) and the sign-gestural (LIST) input and output.

UNL was chosen as the pivot language because of three main reasons: 1) it's an electronic language for representing the semantic structure of utterances rather than its syntactic form; 2) the repertoire of UNL attributes can be extended to comprise semantic visual markers (as '.@round', '.@square', etc) required by sign language processing; and 3) as a multilingual and multilateral project, UNL could be used to assign cross-cultural interpretability to Portuguese and LIST texts. Nevertheless, it should be stressed that the use of UNL as an interlingua does not imply that UNL can only be used in such a way. This was a project strategy rather than a UNL vocation or shortcoming.

In such a multilingual MT environment, HERMETO was conceived as an embedded NL analysis system, which should allow for developer's customization and language parameterization. In its current state, it takes any plain text and enconverts it into UNL by means of a bilingual NL-UNL dictionary and a syntactic-semantic context-free grammar, both defined and provided by the user. The system was developed in C++, but it is still bound to the Windows environment. HERMETO's architecture is presented in the next section.

3 Architecture

HERMETO's architecture is presented in Figure 1 below. The input text - a plain text (.txt) written in ASCII characters - is split into sentences, each of which is tokenized and tagged according to the dictionary entries. Next, each sentence is traversed by a top-down left-to-right recursive parser, which searches for the best candidate matching as defined in the context-free grammar provided by the user. After parsing, the resulting syntactic structure is interpreted into UNL according to the projection rules written in the user's semantic grammar. The output is a UNL document, in its table form, i.e., as a list of binary relations embedded between UNL tags.

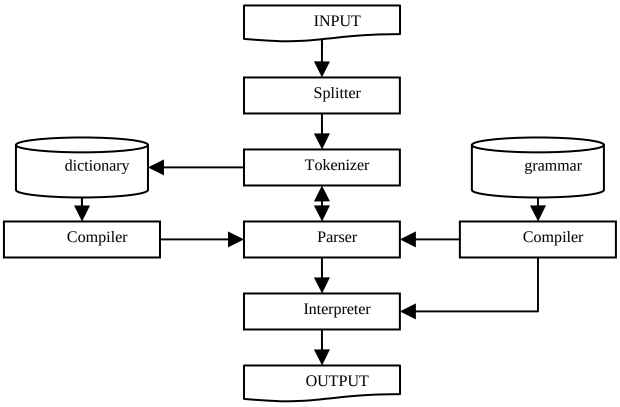


Fig. 1 HERMETO's architecture

4 Resources

HERMETO’s lingware consists of a bilingual NL-UNL dictionary and a NL-UNL transfer grammar. No other language resource (as the UNL KB, for instance) is required for the time being. Both dictionary and grammars are plain text files, which are automatically compiled by the very machine. In order to improve grammar-writing tasks, HERMETO also comprises a grammar editor.

4.1 Dictionary

As EnCo, HERMETO takes a NL-UNL dictionary, whose entries, one per line, must be presented in the following format:

[NLE] {id} NLL "UW" (FEATURE LIST) <LG,F,P>;

NLE stands for "NL entry", which can be a word, a subword or a multiword expression, depending on the user’s choice. NLL stands for "NL lemma". It is an optional field that can be used to clarify the string intended as NLE. The feature list consists of a list of attribute-value pairs, separated by comma. LG stands for a two-character language flag, according to the ISO 639. F and P indicate frequency and priority and are used for analysis and generation, respectively. Finally, any entry can be glossed and exemplified after the semi-colon.

The structure of HERMETO's dictionary is very much the same as EnCo's one: both dictionaries do not state any predefined structure, except for the syntax of each entry, and they can be customized by the user, who is supposed to decide the form of the entry, the need for lemmas and the set of attributes and the values they can take. However, there are three differences that should be stressed: 1) HERMETO compiles the plain text file itself, i.e., there is no need for a tool as DicBuild; 2) in HERMETO, the feature list is not a mere list of features but a list of attribute-value pairs, which allow for introducing variables in the grammar rules; and 3) HERMETO not only indexes but also compresses the dictionary (at the average rate of 65%).

Examples of dictionary entries are presented below:

[mesa] {} mesa “table(icl>furniture)” (pos:nou, gen:fem) <PT,1,1>;
[table] {} table “table(icl>furniture)” (pos:nou) <EN,1,1>;
[mesa] {} mesa “table(icl>furniture)” (pos:nou, ref:phy, fmt:squ) <LI[1],1,1>;

Except for the structure of the feature list and the language flag, HERMETO’s dictionary formalism is the same as the one proposed in the EnCo’s environment.

4.2 Grammar

HERMETO's grammar is a phrase-structure grammar defined by the 6-uple <N,T,P,I,W,S>, where N stands for the set of non-terminal symbols; T is the set of terminal symbols; P is the set of production rules; I is the set of interpretation rules; W is the weight (priority) of rules; and S stands for the start symbol. It is a context-free grammar, written in a plain text file, to be automatically compiled by the machine. The set of terminal symbols to be used as variables should be defined in the top of the grammar file, and the mapping between this set and the dictionary attribute values should be stated at the end of the document.

The rules should follow the formalism: p –> i, where $p \in P$, and $i \in I$. P, which is the syntactic component, can be expanded as a[w] := b, where $a \in N$, $b \in N \cup T$, and $w \in W$. I, the semantic component, is expanded as a list of attributes and relations in the following format: **att_1, att_2, ..., att_n, rel_1, rel_2, ..., rel_n** where att stands for attributive rules, and rel stands for relational rules, both comprised in the UNL Specification.

Attributive and relational rules hold between positions (in the rule string) or indexes rather than words. The grammar also takes a given set of primitive operators (such as '[ ]', for optional; ‘{ }’, for exclusive; '< >' for lemma; '+' for blank space; '#' for word delimiter, etc.) in order to extend the expressive power of the formalism and reduce the necessary number of rules. The ‘@entry’ marker should be stated in every level, and the entry word is to be considered the head of each phrase. As in X-bar theory [6], entry word features are projected to and can be referred by the immediate higher level.

Examples of HERMETO's rules are presented below:

[1] Due to the lack of an ISO 639 code for it, we have been using LI for LIST.

```
; 2.1.2. COMPLEX NOUN PHRASE (CNOP)
CNOP[2] := SNOP + 'and' + SNOP.@entry -> and(:03, :01)
CNOP[3] := SNOP + 'or' + SNOP.@entry -> or(:03, :01)
; 3.3. VERB
VERW[1] := ver.@entry - 'ied' -> :01.@past
VERW[1] := ver.@entry - 'ed' -> :01.@past
VERW[1] := ver.@entry - 'd' -> :01.@past
```

In such a grammar, context-sensitiveness can be stated as internal (dis)agreement between attribute values, such as in:

```
SNOP[1] := DET(GEN:x, NBR:y) + NOU(GEN:x, NBR:y).@entry -> :02.@def
```

The grammar is automatically compiled by HERMETO, which brings it to be an object-oriented scheme, where each non-terminal symbol is defined as an object, to be evoked by the others, during the syntactic and semantic processing. In order to optimize the compilation process, the length of each rule is limited to six symbols, and no nesting is admitted.

Although the expressive power of HERMETO's formalism may be the same as the one stated by EnCo, we claim that it is more intuitive, in the sense grammar writers are no longer supposed to be worried about the position of left and right analysis windows. They can work with (and even import) rules written according to more classic, high-level formalisms in NL understanding tradition.

5 Processes

HERMETO’s resources are parameters for more general, language-independent processes, as splitting, tokenizing, tagging, parsing and semantic processing. These constitute the NL analysis and UNL generation modules. In this sense, HERMETO can be seen as a unidirectional transfer-based MT system itself, where NL is the source and the UNL is the target language.

5.1 Splitting, Tokenizing and Tagging

The process of sentence splitting, in HERMETO, is customized by the user, who is supposed to define, in the grammar, the intended set of sentence boundaries, such as punctuation marks and formatting markers, for instance. Each string of alphabetic characters or digits is considered a token, and blank spaces, as well as punctuation marks and non-alphabetic characters, are understood as word boundaries. Tagging is carried out through the dictionary, and no disambiguation decision is taken at this level. The word retrieval strategy seeks for the longest entries first, in the same way EnCo does. The word choice can be withdrawn, if HERMETO’s parser comes to a dead-end situation.

5.2 Parsing

The tagged string of words is traversed by a chart parser, which applies the left (p) part of the grammar rules according to the priority defined by the user. Backtracking is supported, but cannot be induced. The parsing is rather deterministic, in the sense it provides only one parse tree for each sentence, the one best suited to the rules weight. Part-of-speech disambiguation is carried out during parsing, as the parser gets to the first possible parse tree. Parsing results can be exhibited by the interface and serve as the basis for semantic processing.

5.3 Semantic processing

Semantic processing is carried out together with parsing, in an interleaved way. Although semantic interpretation depends on the result of syntactic analysis, semantic projection rules are applied for any available partial tree, i.e., during the parsing itself. This does not cause, however, any parallelism between the syntactic and semantic modules, as the latter, although triggered by the former, cannot affect it. In this sense, HERMETO cannot deal with any generative semantics approach and is bound to the centrality of the syntactic component. Yet this can bring many difficulties in the UNL generation process, especially concerning the UW choice, i.e., word sense disambiguation, we have not advanced this issue more than EnCo does. The KB solution, which seems to be the most feasible one in EnCo environment, has not been adopted yet, for the trade-off still seems not to be positive, at least so far. As we have been mainly involved with an English sublanguage (the canned structure of English newspaper headlines and leads) and a regularized Portuguese (extracted from the comics), disambiguation can still be solved at the syntactic level.

6 Partial Results

For the POLICARPO and the PUL∅ projects we have been working on the English-UNL and the Portuguese-UNL enconverting respectively. In the former case, we have compiled almost 1,500 web pages, downloaded in September 2002 from the *The NY Times* web site, to constitute our training and assessment *corpora*. Both English-UNL and UNL-Portuguese dictionaries have been already provided for every English word, except proper nouns, appearing in the corpus. The grammar has been split into a core grammar, common to every sentence, and five satellite grammars, specialized in 1) menu items, 2) headlines, 3) leads, 4) advertisements and 5) others. Actually, we have observed that each of these sentence types convey quite different syntactic structures, which can be automatically filtered out of the general corpus. So far, we have already finished the core grammar and the one coping with menu items, and the precision and recall rates, for the assessment corpus, were 77% and 95% respectively, for complete UNL enconverting (i.e., UWs, relations and attributes). Although menu items generally consists on quite simple single word labels, it should be stressed that many of them involved complex morphological structures that had to be addressed by the menu grammar. Anyway, HERMETO, together with the English-UNL dictionary and the core and menu grammars, has proved to be an interesting alternative for fully automatic English-UNL enconverting, at least in this case. For the time being, headlines have been already addressed, but no assessment has been carried out yet.

In PUL∅ project the coverage is rather small. Actually, the project is in its very beginning, and partial results concern a single story, for which HERMETO proved again, not only to be feasible for Portuguese-UNL enconverting, but to be easily integrated in a more complex system as well.

7 Shortcomings and Further Work

At the moment, we have been facing two main shortcomings: HERMETO accepts only ASCII codes and works only in Windows platform. Although we have planned to extend the current version to deal with Unicode and to run under other operational systems, we did not have the time to implement these changes. Furthermore, as we have been working rather on an English sublanguage (the NYT's one) and a sort of controlled (normalized) Portuguese, we have not really faced unrestricted NL analysis problems, which certainly will drive us to reconsider the UNL KB commitments. Therefore, in spite of the results achieved so far, HERMETO has still a long run before it can be considered a really feasible and suitable general NL-UNL enconverting environment. However, as former users of EnCo, we do believe it really represents a user-friendlier environment for fully automatic generation of UNL expressions out of NL sentences.

References

1. Uchida, H., Zhu, M., Della Senta, T. *A gift for a millennium. IAS/UNU, Tokyo, 1999.*
2. UNL Centre. *Enconverter specifications*. Version 3.3. UNL Centre/UNDL Foundation, 2002.
3. Uchida, H,. Zhu, M. *UNL annotation*. Version 1.0. UNL Centre/UNDL Foundation, 2003.
4. Nirenburg, S, Raskin, V et al. 'On knowledge-based machine translation', *Proceedings of the 11th International Conference on Computational Linguistics,* Bonn, 1986.
5. Furuse, O, Iida, H. 'Cooperation between transfer and analysis in example-based framework', *Proceedings of the 14th International Conference on Computational Linguistics*, Nantes, 1992.
6. Chomsky, N. 'Remarks on Nominalization', in Jacobs, Roderick A. & Rosenbaum, P. (eds.), *Readings in English Transformational Grammar*, Waltham (Mass.): Ginn, 1970, p. 184-221.

A Platform for Experimenting with UNL

Wang-Ju Tsai

GETA, CLIPS-IMAG
BP53, F-38041 Grenoble cedex 09 France
Wang-Ju.Tsai@imag.fr

Abstract. We introduce an integrated environment, which provides the initiation, information, validation, experimentation, and research on UNL. This platform is based on a web site, which means any user can have access to it from anywhere. Also we propose an XML form of UNL document as the base of future implementation of UNL on the Internet.

1 Introduction

Since proposed 5 years ago, UNL project has attracted 16 international teams to join and is regarded as a very promising semantic Interlingua for knowledge representation on the Internet. The articles and applications of UNL have been found in many domains such as: machine translation, information retrieval, multilingual document generation, etc. Now we can find on the Internet not only the web sites of UNL language centres but also some discussions. The applications to facilitate the usage of UNL have been produced as well. Now we see the need to create a platform to integrate these applications also to introduce UNL to new ordinary users. We create this platform on a web site SWIIVRE (http://www-clips.imag.fr/geta/User/wang-ju.tsai/welcome.html), which has several goals: for the initiation, information, verification, research, and experimentation of UNL. And since this platform is based on a web site, any user from anywhere can have access to it.

2 Introduction of the Site SWIIVRE

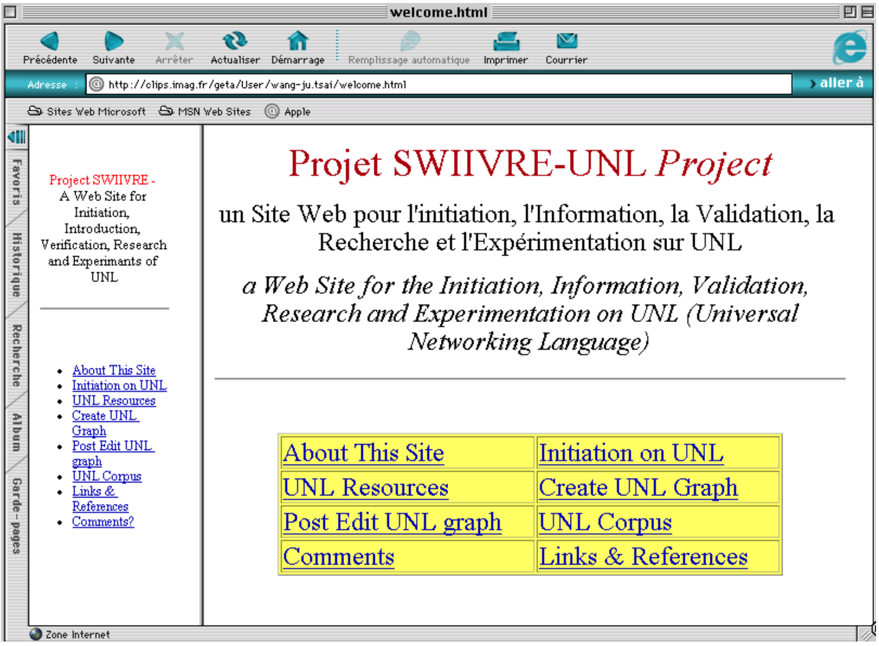
In Appendix I we list all the resources accessible for UNL society members from internet. We can find out that most of the LC's connect vertically to UNL Centre but the horizontal connection among LC's is not enough, which means any user who wants to try the multilingualism of UNL will feel frustrated, since he will need to spend a lot of time try out every LC to know what service he can get.

The main purpose of this site is rather to integrate the current UNL applications and complete the services of Language Centres', when the function is available on a Language Centre, we simply provide the link to it, we also produce some applications to integrate or provide new functions, which all serve to facilitate the usage of UNL. Also we collect the useful information and publications on UNL, the web site is updated regularly. Lastly, by collecting the useful information and recording the related

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 261–267.

data, this site finally can serve as an evaluation of the performance of UNL community.

Here we show the welcome page of this site:



Project SWIIVRE - A Web Site for Initiation, Introduction, Verification, Research and Experimants of UNL

- About This Site
- Initiation on UNL
- UNL Resources
- Create UNL Graph
- Post Edit UNL graph
- UNL Corpus
- Links & References
- Comments?

Projet SWIIVRE-UNL *Project*

un Site Web pour l'initiation, l'Information, la Validation, la Recherche et l'Expérimentation sur UNL

a Web Site for the Initiation, Information, Validation, Research and Experimentation on UNL (Universal Networking Language)

About This Site	Initiation on UNL
UNL Resources	Create UNL Graph
Post Edit UNL graph	UNL Corpus
Comments	Links & References

The following is a description of each link on the welcome page:

About This Site This page provides the introduction, why and how this site exists, the site log and current status of this site, also the new projects to come on this site, lastly all the recent activities of UNL community. When clicked, a news flesh will also show the most recent UNL activities and the new updates on this site. In the future, we think we will at least UNL-ise this page to demonstrate the multilingualism of UNL.

Initiation on UNL This page is to help users to take a first step in UNL, understand how UNL works. We first provide a copy of most recent UNL specifications, for the moment only Spanish Centre has prepared a “multilingual interactive page” can serve as the tutorial and give examples to each UNL relations, thus we put a link to this page. When UNL becomes more well known, there will be more and more tutorials for beginners in the future. Or we might finally create an graphical interface for user to manipulate and show the spirit of UNL. We would also like to introduce the XML-UNL document here. We put an example of XML-UNL document here and with the help of XSLT, we can create the same effect like UNL browser, then the users can choose to read the document in the language they wish. We will explain later in the article why we want to XML-ise a UNL document.

UNL Resources This page provides all the UNL<->NL deconverters / enconverters, dictionaries that are accessible on the Internet. Some deconverters accept the deconversion of one single UW (Universal Word), in this case they can serve as the UNL-NL dictionaries. We can simply add some scripts in our site to help users to access these deconverters as if they are accessing dictionaries. In the future, the status report of each server will be added; we hope we can provide "UNL daily bulletin" to report the updates and status of each server. Currently only French server report can be seen. To complete the services, we developed a "multilingual simultaneous deconverter" (Preedarat 2001), which can handle several deconversions at one time. Users can click on the language versions they want as output, the program will contact these servers at once, thus they don't need to do the deconversions one by one, and they can experience the automatic multilingual generation.

Create UNL Graph Since ordinary users are not able to write UNL graph without being trained, to help users create UNL graph will be an important function to develop. In this page we collect the links to accessible UNL editors, including editor for professional writers or for beginners. We have put a link to our "Basic UNL graph editor" (Preedarat 2001), which is implemented by using a similar XML-UNL format and XSL transformation. The users can manipulate the UNL graph represented in tree-like structure, and save the result in XML format. We also put a link to the "interactive multilingual page" of Spanish Language Centre, here users can manipulate the UNL graph by the options provided, actually users can already generate many sentences based on these examples.

Post-Edit UNL Graph This function is still under development. Our idea is to provide the users the possibility to correct the UNL document after it is deconverted. It provides ordinary users with the ability to correct the faults in the UNL graph and improve the quality of graph.

UNL corpus We collect all the UNL corpora here, and also we are currently working on designing a data base to store these corpora thus to facilitate the further exploitation or calculation. We can finally design an interface to allow users to upload the corpora in different forms, or produce the forms they desire. In Appendix II we show the first statistics we made on the corpus FB2004.

Comments To send comments to the maintainers of the site.

Links & References We collect all the links to UNL Centre, Language Centres, articles, papers, discussion of UNL, and users can trigger the search engines here to find more information about UNL when they want.

3 XML-UNL document

The applications compatible to XML have been increasing a lot and XML can replace HTML as the next norm of a web-based document. And from an XML form, we can further produce other form, exchange or integrate the existing data easily. It would thus be reasonable to XML-ise the UNL document. We would like to propose here an XML form of UNL document as in Appendix III. We created this DTD according to the UNL specification Version 3 Edition 1 (20/02/2002). Based on this DTD, we can create the UNL document in XML form, with an XSL Transformation we can produce the same effect as an UNL browser. Further more, we can easily expand this DTD to enable the XML-UNL document to register all the modifications and corrections on a UNL document, this can be very useful in our post-edition project.

4 Conclusion

We have made the first step in the integration of all the UNL components on a website. Next step is to streamline the procedures between current functions and to include more services.

References

Boitet Ch. (2001) Four technical and organizational keys for handling more languages and improving quality (on demand) in MT, " MT-SUMMIT VIII (2001) ", Proceedings of the Workshop (Towards a Road Map for MT), p.14-21. 18/09/2001

Coch & Chevreau (2001) Interactive Multilingual Generation. Proc. CICLing-2001 (Computational Linguistics and Intelligent Text Proceeding), Mexico, Springer, pp. 239-250.

Sérasset G. and Boitet Ch. (2000) "On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter", COLING 2000, Saarbruecken, Germany 31/07-04/08, p.768-774

Sérasset G. & BOITET Ch. (1999),"UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction" MT Summit 99, 13-17 september 1999, Singapore, pp 220-228.

Boitet Ch. (1999) A research perspective on how to democratize machine translation and translation aids aiming at high quality final output, Machine Translation Summit VII (1999), Singapore, 13-17/9/99

Munpyo HONG & Olivier STREITER (1998) "Overcoming the Language Barriers in the Web: The UNL-Approach" , in 11.Jahrestagung der Gesellschaft für Linguistische Datenverarbeitung (GLDV'99), 1999, Frankfurt am Main.

Preedarat JITKUE (2001) Participation au projet SWIIVRE-UNL et première version d'un environnement web de déconversion multilingue et d'éditeur UNL de base, report de stage de Maîtrise Informatique Université Joseph Fourier – Grenoble 28/05-31/08

Appendix I: The resources accessible at each LC for UNL society members

	Enco	Deco	Dico	Introduction of UNL system	Linked by UNLC	Remarks
Arabic	√	√	√	Arabic	√	
Chinese		√		English	√	
French		√				
Indonesian				Indonesian		
Italian		√		Italian	√	
Russian		√	√	English		
Spanish		√		English Spanish	√	Tutorials / Interactive Page / Document Repository
Thai				Thai	√	
UNLC			√	English		UNL specs/ development modules

Appendix II: Some Statistics about FB2004 Corpus

Corpus Name : FB2004
Original Language : English
Other available versions : French, Spanish, Italian, Russian, Hindi, UNL
No. of Sentences : 122
No. of Words : 2799
No. of Relations in UNL: 1519

Part I. The relation count

Relation	Outside scope	In scope	TOTAL	Relation	Outside Scope	In scope	TOTAL
AGT	**66**	**10**	**76**	SEQ	0	0	0
AOJ	**64**	**37**	**101**	FMT	5	0	5
OBJ	**225**	**89**	**314**	FRM	6	3	9
AND	**63**	**120**	**183**	PLF	0	0	0
OR	26	3	29	SRC	2	0	2
BAS	2	2	4	GOL	17	7	24
CAG	0	0	0	PLT	1	0	1
CAO	0	0	0	TO	5	1	6
COB	1	1	2	INS	0	0	0

PTN	4	1	5	**MAN**	**49**	**17**	**66**
BEN	7	5	12	MET	10	3	13
PUR	28	1	29	PER	0	0	0
CNT	22	6	28	QUA	12	5	17
MOD	**263**	**186**	**439**	PLC	17	3	20
NAM	21	15	36	SCN	13	5	18
POF	5	2	7	TMF	2	0	2
POS	17	8	25	TMT	0	1	1
CON	2	0	2	VIA	1	0	1
RSN	1	0	1	DUR	5	4	9
COO	4	2	6	TIM	20	5	25

Total no. of relations: 1519

Remarks

(a) The 6 most frequently used relations are marked in bold type. The result is not surprising, since these relations have either an important or a broad usage. MAN and AGT's usage are frequent though straight forward. Besides its own static verb and copula usage, AOJ also shares part of adjective-noun relation, otherwise the frequency of MOD will be even higher.

(b) AND relation appears much more frequently within a scope, which is not surprising, since scope is used to represent the union of the similar things or ideas, and AND relation links these UW's in te scope.

(c) Some other relations' usage is not very braod, so they didn't appear.

Part II. Attribute count

(1) Time Attribute
.@past 40 / **.@present 114** / **.@future 187**

(2) Aspect Attribute
.@complete 20 / .@progress 13 / .@state 16 / else 0

(3) Reference Attribute
.@generic 9 / **.@def 659** / .@indef 79 / .@not 2 / .@ordinal 8

(4) Focus Attribute
.@entry 530 / .@topic 48 / .@title 21 / else 0

(5) Attitude Attribute
.@exclamation 1 / else 0

(6) Viewpoint Attribute
.@ability 7 / .@obligation 7 / .@possibility 8 / .@should 2 /
.@unexpected-consequence 2 / else 0

(7) Convention Attribute
.@pl 558 / elso 0

Remarks

The original text langue is English, so the frequency of .@pl, .@def, .@indef and time attributes are among the highest. If the original language is one of those isolated languages, such as Thai, Vietnamese, Chinese, which does not provide so much information about definitiveness or time, it might be difficult to use or to decide these attributes. It's not because that the graph authors or enconverters are bad, it's simply because they can't find this information from the text when encoding.

Appendix III. An XML form of UNL document

```
<!DOCTYPE D [
<!ELEMENT D (P+) >
<!ELEMENT P (S+)>
<!ELEMENT S (org,unl,GS+)>
<!ELEMENT org (#CDATA)>
<!ELEMENT unl (#CDATA)>
<!ELEMENT GS (#CDATA)>

<!ATTLIST D dn CDATA
    #REQUIRED
    on CDATA #REQUIRED
    did CDATA #IMPLIED
    dt CDATA #IMPLIED
    mid CDTAT #IMPLIED>
<!ATTLIST P number CDATA
    #REQUIRED>
<!ATTLIST S number CDATA
    #REQUIRED>
<!ATTLIST org lang CDATA
    #REQUIRED
            code CDATA #IMPLIED
    >
<!ATTLIST unl sn CDATA
    #IMPLIED
    pn CDATA #IMPLIED
    rel CDATA #IMPLIED
    dt CDATA #IMPLIED
    mid CDTAT #IMPLIED>
<!ATTLIST GS lang CDATA
    #REQUIRED
    code CDATA #IMPLIED
    sn CDATA #IMPLIED
    pn CDATA #IMPLIED
    rel CDATA #IMPLIED
    dt CDATA #IMPLIED
    mid CDTAT #IMPLIED>

]>

<!-- GS = generated sentence -->
<!-- dn = document name -->
<!-- on = owner name -->
<!-- did = document id -->
<!-- dt = date -->
<!-- mid = mail address -->
<!-- lang = lang tag -->
<!-- code = character code name -->
<!-- sn = system name -->
<!-- pn = post editor name -->
<!-- rel = reliability -->
]>
```

A Framework for the Development of Universal Networking Language E-Learning User Interfaces

Alejandro Martins,[1] Gabriela Tissiani,[2] and Ricardo Miranda Barcia [3]

[1] Distance Learning Laboratory, Federal University of Santa Catarina,
Campus da Trindade, Florianópolis, SC, Brazil,
martins@led.br

[2] CNPq Researcher at Centre Universitaire d'Informatique,
University of Geneva, Switzerland,
Gabriela.Tissiani@cui.unige.ch

[3] Distance Learning Laboratory, Federal University of Santa Catarina,
Campus da Trindade, Florianópolis, SC, Brazil
rbarcia@eps.ufsc.br

Abstract. The UNL infrastructure aims to overcome the language barrier on the Internet. At the same time, distance learning (DL) is becoming the best way to promote the knowledge diffusion across countries. However, the distance learning process still presents some obstacles to be overcome. The UNL can help to reduce particularly those problems and to provide a common educational environment across different languages. Here we discuss the development of the UNL version of an existing web platform for distance learning. The overall goal of this project is to create a framework to support the development of UNL user interfaces applied for e-learning platforms.

1 Introduction

This paper presents part of a research project that aims to build an e-learning platform using Universal Networking Language (UNL) technology.

It envolves the prototype development of the UNL version of an existing e-learning platform called VIAS-K (Virtual Institute of Advanced Studies - Knowledge Environment). This platform is provided by the Distance Teaching Laboratory, LED, from the Federal University of Santa Catarina, UFSC, Brazil [15]. It supports a huge group of interactive models composed of actors, contents, management, users support and collaborative tools. In order to full fill each user's specific needs, theses models will consider also the variety of users' mother language, based on the UNL system.

Although UNL have been developed with success, it is still a brand-new technology [17]. It is an artificial language that exchanges the knowledge from a natural language to make possible the access of its content through different languages. With the purpose of promoting the development of UNL and the effectiveness of DL, this research proposes a case study that brings UNL into the existing VIAS-K environment.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 268–275.

A framework for UNL user interface design will be established in order to help developers to well represent e-learning contents from diverse linguistic sources and cultural backgrounds. Its research includes the implementation of modules that will allow the visualization of the case study in three natural languages: Portuguese, French and English. Moreover, it may contribute to expand both fields of research, UNL and DL, through its diffusion by an e-learning application. The project takes in account graphical user interface principles and ergonomics aspects as well as usability ones.

In this prototype, the UNL infrastructure of multilingual functions is being used: 1) to generate a framework for UNL user interfaces applied to e-learning platforms as well as subsidies to build a guideline for UNL interfaces in general; 2) to create new concepts to the UNL KB (new UWs); 3) to provide visualizations of VIAS-K platform in different languages; and 4) to improve what is known in the fields of UNL and DL.

2 Background

Education in its several modalities has become more and more indispensable to the formation of highly skilled professionals that can really answer the needs of the globalized market. The power of knowledge and principles becomes part of a system of innovations, not a moral or cultural force, but an incubator of new industries in an economy dominated by technology [15].

Higher Education Institutions need to work along with the productive sector in order to satisfy the demands of professional formation and qualification, as a consequence of the following aspects [7][13]: a shift from the model centered on the campus to a model centered on the student; greater flexibility of the teaching institutions for their own survival; a considerable increase of the demand for further education, having as a common cause the continuous teaching and, consequently, generating the need of increasing the number of places to satisfy it; social pressure for knowledge (globalization, productivity, explosion and generation of new knowledge).

The factors abovementioned show that the educational system is not ready to develop in the same pace of the technological changes. Therefore, it is necessary to search for new approaches to the training of learners within a new enterprising vision. Thus, the most reliable way for an infrastructure of education passes through a change in the existing educational model. The adoption of distance educational programs along with the use of other media seems to be a possible solution. In addition to it, it is necessary to improve the development of environments with several languages to reach the maximum of people. A Technology such as UNL combined to the construction of friendlier user interfaces will certainly make use of all the Internet potenciality to help in the diffusion of knowledge across the world [4].

It is paramount to insist on continuous, open, personalized and collaborative education that allows individuals to update their knowledge throughout their professional lives, no matter their geographic or temporal settings;

2.1 E-learning Platforms

Over the last years, attempts have been made to allow the construction of learning platforms that can make possible the transmission of contents in an efficient and meaningful way by Internet.

According to Rosenberg [11], “Web-based technology is the key to a deep revolution on learning». The challenge is to transmit the information to reach the greatest possible number of users, independently of different features and purposes. In this context, the development of web applications is required to allow the transmission of content in many different ways and to many different kinds of users, leading then through the environment and customizing it to every need and goal.

In VIAS-K's case, many different parameters can be adapted, considering all the different styles of users (actors) that use the platform. Each user category may have an adapted platform in terms of content, navigation bars, and interactive tools.

VIAS-K platform target audience is performed by isolated and geographically dispersed students that are looking for specific kinds of knowledge (knowledge on demand) and by specific institutions interested in promoting web-based closed courses for its collaborators.

Among it we believe that the experience of VIAS-K is a good case to be developed as an UNL application, since it introduces an important issue to the e-learning field: it is a case where the user, developer or administrator have the power to customize the interface. Thus, why not make it really universal, giving the user the possibility to visualize the contents in this/her mother language? The VIAS-K platform is presented as a case study model to diffuse the knowledge across different countries, and contribute to the United Nations initiative of creating a multilingual infrastructure on UNL.

2.2 UNL and Distance Learning

The UNL is a multilingual system that involves linguistic aspects besides engineering ones. Although it is still at an early stage of application development, most of the research for establishing it as a valid tool to overcome linguistic divide has already been done, since 1997, by 16 computational linguistics research groups all over the world [18] [19]. The reason why UNL is being adopted in this project is because it is an emerging technology that will be largely used in the Web. To substantiate this fact one can be aware that the patent of UNL was requested by the UNL secretary general itself.

Although most of the UNL infrastructure and architecture has been already researched and designed, there is still a lot of work to be done. One of the goals to be reached during the UNL application development process is how to represent the distribution of a same textual content in different languages, considering graphical user interface elements and its ergonomic rules. Since it must be adaptive, it is necessary to research and develop techniques that allow making standard user interface principles applicable to them.

In order to assure its functionality and aesthetic, it is needed to situate its special interface issues in the context of HCI-systems (Human Computer Interactions) [1–3].

The application of usability evaluations of the text content as well as the textual menus and the labels for the navigational task may contribute to provide the desired framework.

Considering the DL context, UNL can be used to increase the use of the technology, since it allows a broad diffusion of the information, extending its range to people outside of a specific place. For instance, the same course directed to people from Swiss could also be taught to people from Egypt or Brazil. If the content of the course is made with UNL, the students who will access the course will be able to understand its content when they get into the learning platform. In other words, people from different countries living far from each other, will be able to share their experiences in the same environment using their own language.

In the VIAS-K application, UNL may be used to represent the text included in the contents. The existing texts will be translated according to the language selected by the user when s/he first accesses the environment. Technically, UNL will represent all the text as UNL sentences. The UWs will be chosen and the UNL sentences will be built. After that, the dictionary entries will be selected to every language that it is going to be able to translate. The last step is to build the set of rules to execute the translation. With the sentences and the language and rules dictionaries, a "proxy" will be built to allow the interpretation of the UNL by a browser.

Even though UNL is not totally developed yet, experiments carried out worldwide have shown that it has a great potential, and that it can be one of the ways to answer the ever-growing demand for education across the world.

3 The Case Study

To achieve the HCI case study diagnosis the Shneiderman's taxonomy [12] is established as the basis for the usability evaluations. This approach may guarantee the analysis of usability and HCI rules requirements.

Finally, the framework will be generating based on the evaluation. As a result, it may contribute to the proper design and represent of UNL e-learning user interfaces for a universal understanding as well as to improve the power of UNL system's communication.

The UNL VIAS-K version being here proposed will be built to allow comparisons with the original one. For validating a new proposition, which will be generated through Shneiderman's taxonomy [12], tests will be applied using both the original VIAS-K version, in Portuguese, and the UNL version, in English and French. In the first case, the tests will take place with Brazilian students at UFSC. In the second one, the tests will happen in Switzerland, with students at UniGE (University of Geneva).

3.1 The evaluation of VIAS-K UNL version

Although the UNL version of VIAS-K is still at an early stage of development, its construction can be helpful in providing theories for UNL interface design. Therefore, it may be helpful in defining e-learning systems properties and situating its special interface issues in the context of HCI (Human Computer Interaction) field. Nevertheless impressive researches about adaptive approaches to user interface design have been made and analyzed, a few recommendations have been leaded, in order to give a briefing of its task performance. This topic intends to establish a sort of considerations about VIAS-K UNL version design as a way to point out some criteria for the design of the e-learning user interface.

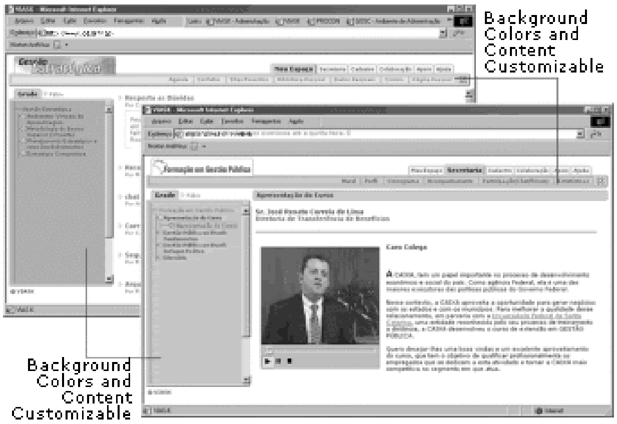

Background Colors and Content Customizable
Background Colors and Content Customizable

Fig. 1. Two different courses on VIAS-K platform

The prototype is currently being implemented. The first part of the project focuses on designing and prototyping the chose content in there natural languages: English, French and Portuguese. To achieve it, the following UNL elements are being developed:

- UWs related content;
- UNL sentences;
- English, French and Portuguese dictionaries and grammar rules;

After that, the UNL system will be implemented into the VIAS-K platform.

In order to produce an equivalent UNL for a the original VIAS-K contents, the UNL editor of the appropriate Language Server will be used to start the process called “enconversion”. After that, the UNL viewer is used to "deconvert" the UNL text into the user’s natural language, by using the UNL viewer of hi/her appropriate Language Server.

4 Framework for the design of UNL e-learning platforms

According to Shneiderman [12] the user interface designers must: “try to predict subjective satisfaction or emotional reactions”. It implies that the user may have control on what he wants to see.

Control is only one of the eight golden rules, presented by Shneiderman [12] based on his user interface studies. It is considered that these are the rules that can assure a successful interface. They are used to formulate the proposed earlier framework, which is presented below. This framework corresponds to the rules that will be applied to evaluate the comparisons between the use of the original platform version and the UNL one. They comprises:

1. **Consistency:** Set standards and keeping the elements of design, specially the navigation and support menus, located in a consistent way from screen to screen, from version to version and from window to window, independently of the labels and text blocks visual mass. According to Shneiderman [12], identical terminology should be used in prompts, in menus and help screens. It means that the UNL system may be planned in order to assure a consistent customization among all the language possibilities that can be displayed. A consistent presentation of menus allows the user access then during the whole interaction, which helps to reduce the disorientation caused by many levels of contents. This concept includes standards qualities of graphic design like: a) Grouping: group elements by connecting the items that are similar, such as content, navigation and support menus on the screen; b) Hierarchy: display established groups, such as navigation menus and hyperlinks, in a logical sequence according to the target audience and the context of tasks to be performed; c) Relationship: reinforce grouping and hierarchy by supplying elements related to each other through the use of colors, image representation, alignments, etc [1] [5] [8] [9];

2. **Shortcuts:** Frequently users prefer carry out then tasks faster, which means a reduction in the interaction processes. Shortcuts can enhance tasks procedures (choosing and performing actions faster). A good adaptation may allow the representation of actions and tools by its initials, which may be dynamically adapt to the different languages that the UNL system can present. Another important thing to be carried out involve the use of icons [6]. The iconic signage for menus, actions and tools has cast doubts. All the people recognize not all the symbols. In an international interface, the icons must be integrated with words in the same small communication unit, at least by a "tool tip". The native language of the user may also convert this small text unit.

3. **Feedback:** It is important to provide feedback during the whole user navigation as well as while the user access the website. We believe that in the case of an e-learning platform, which uses UNL technology, the system may react promptly on the whole application to avoid subsequent delays.

4. **Closure:** To measure the duties amount (that the user will have to accomplish in the platform) or how long it will take to navigate (searching for tasks or accomplishing some goal) can be stressful and discouraged. Grouping tasks and let the user know how much effort it will take to accomplish assignments and/or how much longer it will take to navigate are important issues to motivate the user. This will also provide us a feedback of the UNL translation, since we can make comparisons between the original and the UNL versions of VIAS-K.

5. **Error:** Either anticipation or handling should be considered for any sort of application. A common error that can occur in a user interface is the bad text position that may be consequence of bad adaptation or can be caused by bad interpretation of the content, menus navigation and labels, as well as dialog boxes texts.

6. **Reversal of actions:** According to Shneiderman [12], the actions should be reversible as much as possible. It has to do with the control and the feedback properties, and all of then together can assure the user satisfaction. An UNL application must keep both the web browser and the navigation menus always present in all pages, so that the actions of going “forward” and “backward” can be made independently of the browser. That’s why it is important to assure the good translation of menus and navigational tasks.

7. **Control:** Give to user the power to control his actions rather than make him follow or respond automatic events will promote more involvement and better results throughout the interaction. The purpose of the UNL system is to facilitate the communication rather then creates a negative reaction of the user against the bad control over system’s adaptation options. That’s why it is important to make as much options available to the user as possible, in an organized way, aiming to help the user on his own decisions about the languages he/she want to receive the information [8, 10].

8. **Reduce short-term memory load:** In order to assure comprehensible text displays, it is essential to consider that information load is quite big to be remembered by the user. It is important to keep the whole interface simple, especially because in the case of the UNL version it may became understandable in any state of customization.

Our future work is the validation of the VIAS-K platform UNL version, based on these rules presented above. The product of the evaluation may allow us to reformulate the proposed framework, based on its effective results.

5 Final Considerations

All of the aspects cited above can be considered a challenge for UNL developers. This paper describes the prototype that is being developed as a current project for web-based distance learning. Starting from its development experience, it will be possible to describe a number of issues and techniques that may be considered to help enhance the presented framework for the development of UNL e-learning environments as well as user interfaces for general UNL applications. This classification may help web designers to achieve an effective and harmonic UNL interface, the activeness of usability and assure the 8 golden rules.

References

1. Ambler, S. W. User Interface Design: Tips and Techniques. USA: SIGS Books/Cambridge University Press, 1998. (www.ambysoft.com/userInterfaceDesign.pdf).
2. Apple Computer (1992). Human Interface Guidelines: The Apple Desktop Interface, USA: Addison-Wesley. 2nd Edition, 1992, 410p.
3. Cybis, W. "Apostila do LabUtil: Recomendações para Desgin Ergonômico de Interfaces." BR: Programa de Pós- Graduação em Engenharia de Produção. UFSC, 1997,145 p.
4. Iba, W., Hirsh, H., & Rogers, S. Machine Learning Special Issue on Adaptive User Interfaces. Call for papers of AAAI 2000 Spring Symposium on Adaptive User Interfaces, at Stanford University March 20-22, 2000. (www.isle.org/~aui/aaaisymp00.html)
5. Jacobson, R. E., Information Design, England: The MIT Press. 1st Edition, 2000. 357 p.
6. Marcus, A. (1998). Metaphor Design in User Interfaces. The Journal of Computer Documentation ACM/SIGDOC, New York, NY, May 1998, volume 22, No. 2, pp. 43-57.
7. Modelagem de um Ambiente Inteligente para a Educação baseado em Realidade Virtual. IV Congresso RIBIE, Brasília, 1998. http://phoenix.sce.fct.unl.pt/ribie/cong_1998/ trabalhos/106/106.html.
8. Mullet, K. and Sano, D. Designing visual interfaces- communication oriented techniques. USA: SunSoft Press, 1st edition, 1995, 273 p.
9. Nielsen, J. Usability Engineering. USA: Morgan Kaufmann Publishers. 1993: 1st edition, 362 p.
10. Galitz, W.O. The Essential Guide for User Interface Design. USA: Wiley Computer Publishing, 2nd Edition,1997, 743p.
11. Rosenberg, Marc J., E-learning: estratégias para a transmissão do conhecimento na era digital. São Paulo: Makron Books, 2001.
12. Shneiderman, B. Designing the User Interface: Strategies for Effective Human-Computer interaction, 3rd Edition. USA : Addison Wesley Longman Inc, 1998. 638p.
13. Soares, M.V., Padronização: Tendências ..., Artigo da Disciplina Tópicos em Engenharia e Computação V, UNICAMP, 1999.
14. Tissiani, G.; Garcia, F. Guideline for Adaptive Graphical User Interfaces Using Universal Networking Language. SPAIN: to be in ICTE Conference Proceedings, November 13–16th, 2002.
15. Tissiani, G.; Bortolon, A.; Fialho, F.; Garcia, F.; dos Santos, J. S. Virtual Reality and Universal Networking Language: a case study for Distance Learning. SPAIN: to be in ICTE Conference Proceedings, November 13-16th, 2002.
16. The Knowledge Factory. The Economist, A Survey of Universities. Outubro, 1997.
17. Uchida H., Zhu M. and Della Senta T., The UNL, a Gift for a Millennium, Institute of Advanced Studies, United Nations University, 1999
18. Uchida H., Zhu M., The Universal Networking Language beyond Machine Translation, presented in the International Symposium on Language in Cyberspace, Seoul, 2001

A WEB Platform Using UNL: CELTA’s Showcase

Lumar Bértoli Jr.,[1] Rodolfo Pinto da Luz,[2] and Rogério Cid Bastos [3]

[1] Instituto UNDL Brasil, Rodovia SC 401 - Km 1 –
ParqTec Alfa, Ed. CELTA - Bloco B - 3º pavimento - Módulo 2.10
João Paulo – 88030-000, Florianópolis– SC- Brasil
bertoli@undl.org.br

[2] Instituto UNDL Brasil, Rodovia SC 401 - Km 1 –
ParqTec Alfa,Ed. CELTA - Bloco B - 3º pavimento - Módulo 2.10
João Paulo – 88030-000, Florianópolis– SC- Brasil
luz@undl.org.br

[3] Rogério Cid Bastos, Departamento de Informática e Estatística, UFSC,
Campus Universitário - Trindade - 88040-900, Florianópolis - SC – Brasil
rogerio@inf.ufsc.br

Abstract. Economic globalization is changing the way companies communicate. The ease and speed of accessing information and taking decisions is better for everyone on the decision chain. To speed up access to information it is important to present information in one’s native language and create a language-independent communications channel. To assist business-to-business operations, the development team at the Instituto UNDL Brasil designed the pilot project “CELTA's Showcase” to demonstrate that it is possible to create a multilingual business-to-business platform using UNL.

1 Introduction

The interconnection between producer and consumer is becoming extremely important. The expansion of markets from local to global influence requires the use of new technological resources in order to support the majority of these relationships. In addition, the specialization of markets requires the development of automated tools to facilitate the pairing of small groups of producers and consumers.

Due to the irreversible globalization of markets and the specialization of production areas that create high technology products, there is a growing need for perfect matching between producers and consumers, to allow maximum performance in efforts to connect both sides.

To increase the chances of matching the best producer-consumer pair, the *Instituto UNDL Brasil is* proposing a project in this field. The main objective of this project is the development of a multilingual Web platform that allows integration between producing companies and their customers. This project is being developed by the *Institute UNDL Brasil*, and was made possible by the creation of the *UNL Research and Development group (R&D)* in the year 2003 [1]. The R&D group has a highly trained IT team whose main objectives are:

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 276–285.

- UNL research; including the creation of tools and linguistic resources for Portuguese; and
- Application development using UNL for the Brazilian market.

The *Instituto UNDL Brasil* is located at the technology base company-incubator, CELTA [2]. CELTA is composed of startup companies that work with high technology and that are based in Santa Catarina State, Brazil. CELTA’s objectives are to support startup companies in their administrative procedures, and to provide an atmosphere of technological interaction among them, the scientific community, and the market. CELTA is currently the largest high technology incubator in Latin America, incubating more than 30 companies in areas such as electronics, computer science and knowledge management [2][3].

The *Instituto UNDL Brasil has* presented a proposal that recognizes the concrete needs of the companies at CELTA. This proposal is the pilot project "CELTA’s Showcase". The deliverable product of this project is a multilingual Web platform to serve startup companies incubated at CELTA. This project involves the development of a platform that will allow CELTA’s companies to display their products and services, and to exchange information with their customers, partners, investors, suppliers, and potential customers. Initially it will support six languages defined as requirements by the incubated companies. More language options will be available as UNL evolves and UNL System integration becomes available. Product sales are often targeted only to markets of English and Portuguese speakers, reducing the chances of enlarging a company’s market share.

The main objective of this project is to enhance the market interaction channels of CELTA’s companies. The approach will increase the exchange between the companies’ staff and potential foreign customers by allowing each party to use their native language.

2 Platform description

The pilot project "CELTA’s Showcase" consists in the development of a platform to enlarge market share for companies incubated at CELTA.

CELTA’s Website currently presents information only in Portuguese, impeding information access by foreign investors' and customers. In addition, there is no specific information about the companies; there is only a link to each company’s website with no other description at all. This link allows the visitor to reach the company’s website’s if it has one.

Some companies display their products on their own website. In order to broaden the website’s accessibility, some websites were translated manually for English and a few for Spanish. At this moment, two solutions have been used to accomplish this translation task. The cheaper solution is to leave this task in the hands of company staff (usually IT or administrative staff, and not professional translators). A more expensive and suitable solution is to contract a service or a professional to perform the task. In addition, these translations are only valid at the time of the translation, which creates a need for continuous contracting of translation services to keep the website’s information updated. Unfortunately, incubated companies cannot implement these solutions because in the first case they involve a misuse of highly qualified staff, and in the second because of high costs.

The solution presented here was based on the analysis of the companies' requirements. It resulted in detailed specifications for the platform.

The platform will allow all of CELTA's incubated companies to have their information registered in a standardized multilingual website. This website will allow visitors, such as customers and investors, to access the companies' multilingual information. The principal content that will be available for visitors are:

- General company information;
- Products and services; and
- Customer lists

The content should be displayed using the chosen languages: English, Spanish, Chinese, German, and Arabic, in addition to Portuguese. These six languages encompass a high percentage of potential investors and consumers for CELTA companies., With the use of UNL, the insertion of more languages will not require complex changes and could be included at any phase of project implantation.

In addition to the content, an interface will be provided to allow interaction between visitors and CELTA companies. This interface will permit communication between companies and their customers and investors. This tool will allow language independent communication. The companies' staff will use the Portuguese interface, and the other party (customers and foreign investors) will use their native language, limited to the six languages supported by the platform.

The software development methodology adopted is the evolutionary prototyping model [4]. This methodology is based on software development using traditional cyclical software engineering phases, where all phases are repeated in the correct sequence until a deliverable prototype is reached. It was selected for two main reasons: 1) the UNL System is not yet fully operational and certified; 2) CELTA's company members are constantly changing and new requirements could emerge.

The platform is divided into three views: visitor, company, and administrator as presented in figure 1.

The visitor's view gives access to a website with information about all of CELTA's companies. Using this view, the user is able to navigate through each company's content. In addition to the navigation option, a search engine based on UNL will be available to help the visitor find specific information about the companies and theirs products and services. UNL will be used in this case not only for simple translation, but also to confirm that the correct meaning is being selected. This will be possible by integration with the UNL Knowledge Base (KB). Using the proper tools embedded in the UNL KB, the identification of the precise meaning of a concept is possible. For example, a visitor can search for a particular field, and all directly and indirectly related products will be displayed.

At each company's space, there also will be a tool that allows visitors to communicate with the company in his or her native language. This tool initially will be limited to the platform language support and later to UNL System support. As the platform development advances, this tool will transform from a fully restricted content solution to one with free text domain restrictions.

The UML use case [5] for the actor visitor is presented in figure 1(use case "a"). The visitor's view will be implemented as the *visitor's module*.

The company's view, allows a company's staff to manage the company's content and to interact with visitors that have sent messages through the visitor's view. The UML use case of the actor company is presented in figure 1(use case "b"). The company's view will be implemented as the *company's module*.

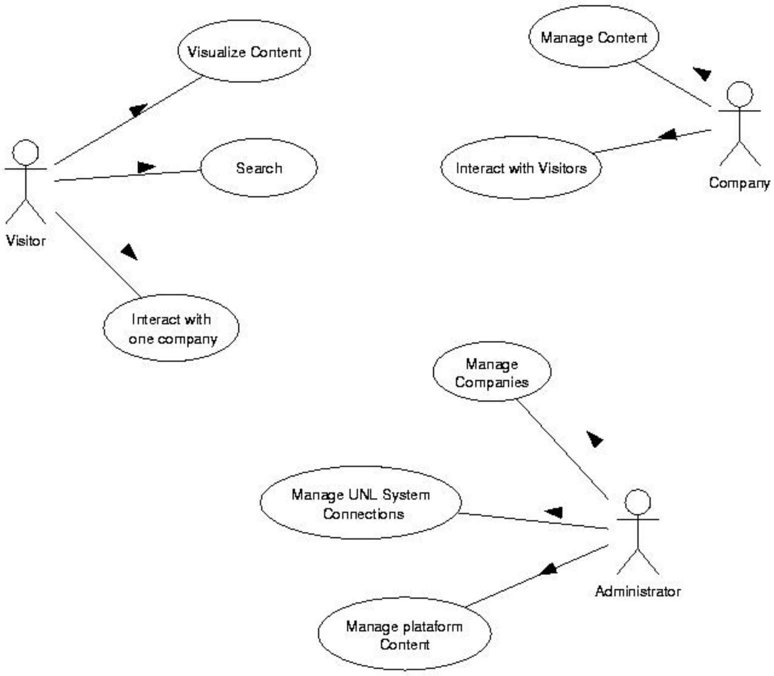


Fig. 1. UML use cases

The administrator's view allows CELTA administration to manage registered companies, to manage the connections with the UNL system and to administrate the platform's overall content. The UML use case of the actor administrator is presented in figure 1(use case "c"). The administrator's view will be implemented as the *administrator's module*.

The project was divided into 3 main phases (prototypes). In the first phase, to be developed within a window of six months, information about each company will be collected in Portuguese, using a web application at the end of the first phase. A UNL specialist will manually transform this content to UNL [6]. As the information becomes represented in UNL, the translation to the visitor's natural language will be conducted automatically and transparently, depending only on the language supported by the platform.

The second phase of the project will have a planned duration of 18 months and will allow the companies to describe their activities, services, and products using free text, instead of the limited words and text used in the first phase. In this phase, companies will be allowed to introduce more accurate and extended information.

The third phase will take 36 months, for a total project time of 60 months. The goal for the end of this phase is for companies to be able to interact with their customers using their native language in real time within the scope of the business transaction. This feature depends exclusively on UNL development, at least for the six intended languages.

3 Development Status

The first and present phase began to be developed in August of 2003 and is expected to be concluded in March 2004. All systems involved in the development are open source, as is the platform [7].?? The platform was planned to be operational system and hardware independent. Its requirements were identified through interviews and forms analysis of pre-compiled data from CELTA company websites. These interviews were accomplished in a sample of the universe of companies that considered various technological specializations.

The technical solution for the interaction between the visitor and the company in the native language of each one is limited in this first phase to the use of predefined forms based on UNL that limit the questions and answers. This technical restriction will not affect the ability to exchange the most important information on each company’s field. This communication solution comprises one of the initial requirements of the platform.

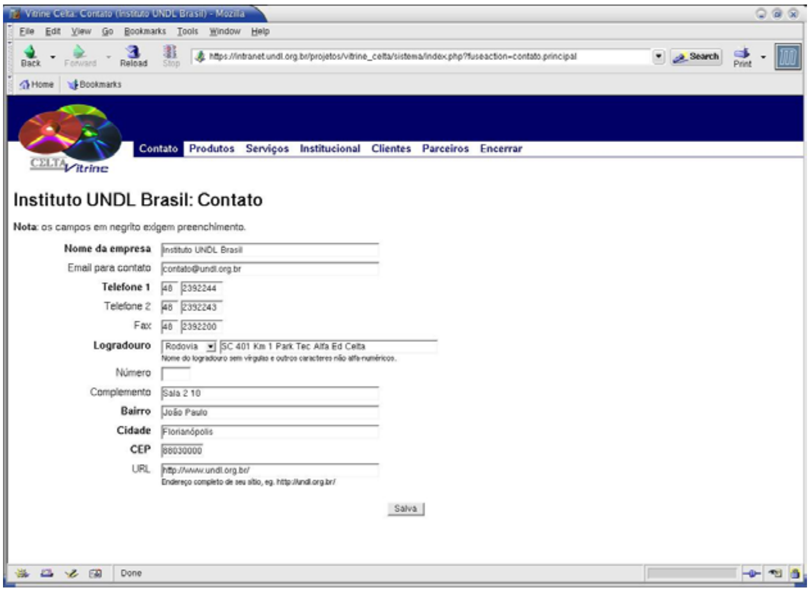
After the implantation of the system at the end of the first phase, feedback from the companies will provide the information needed to broaden the inclusion range of this communication tool, including more users' needs. At the end of the project, it is expected that all needs will be covered. In the following phases of the project, this communication tool gradually will begin to support free real-time conversation within a limited scope.

The first phase of the project was distributed in the following steps:

1. Requirements analysis and data acquisition

In this already completed step, a questionnaire targeting the companies was prepared with generic questions. It was distributed to five companies as a way of identifying common institutional information, the characteristics of products and services, as well as terms and concepts commonly used in customer contacts. This questionnaire was based on the first set of interviews and website analysis.

After the characteristics were identified, construction began of the *company module*. The implementation was accomplished through a web form (Figure 2) allowing the CELTA companies to register their information on the platform’s database. The module was developed using PHP [8] and the database was generated using MySQL [9]. The objective of this step was to collect the content for conversion in UWs and UNL sentences. This content will be part of the corpus of the *visitor's module*.


Vitrine Celta: Contato (Instituto UNDL Brasil) - Mozilla
File Edit View Go Bookmarks Tools Window Help
Back
Forward
Reload
Stop
https://intranet.undl.org.br/projetos/vitrine_celta/sistema/index.php?fuseaction=contato.principal
Search
Print
Home
Bookmarks
CELTA Vitrine
Contato
Produtos
Serviços
Institucional
Clientes
Parceiros
Encerrar
Salva
Done

Instituto UNDL Brasil: Contato

Nota: os campos em negrito exigem preenchimento.

Nome da empresa Instituto UNDL Brasil
Email para contato contato@undl.org.br
Telefone 1 48 2392244
Telefone 2 48 2392243
Fax 48 2392200
Logradouro Rodovia SC 401 Km 1 Park Tec Alfa Ed Celta
Nome do logradouro sem vírgulas e outros caracteres não alfa-numéricos.
Número
Complemento Sala 2 10
Bairro João Paulo
Cidade Florianópolis
CEP 88030000
URL http://www.undl.org.br/
Endereço completo de seu sítio, eg. http://undl.org.br/

Fig. 2. Company Module – web form

A support tool for dealing with UWs and UNL sentences was developed by *Instituto UNDL Brasil*. This tool is helping UNL specialists to perform the *enconversion* [6] activity, transforming the content in Portuguese to UNL. This support tool allows the specialist to browse all content entered by the companies. It marks content already in UNL, and presents content to be *enconverted* [6]. This tool is necessary only during the time that a *UNL Encoverter* for Portuguese is not available. It is expected that this tool will be removed at the end of the project, in the third phase.

2. Visitor's module development

To create the visitor's module, two items should be developed: software structure and content.

The structure for the visitor's view contents was developed using PHP in order to present the content in the visitors' native language. Using one of PHP's capabilities - the generation of dynamic content for web navigators - the visitor's view was created, reserving unique spaces for each company.

When the visitor accesses the website of the CELTA's Showcase platform, the visitor's web navigator will visualize the content in the visitor's language. If the language set on the web navigator is not already supported by the platform, a message is displayed requesting the selection of one of the supported languages. This message is displayed in all the supported languages of the platform at the same time. After the selection, all the content is displayed using the target language.

The accomplishment of the content display in different languages is possible due to the visitor's module. The visitor's module will access UNL System to present the dynamic generated pages. This is possible since all content is stored in a database using UNL.

The content, gathered in the last step will be converted to UNL using the tool developed in the first step. The other activities of this step have already been accomplished. This step is still being developed mainly due to the *enconversion* of the content to UNL.

3. Development of the communication tool

In this third step, a communication tool will be developed. Two classes of users, the visitor, and the company will use the communication tool. The implementation will be quite similar in both cases.

At the *visitor's module*, the communication tool will offer the visitor the opportunity to exchange information with CELTA companies using the visitor's native language. In order to make this possible, interactive forms will be used that allow limited communication among the parties in this phase. With these forms, the visitor can compose different questions and answers that will be automatically converted to UNL. These forms are restricted to a set of questions and answers. This set of questions and answers will be preprocessed for both customer and company.

The communication tool will convert the questions from the visitor's language to UNL automatically. This solution eliminates the need to use ENCO. This is only possible due to the simplicity and the restrictions at this phase.

After the UNL sentences are composed, the questions will be sent to the addressee, the CELTA company's responsible staff person, who will receive the message in his or her, native language, Portuguese, after the UNL System decodes the sentences.

This message will then be presented at the platform company's module. This application also will be developed using PHP. It will allow the company representative to answer the questions. This application will be built in a way similar to the one that the visitor used. The answer given by the company's representative will be sent to the visitor's e-mail in his or her native language, with a link attached to the platform allowing further conversation.

At this phase, the mapping from the language of the language set, including Portuguese, to UNL will be accomplished using the same methodology presented before to create the UNL content for the platform. The words and sentences used will be previously chosen and the same *enconversion* method will be applied to convert them to UNL. If necessary, some concepts will be generated as temporary UWs.

4. UNL System link

For the immediate operation of the platform, the current version of the UNL System [10] will be utilized. However, the platform will be loosely coupled to the UNL System, allowing future modifications of both the UNL System and the UNL without interfering in the platform directly.

The dynamically generated content for the platform (by PHP) will be sent to the UNL System Interface, UNL Web Service, before being returned to the visitor. This interface was planned as a web service [11] responsible for connecting the platform with the UNL System. The UNL Web Service mainly will be responsible for the following operations: processing the UNL requests, and selecting the most appropriate server for the required conversion. All these processes are transparent to the visitors and companies.

4 How does it Work?

There are two possible scenarios for accessing the platform as a visitor. Both scenarios integrate the same path up to a point where a division occurs. The visitor accesses CELTA's Showcase website through his or her web navigator where one can find information about the participating companies.

The content of the website is stored in UNL and in native languages already cached in previous access.

In the first scenario, the visitor's module will prepare the pages in the native language and simply display them to the visitor.

In the second scenario (figure 3), the visitor's module will prepare the pages based on the answer that the UNL System gives to the platform. If the user requests the information and it is not yet cached on the platform, the visitor's module sends a request to the UNL Web Service. This request contains the UNL sentences or UWs and the intended target native language.

The UNL Web Service is responsible for handling the data exchanged between the platform and the UNL System. It has a dynamic table that helps locate the intended native language UNL Language Server. It will communicate with the platform in a standard protocol for web service. The interface with the UNL System follows the UNL Language Server protocol.

The UNL System converts the content using the DECO and the linguistic resources that correspond to the native language requested, such as dictionary and grammar.

After it receives the native language content corresponding to the requested UNL content from the UNL System (UNL Language Server), the UNL Web Service returns it to the platform.

As the content arrives, it is stored on the corresponding language cache allowing the completion of the page construction to be presented to the visitor, as a web page.

This whole process will be fast, even though it is complex, because the information once transformed to the visitor's natural language will be stored locally at the platform's database language cache.

5 Final Considerations

This platform could be based entirely on translations performed by humans and all of the companies' content could be converted to the five other languages, providing them with the type of solution that big companies use. But these services would be very expensive for incubated companies and would represent a large share of their monthly expenses.

The implementation of this platform adopted an incremental method of using UNL resources to avoid interference from a technology that is not yet mature. In order to isolate the platform from problems related to UNL development, an interface will be provided by UNL Web Service.

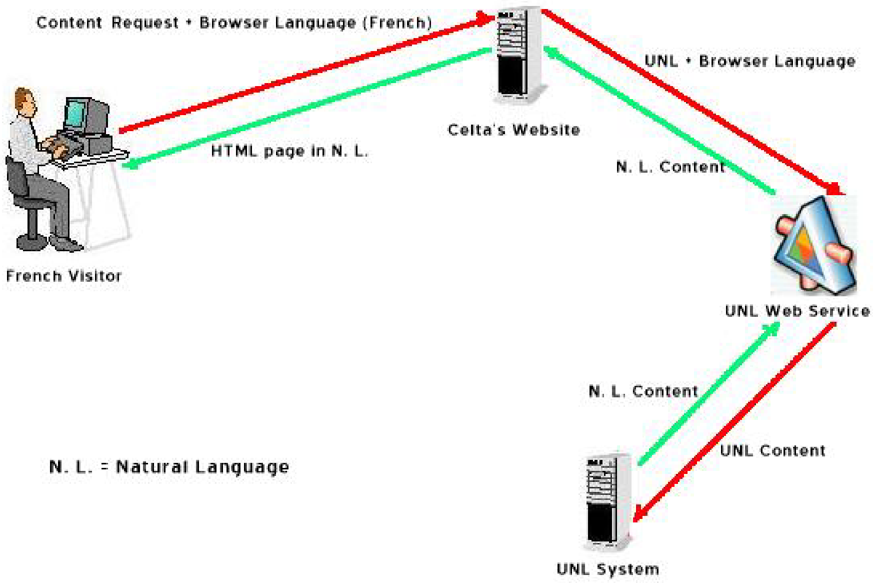


Fig. 3. System works

In addition to isolating the proposed platform, the creation of the UNL Web service in the future will allow any application to have access to a MT system and to knowledge management tools such as UNL KB and UNL Encyclopedia.

It is likely that the UNL Web Service, responsible for communication between the platform and the UNL System, can contribute in some way to the enhancement of the UNL System.

One of the main advantages of using UNL, instead of conventional translation, is the ability to write the content just once, which will then be replicated automatically to several languages through the UNL System. This advantage reduces deviation when updating content and reduces translation costs.

Another important advantage of using a platform like the one proposed here is that the companies can exchange simple information with customers, partners and foreign suppliers, with each one writing in their native languages, without a need for human translators'. This will already be possible at the end of the first development phase.

It is expected that even at the end of the first phase the platform will begin to achieve the planned goals and will increase the efficiency in communication among the companies at CELTA and their customers, partners, suppliers, and investors. It is also expected that this application will broaden the possibilities for using UNL, especially those related to commercial applications, and based on this platform, similar projects and products can be developed. The users of such a platform will not have any contact with UNL after its completion.

As the second development phase begins, the developers will receive feedback from the users. This feedback will begin after the results of the first phase are achieved. The continuation of the phases, will allow users to enjoy the capabilities of the UNL System to a larger degree.

Acknowledgements The authors would like to thank the Instituto UNDL Brasil, Fundação de Ciência e Tecnologia do Estado de Santa Catarina, and Universidade Federal de Santa Catarina for their support

References

1. Instituto UNDL Brasil. Conferência Sobre Tecnologia UNL. Available at: <http://www.undl.org.br/>. Accessed on: 15 oct.2003.
2. CELTA. CELTA On Line. Available at: <http://www.celta.org.br/>. Accessed on: 15 oct.2003.
3. Castiñeira, M. Inés;Mülbert, Ana Luisa; Schuhmacher, Vera; Madeira, Mauro. "Projeto Pedagógico em Ciência da Computação: Como Atender a Diversidade Regional?", Weimig2003, II Workshop de Educação em Computação e Informática, Poços de Caldas, May, 2003, 9 pp.
4. RUMBAUGH, J. Object-Oriented Modeling and Design. NJ: Prentice Hall, 1991. 500 pp.
5. RUMBAUGH, J.; JACOBSON, I.; BOOCH, G. The Unified Modeling Language Reference Manual. Reading, MA, USA: Addison-Wesley, 1999. 550 pp.
6. Uchida, Hiroshi; Zhu, Meiying; and Della Senta, Tarcisio. "The UNL, a Gift for a Millennium", Institute of Advanced Studies, United Nations University, 1999, 64 pp
7. Open Source Initiative. "Frequently Asked Questions". Available at: < http://www.opensource.org/>. Accessed on: 15 oct.2003.
8. PHP project. "What is PHP?". Available at: <http://www.php.net/>. Accessed on: 15 oct.2003.
9. MySQL. "MySQL Reference Manual". Available on: www.mysql.com/doc/en/What-is. html. Accessed on: 15 oct.2003.
10. Uchida, Hiroshi; Zhu, Meiying. "The Universal Networking Language beyond Machine Translation", International Symposium on Language in Cyberspace, Seoul, Sept 2001, 13 pp.
11. World Wide Web Consortium. "Web Services". Available on www.w3.org/2002/ws/. Accessed on: 15 oct.2003.

Studies of Emotional Expressions in Oral Dialogues: towards an Extension of Universal Networking Language

Mutsuko Tomokiyo,[1] Gérard Chollet,[2] Solange Hollard [3]

[1] GETA-CLIPS-IMAG & ENST
BP 53 38041 Grenoble cedex 9 France
mutsuko.tomokiyo@imag.fr ; tomokiyo@tsi.enst.fr
[2] ENST
46 rue Barrault, 75634, Paris
chollet@tsi.enst.fr
[3] GEOD-CLIPS-IMAG
BP 53 38041 Grenoble cedex 9 France
Solange.Hollard@imag.fr

Abstract. Emotions entail distinctive ways of perceiving and assessing situations, processing information, and prioritizing and modulating actions [24]. The paper aims to study theoretical and pragmatic aspects of emotions and to propose a semantic representation of emotions for oral dialogues, based on an analysis of real-life conversations, telephone messages and recorded TV programmes, focusing on a relationship between prosody and lexeme for the purposes of a speech to speech machine translation. The semantic representation is made, by using the **U**niversal **N**etworking **L**anguage (UNL) formalism, in a way where lexeme, phatics, gestures, prosody and voice tone are taken into account at the same time.

1 Introduction

This work has been carried out in a continuation of “VoiceUNL” [21], which is one of subprojects of the “LingTour” [1] project. “VoiceUNL” is an extension of **U**niversal **N**etworking **L**anguage (UNL), which is a text-oriented formalism of semantic graphs,, to oral dialogues.

As for **s**peech to **s**peech **m**achine **t**ranslations (SSMT) or man-machine interactive systems, the detection and generation of emotions are an important issue from the viewpoint of the naturalness of dialogues [7], because *emotion entails distinctive ways of perceiving and assessing situations, processing information, and prioritizing and modulating actions* [24]. It's the key reason for proposing a semantic representation of emotions.

In this paper, section 2 is devoted to previous emotion studies mainly focussed on prosody: a survey of existing approaches to emotion detection and generation, theo-

[1] The Lingtour project was launched in 2002 by the partnership which consists of TsingHua University (China), Paris 8 University (France), INT (France), ENST-Paris and Bretagne (France), and CLIPS (France). One of the objectives of the projects resides in R & D to enable multilingual-multimedia MT on user-friendly tools [1].

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 286–299.

retical approaches to emotions, emotion definition, and emotion categories. In section 3, we investigate our corpus, detect emotion categories and emotion eliciting factors, and extract emotional expressions from it. In section 4, after having introduced UNL briefly, we propose a semantic representation of emotions within the UNL formalism, by adding tags representing speech dialogue properties to UNL to suit it to SSMT.

2 Previous Studies into Emotions

2.1 Existing Approaches to Recognition and Generation of Emotions

Much work has been carried out in detection and identification of emotions in written texts or oral dialogues for various applications. Existing approaches are grouped into three types:

- observation of paralinguistic elements such as prosody, facial and body movements in spoken languages,
- detecting lexical items expressing emotions by using a shallow word match parser or by physiologic manual evaluation and defining emotions by the distance between two emotions according to the distance values given [7][44], or
- discovering syntactic and lexical patterns in the text that allow emotion tagging [6].

Our approach employs a method where spoken language properties such as lexeme, gestures, prosody, etc. are recognized, translated and generated, since one objective of our emotion representation is SSMT using the UNL framework.

In fact, in order to determine the type of emotion, these elements are taken into account at the same time, because the same variable can express different classes of emotions. For example, an increase of elocution speed or the rising tone can indicate happiness as well as anger [22] [36].

2.2 Theoretical Approaches to Emotions

Cornelius mentioned there would be *four of the most influential theoretical perspectives and research traditions in the study of emotion in the past 125 years or so* [28] without citing Greek philosophers like Plato and Aristotle, nor French ones like Descartes. These four perspectives are the Darwinian , Jamesian [30], Cognitive , and Social constructive perspectives.

In the Darwinian perspective, emotions are considered fundamental because they represent survival-related patterns of responses to events in the world that have been selected in the course of our evolutionary history.

James [30] says that *the bodily changes follow directly the PERCEPTION of the exciting fact, and that our feeling of the same changes as they occur IS the emotion.*

In the experimentations of Levenson and al. [29], subjects received muscle-by-muscle instructions and coaching to produce facial configurations for anger, disgust, fear, happiness, sadness, and surprise while heart rate, skin conductance, finger temperature, and somatic activity were monitored. Results indicated that *voluntary facial activity produced significant levels of subjective experience of the associated emotion, and that autonomic distinctions among emotions were found both between negative and positive emotions and among negative emotions*.

Consequently, the Jamesian group considers emotions as the expressive process of affective programmes which activate different subsystems.

Cornelieu pointed out that there is a considerable crossover between the Darwinian and Jamesian traditions in psychology. Its main point is that the bodily responses are associated with emotions. Thus, his suggestion [28] being based on the Levenson's experiments enables us to assume that the prosody can be also considered as one of these activated systems by one of affective programmes.

In the Cognitive perspective, as mentioned by Arnold [31], *thought and emotion are not separable, and all emotions are seen… as being dependent on the process by which events in the environment are judged as good or bad for us* [28]. Hence, every emotion is associated with a specific and different pattern of judgments of the worth, value, or condition of something, called appraisal.

The Social constructivists like Haviland [32] or Averill [33] assume emotions are cultural products that owe their meaning and coherence to learned social rules. For instance, anger plays a positive and constructive role in our social relationship, because, on the one hand, anger is generated only when one is intentionally wronged, and on the other hand, anger depends upon culture one belongs to.

Randall [8] states that most cultures have emotions and emotional vocabularies that have two components: a universal element, and a component or parameter that is peculiar to the beliefs and values of that culture. Hence, for the social constructivists, culture plays central role in the organization of emotions.

We could regard emotions a processes, consisting of several components from these perspectives: physiological, cognitive, sociomotivational and 'action tendency' as Scherer mentioned [40].

2.3 Definition and Emotion Category

Randall [8] defines emotion as a feeling that has been caused by certain beliefs, directed toward a primarily conceptual and non-perceptual target that typically produces some physiological, behavioural, or cognitive effect.
Cowie [27] mentions there are two different senses for the word 'emotion':

- The first sense uses the word, in plural form, *to refer to entities – natural units that have distinct boundaries, and that can be counted*.
- The second sense uses the word to *refer to an attribute of certain states. That is the sense that is involved when we say that somebody's voice is tinged with emotion.*

Our main concern to 'emotion' is one in the first sense, so our primary task is to obtain a list of emotion categories.

How many and what kind of emotional states are expressed in general dialogues?

Ekman [11] mentioned that there would be a linking of a second emotion with an initial emotion, and emotions rarely occur simply or in pure form. There is, however, quite general agreement on the so-called 'big six' as the initial emotion: fear, anger, happiness, sadness, surprise and disgust [28].

In OCC [9], there is an assumption that there are three major aspects of the world, namely, events, agents, or objects. Emotions *are valenced reactions, and any particular valenced reaction is always a reaction to one of these perspectives on the world.* Emotion types in OCC include 'happy' for resentment, gloating 'pity', 'hope', 'fear', 'joy', 'distress', 'pride', 'shame', 'admiration', 'reproach', 'love', 'hate', 'gratification, 'remorse', 'gratitude', and 'anger'.

Plutchik [10] believes that emotions are like colours. Every colour of the spectrum can be produced by mixing the primary colours. His "emotion's wheel" consists of eight primary emotions: fear, surprise, sadness, disgust, anger, anticipation, joy, and acceptance, and he lists 142 categories as second order emotion.

The emotion study group in Southern Kings Consolidated school [20] classifies the emotions as follows: Thankfulness, Envy, Disgust, Worry, Kindheartedness, Stress, Boredom, Sadness, Loneliness, Bravery, Paranoia, Optimism, Stubbornness, Fear, Anxiety.

Morita [41] classified Japanese emotional words into 40 categories, and it contains negative or positive judgment, sense to color or sound, psychological reactions, etc.

We extract and classify emotions by investigating our corpus.

3 Dialogue Corpus Analysais

3.1 Corpus

After surveying the theoretical aspect of emotions from the literature and available research papers on emotions and speech, we have developed in our first approach to emotions a corpus, which contains:

a. 30 minutes of English instruction programmes on TV,
b. a 40-minute French TV interview [5],
c. 5 hours of real-life vocal messages left on a telephone answering machine, sent from medical stuff to a group of computer engineers in a French public hospital [13],
d. 1 hour of real-life telephone conversations between administration staff of a French university [12] and
e. 6 basic conversations on transport in English, French, Japanese and Chinese [25].

We mainly used a. c. d. in the corpus.

3.2 Emotion Categories

We analyzed the corpus, either to fix emotion categories, or to find emotion eliciting factors.

We previously prepared a working sheet[2] to transcribe recorded material while listening to them or watching them, and checked the lexemes concerning emotional feeling, emotional prosody, emotional gestures, etc.

For emotion categories, we found the following categories in our corpus including the big-six we mentioned above: happiness, sadness/disappointment, disgust, surprise, fear, anger, irritation, hesitation/uncertainty, anxiety, and neutral.

These category names are used later to describe emotion states of the speaker as well as to annotate lexemes having emotional content in a semantic representation of emotions.

3.3 Emotion Eliciting Factors

For emotion eliciting factors, the followings are the major ones in our corpus :

- lexemes (sad, happy, etc.)
- phatics (ah, hein, etc.)
- prosodic cues (fast, slow, strong, etc.)
- voice (noisy, soft, etc.)
- gestures (movement of hands, mouth, eyes, etc.)

As an example, “No!” in the example

Victor - “May I smoke?”
Victor’s father - “No! you may not, Victor” [5]

expresses Victor’s father’s surprise, because Victor is a small boy.

So, the surprise can be represented by the lexeme “No!”. However, at the same time, on TV, the father also made a grimace while saying “No!”. Thus, surprise can also be expressed by the movement of the eyebrows and the voice tone.

From the example, we can suppose potential emotion expressions: ‘happiness’ is expressed by lexemes, phatics, prosody, voice, hand movements, mouth and/or eyes ; ‘sadness/disappointment’ is done by lexicon, phatics, prosody, voice, mouth and/or eyes ; ‘disgust’ is expressed by lexemes, phatics, prosody, voice, mouth, eyes, eyebrows and/or shoulder movements ; ‘surprise’ is expressed by lexemes, phatics, prosody, voice, mouth, eyes and/or head, and so on.

3.4 Lexicon and Prosody with regard to Emotions

3.4.1 Prosodic Levels for Emotions

Much research has been conducted on prosodic characteristics in utterances according to each emotion category. For example:

[2] The Table 2 comes from the working sheet simplified.

- Yamazaki [34] reports French subjects associate ‘positive’ emotions with F0 raising contour and the ‘negative’ emotions with F0 falling contour from her experiments of perceptual aspects of ‘positive’ or ‘negative’ emotions for synthetic stimuli.
- Halliday [35] points out a correlation between sentence types and prosodic manner, and claims a wh-question with a rising tone is ‘tentative’, while a yes/no question with a falling tone is ‘peremptory’.
- Wichmann [36] mentions *High and Low* contour of please-request sentences can express a request of greater urgency.
- e.g. (High * Low) Please open the (Low*High)door.
- She concludes that *affective meanings are conveyed not only by continuously variable phonetic parameters, nor only through iconic associations with relative pitch height, but also by the conjunction between categorical choices of contour and utterance type*.
- Cælen-Haumont [37] defined ‘melism’ as a notion characterizing acoustic modifications of F0 (a great amplitude of F0), related to the expression of a linguistic meaning in affective conditions of speaking, and developed two models which she integrated in the Praat speech analysis software to measure the ‘melism’ [38].
- She confirmed by testing the models that the application of ‘melism’ to semantic or pragmatic analysis of utterances is possible.
- Aubergé shows that *prosody is one of the medium to express emotions in speech, through an in voluntary control* [39] by measuring acoustic parameters for strategic smile and spontaneous smile.
- Amir and al. extract an acoustic feature set of 12 elements from their corpus evaluation, which enables to classify emotional contents in speech: pitch and intensity statistics [44].

Thus we can confirm that information of prosody in utterances is indispensable for automatic recognition of emotions as well as lexemes and utterance types.

3.4.2 Manual Annotation for Emotions

The subjects of telephone conversations recorded at a French university are room reservation, schedule arrangement, taking a message, order of office supplies, etc., and some chats also are contained.

In the telephone messages at a public hospital, callers complain about problems with their computers or the software they use, and ask for technical help from an engineer, or ask for a rapid validation of an electronic access card for newcomers. In this con- text, a typical lexicon, or set of phrases expressing irritation, uncertainty or hesitation appear in the messages: *pénible, très pénible, drôlement embêté, une catastrophe, désespéré, relativement énervant, Ça me dérange beaucoup, ceci est assez désespérant, c’est embêtant,* etc. There are also "*C’est vraiment très urgent, Pourriez-vous venir voir?, Si vous pouviez passer rapidement"* etc. as more context-dependent examples.

In Table 1, we illustrate expressions of different emotions used in the telephone messages and the conversations. The first column shows emotion types, and the second and third column contain cited examples for the indicated emotion type.

Table 1. Lexemes having emotional content.

emotions	lexicon in the telephone messages	lexemes in the telephone conv.
happiness	merci beaucoup bonne journée,	Ah chouette!, Ouais!, impeccable!, Y a pas de souci!, C'est gentil. Merci!
disgust	nous avons un ordinateur qui est foutu où on travaille beaucoup,	
fear	J'ai peur que, je crains que,	J'ai peur que, je crains que
irritation	C'est embêtant, on est drôlement embêtés, ça me dérange, ça pose un réel problème, c'est relativement énervant, C'est très pénible, ma carte professionnelle de santé ne fonctionne plus, enfin, ça me marque 'illisible' ah !,	Ah non!
hesitation uncertainty	Je ne sais que faire, comment faire, nous aimerions savoir, je ne sais plus quoi faire. Euh, mon outlook ne s'ouvre plus, euh, et auparavant, il y a un message qui dit qu'il y a un problème avec le lecteur, on m'a dit qu'il y avait un problème sur... c'est que c'était plein, mon dossier était plein ?	Voyons voir, attends… je regarde, ben ben…, attends voir; heuh heuh, heueueueuh, Bof bof bof, hum hum, je ne sais pas, je vois pas bien, ça va faire un peu juste; on sait pas
sadness/ disappointment	Ici infirmière en état désespéré, j'ai mon poste qui est bloqué, on arrive pas non plus à arrêter, et ça c'est depuis hier, quoi,	Oh la pauvre! ah mince, ah zut, c'est dommage
surprise	Nous avons un écran noir ! nous pouvons ni voir les résultats, ni faire les mouvements à ce jour !	Oh, ça alors, ah bon?, tu crois?
anger	Les portables ne fonctionnent plus, nous sommes bloqués !, C'est urgent pour nous, alors je pense que ce soir, c'est plus envisageable, mais demain il faut impérativement, demain matin que ce problème doit être réglé.	Y en a marre!, C'est pas vrai!,
anxiety	C'est une catastrophe,	ça m'ennuie un peu, nous sommes ennuyés, c'est ennuyeux, y a une boulette, y a un souci, y a un truc qui me chiffonne,

Emotion eliciting words or phatics for each emotion class are surely found, but the prosodic features for each emotion class are divergent, as shown in Table 2., whereas there are clear prosodic signs which are confined to only one word which is semantically less significant. For example, in the utterance *"Il faudrait impérative-*

ment résoudre ce problème ce matin.”, only ‘*matin*’ is heavily accented, all of other words are uttered in a neutral tone, and we can interpret this accentuation as an implicit insistence on an urgent intervention.

We also have verified prosodic characteristics for some lexemes and set of phrases in the messages on Praat [14], but further examinations should be made to study the variety of prosodic features for each emotion class.

For instance, “*c’est relativement énervant*” or “*c’est très pénible.*” is uttered either with a neutral tone or at a raising tone.

Table 2. Lexical units and their prosody in Corpus Hotline CHRU

items	lexical units	emotional state	examples
same lexical unit with different prosodies	urgent	emphasized	on a un petit problème urgent
	extrême urgence	neutral	il nous le faut d’extrême urgence
	ce matin	emphasized	Il faudrait impérativement résoudre ce problème ce matin
		neutral	parce que ma collègue déjà appelé ce matin
lexemes having emotional contents	au secours, au secours	neutral	au secours, au secours, il faut absolument que je travaille sur cet ordi…
	c’est infernal	neutral	on passe des heures à connecter déconnecter l’ordinateur […] pour travailler, c’est infernal.
	énervant	neutral	c’est relativement énervant
	désagréable	neutral	ceci est assez désagréable, toutes les semaines
	urgent devoir vouloir	emphasized neutral neutral	c’est urgent ! vous deviez passer, mais nous avons pas de nouvelles de votre part. Veuillez me rappeler le plus tôt possible
utterance without any lexeme having emotional content	Il y a 9 étiquettes sur une et 16 étiquettes sur l’autre	irritated	L’imprimante nous imprime les étiquettes sur deux feuilles, qui sont toutes les deux incomplètes. Il y a 9 étiquettes sur une et 16 étiquettes sur l’autre
lexemes not having emotional contents	connecter and lundi matin	emphasized	Je n’arrive absolument plus à me connecter […] Est-ce que vous pourriez intervenir lundi matin?
	toutes les semaines	insistent	ceci est assez désagréable, toutes les semaines
	écran noir	emphasized	Nous avons un écran noir ! nous pouvons ni voir les résultats, ni faire les mouvements à ce jour !
anglicism	out	emphasized	Notre ordinateur est “out”, et deviez passer, mais nous avons pas de nouvelles de votre part.

These phenomena are parameterized as emotion eliciting factors in the structures of attributes and its values in the emotion representation.

Choice of lexemes and prosody are a bench mark for detecting and identifying emotions as mentioned above. So, it's useful to mark lexemes with some labels in a dictionary used just like restricted UWs in UNL.

We propose, due to this fact, a set of emotion labels composed of 9 classes excluding "neutral" in our emotion classes and annotate lexemes in a UNL manner.

e.g.
désagréable (icl>sentiment>disgust)
catastrophe(icl>sentiment>anxiety)

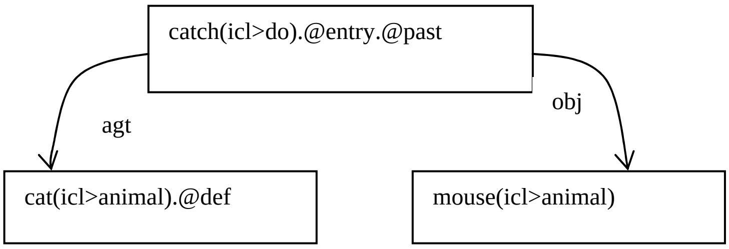
4 Semantic Representation of Emotions

4.1 UNL

A UNL graph consists of “UWs”, “Relations”, “Attributes”. It can be represented using tags. The “UWs” form the vocabulary of the UNL language, and denote "interlingual acceptions" (word senses). “Relations” and “Attributes” mainly make up the syntax, and the "knowledge base" (KB) covers the semantics of UNL [4]: it is a network containing all UWs, with all possible binary "semantic" and "thesaurus" relations between them. The 41 semantic relations contain "volitive agent, "coagent", "deep object", "instrument", "place", "place to", "time", "reason", "scene", "apposition", etc. Thesaurus relations contain "part of", "synonym", "is a", "field", and "antonym".

Here is an example of UNL graph, with one of its linear writings.

e.g. The cat caught a mouse.
[S] agt(catch(icl>do).@entry.@past, cat(icl>animal).@def)
obj(catch(icl>do).@entry.@past, mouse(icl>animal)) [/S]



An UW is made up of a character string followed by a list of constraints. UWs include basic UWs (bare English words), restricted UWs (English words with a constraint list), and extra UWs, which are a special type of restricted UWs [4].

One of the main advantages of UNL is the Universal Word (UW) dictionary, which enables us to specify word meanings at a deep level and to perform lexical disambiguation in a semantic oriented formalism.

In the example, “icl” in the constraint list enables us to define a subconcept of a basic UW. We will also apply later this constraint way to lexemes having emotional content for the purposes of representing emotions.

“agt” and “obj” are Relation tags, which indicate dependency relations between a head word in linguistic categories and other words, based on a case grammar type specification. “.@entry”, “.@past” and “.@def” are called Attribute tags, which indicate the grammatical conditions of a given utterance. The graph in the example does not contain any tags apart from the UNL tags, so they will be merged with other tags added as well as embedded in another format for the purposes of SSMT.

4.2 Semantic Representation of Emotions within UNL

The UNL semantic representation for written texts is actually designed by a set of 111 tags, which are divided into the 41 Relation tags and the 70 Attribute tags [4]. As for SSMT, some tags covering spoken language properties are merged with the UNL tag set: the added tags are 9 emotion tags we proposed, 8 prosody tags coming from the W3C recommendation [15], 12 behaviour tags from MPEG-4 [16] and, in particular, 28 speech act tags from a speech act research team [17] [18], and 5 interaction manner tags from GDA [19].

The UNL representation is a graph and consequently is not easy to encode in a linear data stream[3]. However, it is feasible to project it onto a description format such as XML, which authorizes the definition of elements and attributes. The representation obtained offers the same expressive power as graphs, but in the form of tags, and is easy to transmit. It is therefore easily interpreted by a DTD (Document Type Definition) conforming to the XML norm [26]. Thus, we attempted to transform UNL graphs into XML format as it facilitates speech synthesis information after the generation of the target language.

The representation schema of emotions proposed is made by adding tags expressing emotions according to the UNL. There are three ways to add such tags, that’s adding tags: “outside” of UNL makers as <VoiceUNL>, “inside” UNL text or a combination of both [21]. When emotions are formalized “inside” of the UNL makers, all tags representing prosody, behaviour and the speech act one are put in an UW. Therefore, in UNL graphs the arc concept representing a semantic relationship between two UWs might turn out to be unclear.

On the other hand, when emotions are formalized “outside” the UNL marker, in order to synchronize character’s strings and speech and visual items occurring simultaneously in an utterance, the same character’s string appears several times in a semantic representation. In such a dilemma, we create an additional UW type, which enables us to link speech, gesture, emotion and prosody tags: SP01, SP02, SP03...., and we use them in an “outside” and “combined” manner.

The following is a representation of an exchange in the “combined” manner:

[3] On Ariane-G5, which is an environment of MT into French language, UNL graphs are converted into tree structures [42].

```
  <?xml version="1.0" encoding="iso-8859-1 ?>
  <D dn=" TV " on="mt" dt="2003">
  <Paragraph number="1">
  <Sentence  snumber="2">
  <org lang="el"> No! you may not drink, Victor.[4]</org>
  <unlsem>
  agt:SP01(drink(icl>do).@entry.@present.@obligation-not,
you.@emphasis)
  mod:SP02(drink(icl>do).@entry.@present.@obligation-
not,no(icl>sentiment>surprise).@emphasis)
  mod:SP03(no(icl>sentiment>surprise).@emphasis,
!(icl>symbol>surprise).@surprised)
  mod:SP04(you,Victor(icl>name).@vocative)
  </unlsem>
  <VoiceUNL><speech-act>type=‶inform″    mod:SP01,    type=‶No″
agt:SP02 </speech-act>
  <interaction> ref=‶drink″ agt:SP01 </interaction>.
  <emotion> class=‶surprise″ mod:SP02 </emotion>
  <gesture>eyebrows=‶left-and-right-raised″ mod:SP02 </gesture>
  </VoiceUNL>
  </Sentence >
  </Paragraph>
  </D>
```

Note that "no" is annotated as "no(icl>sentiment>surprise)" by one of emotion class tags. It means that this "no" refers to a surprise as well as to a negation[5].

On the other hand, prosody tags (.@emphasis) are attached on UWs between <unlsem> and </unlsem>, and the gesture, emotion and discourse tags are external to <unlsem>, because only prosody is identified at the level of UWs, and the rest is often associated with utterance fragments or an entire utterance.

"drink", "you", "no", etc. are pivot languages called UW, and are converted into "boire", "tu", "non" respectively in the French generation module [2] [20][23]. Therefore, the transcription of this utterance is: “Non! tu ne peux pas boire, Victor”.

5 Conclusion

Our emotion studies have shown emotion analysis and generation were important issue in SSMT and our dialogue corpus analysis has suggested that lexemes are the most eleciting elements of emotions and that there is a delicate relationship between the lexeme uttered and its prosody. Thus we have proposed a semantic representation of emotions where all emotional expressions such as lexemes, prosody, gestures, etc. are described at the same time, by annotating lexemes with a set of labels, and adding speech property tags, speech act tags, interaction manner tags and behaviour tags to UNL in order to suit it to SSMT.

[4] The example cited here is: “May I drink?” “No!, you may not, Victor”. We have minimally changed it for convenience’s sake.

[5] Many previous studies have indicated that F0 raising contour is evoked by the happiness, surprise and anger in contrast to F0 falling contour which is evoked by the sadness or the uncertainty [3, 22]. This "no" is uttered in strong raising F0 contour on Praat.

We have found overlapping of utterances, irregular turn taking, category omission, deictic expressions, discourse ellipsis, etc. in our corpus [17]. Such interaction manners also are concerned with emotions of the speaker.

We actually use 5 tags from GDA[6] tag set [19] in the same way as paralinguistic tags to represent them as specificity of oral interaction manners. However further reflection is needed for discourse processing. For example, "VoiceXML" recommended by W3C [43] is designed for generating audio dialogues in monolingualism mainly on man-machine interactive system. A voice XML document is composed of top-level elements called "dialogs", and there are two types of "dialogs" : "forms" and "menus". "Forms" present information and gather input according to "Form Interpretation Algorithm", and "menus" offer choices of what to do next by referring one or more grammars associated with "dialogs". We also might need a mechanism which enables constantly to watch a flow of dialogues for coping with discourse ellipsis, anapholic expressions, etc. in oral dialogues.

The next step will be to develop a prototype with a speech and image interface as well as to enrich our corpus with speech and sound.

Acknowledgements The authors thank Professor Ch.Boitet, head of GETA in the CLIPS lab, for many inspiring discussions and the opportunity to conduct this research in the framework of the "UNL" project.

References

1 Chollet G., *Lingtour, Project Brochure*, GET, Paris, 2003.

2 Blanc E., GETA, CLIPS, IMAG, French deconverter, 2000.

3 Scherer, K. R., *Psychological models of emotion*, in J. Borod, *The neuropsychology of emotion*, Oxford/NewYork University Press, 2000.

4 UNL Center and UNL Foundation, *The Universal Networking Language (UNL) Specification*, Version 3, edition 1, UNU, IAS, Tokyo, Japan, 2002.

5 TV programmes on the 5th of October, 'Victor', Arte, 2003.

6 Holzman Lars E. and Pottenger William M., *Classification of Emotions in Internet Chat: An Application of Machine Learning Using Speech Phonemes*, 2003.

7 Fitrianie S., Wiggers P., and Leon J.M.,Rothkrantz L., *A Multi-Modal Using Natural Language Processing and Emotion Recognition*, Text, Speech and Dialogues, 6th International conference, TDS 2003, Czech Republic, Proceedings, LNAI 2807, V. Matousek and P.Mautner (Eds), Springer, 2003.

8 Randall, R.D., *The nature and Structure of Emotions*, U.S. Military Academy, USA, 1998, http://neologic.net/rd/Papers/EM-DEF19.html

9 Ortony A., Clore G. L, Collins A, *The Cognitive Structure of Emotions*, Cambridge, 1990.

[6] The Global Document Annotation (GDA) Initiative research team has proposed (2001)a XML-based tag set to help computing machines automatically infer the underlying semantic/pragmatic structure of documents. The GDA tag set is designed so that the GDA-annotation reduces the ambiguity in mapping a document to a sort of entity-relation graph (or semantic network) representing the underlying semantic structure [19]. There is a mapping schema between UNL specification tags and GDA one.

10 Plutchik R., The nature of emotions, American Scientist 89 344, USA, 2001.

11 Ekman P., *Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life*, Times Book, USA, 2003.

12 Corpus DialAdmin, document CLIPS, Grenoble, 2003.

13 Corpus Hotline CHRU, document CLIPS, Grenoble, 2003.

14 Boersma P. and Weenink D., Praat, *Doing Phonetics by Computer, (version 4.1),* Institute of Phonetic Sciences, University of Amsterdam, http://www.fon.hum.uva.nl/praat/, 2003.

15 W3C, *Speech Synthesis Markup Language Version 1.0*, W3C Working Draft 02, 2002, http://w3.org/TR/2002/WD-speech-synthesis-20021202, W3C, *Semantic Interpretation for Speech Recognition*, W3C Working Draft 02, 2003, http://w3.org/TR/2003/WD-semantic-interpretation-20030401

16 Koenen P. R. (ed), MPEG-4 Overview, http://mpeg.telecomitalialab.-com/standards/mpeg-4.html, 1999.

17 Tomokiyo M., Analyse discursive de dialogues oraux en français, japonais et anglais, Septentrion, Lille, 2001

18 Seligman M., Tomokiyo M. and Fais L., *A Bilingual Set of Communicative Acts Labels for Spontaneous Dialogues*, Rap. ATR, TR-IT-161, Japan, 1996.

19 Hasida K., *The GDA Tag Set (version 0.68),* http://i-content.org/gda/tagman.html (version 0,71), 2003.

20 Southern Kings Consolidated School, *Types of emotions,* 2003 http://www.edu.pe.ca/southernkings/emotionstudi.html

21 Tomokiyo M. and Chollet G., *VoiceUNL: a proposal to represent speech control mechanisms within the Universal Networking Digital Language*, International Conference on the Convergence of Knowledge, Culture, Language and Information Technologies, Egypt, 2003.

22 Caelen-Haumont, G., BEL, B. *Subjectivité et émotion dans la prosodie de parole et du chat : espace, coordonnées et paramètres*. Colloque international « Emotions, Interactions & Développement », Grenoble, 2001. In Coletta, Jean-Marc; Tcherkassof, Anna (eds.) Perspectives actuelles sur les émotions. Cognition, langage et développement. Mardaga: Hayen. 2002.

23 Sérasset G., Boitet, Ch., *UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction,* MT-SUMMIT VII, Singapore, 1999

24 HUMAINE, *HUMAINE Technical Annex*, 2004, http://emotion-research.net/

25 Corpus TRANSPORT: *Basic conversations on transport in English, French, Japanese and Chinese,* CLIPS, 2004.

26 Tsai WJ., *UNL news*, http://www-clips.imag.fr/geta/User/wang-ju.tsai/showunl.html, 2003.

27 Cowie R., *Describing the emotional states expressed in speech,* Proceedings of the ISCA Workshop on Speech and Emotion, Belfast, Proceedings on line, http://www.qbc.ac.uk /en/isca/proceedings, 2000

28 Cornelius R.R, *Theoretical approach to emotion,* Proceedings of the ISCA Workshop on Speech and Emotion, Belfast, Proceedings on line, http://www.qbc.ac.uk /en/isca/proceedings, 2000

29 Levenson R.W., Ekman, P., Friesen, W.V., *Voluntary facial action generates emotion-specific automatic nervous system activity,* Psychophysiology, 27, (4) 363-384, 1990, http://www.ncbi.nlm.nih.gov/entrez/query.fcgi-?cmd=Retrieve&db=PubMed&list_uids=2236440&dopt=Abstract

30 James, W., What is an emotion?, Classics in the History of Psychology, an Internet resource developed by Green, Ch. D., 1884, http://psychclassics.-yorku.ca/James/emotion.htm

31 Arnold, M. B., *Emotion and personality,* vol1, Psychological aspects, New York, ed Columbia University Press, 1960.

32 Handbook of Emotions, Edited by Lewis, M., Haviland-Jones, J. M., Guiford, USA, 2004,

33 Averill, J.R., *A constructivist view of emotion,* Emotion : Theory, research and experience, vol.1, New York Academic Press, edited by Plutchik, R. and Kellerman, H., 1980.

34 Yamazaki H., *Perception des émotions "positive'' et ''négative'' chez les auditeurs français et japonais à travers le contour de F0,* Proceedings on Webs of JEP2004, 2004 http://aune.lpl.univ-aix.fr:16080/jep-taln04/proceed/actes/jep2004/Yamasaki.pdf

35 Haliday M.A.K., *An Introduction to Functional Grammar,* London, Edward Arnold, 1994.

36 Wichmann A., *Attitudinal Intonation and the Inferential Process,* International Conference Speech Prosody 2002, 2002 http://www.lpl.univ-aix.fr/projects/aix02/sp2002/pdf/wichmann.pdf

37 Cælen-Haumont G., Auran C., *The phonology of Melodic Prominence: the Structure of melisms,* In the "2nd International Conference on Speech Prosody Proceedings, SP 2004, http://www.isca-speech.org/archive/sp2004/sp04_143.pdf

38 Cælen-Haumont G., *Valeurs pragmatiques de la proéminence prosodique lexicale : de l'outil vers l'analyse,* Proceedings on Webs of JEP2004, 2004, http://aune.lpl.univ-aix.fr:16080/jep-taln04/proceed/actes/jep2004/Cælen-Haumont.pdf

39 Aubergé V., *The Prosody of Smile,* ISCA Workshop on Speech and Emotion ; A conceptual framewor for research, Proceedings on line, 2000 http://www.qub.ac.uk/en/isca/proceedings

40 Scherer, K. R., *Toward a concept of "modal emotions".* In P. Ekman & R. J. Davidson (Eds.), The nature of emotion: Fundamental questions (pp. 25-31). New York/Oxford: Oxford University Press, 1994.

41 Morita Y., *Japanese dictionary – adjectives and adverbs,* Koudansya, Japan, 1989.

42 Blanc E., *From the UNL hypergraph to GETA's multilevel tree.* Proceeding of Machine Translation, University of Exeter, 18-21 oct., pp9.1—9.9. Ed. British Comp. Society, 2000.

43 *Voice Extensible Markup Language (VoiceXML) Version 2.0,* W3C Recommendation, 2004, http://www.w3.org/TR/voicexml2

44 Amir N., Ron S., Laor N., *Analysis of an emotional speech corpus in Hebrew based on objective criteria,* ISCA Workshop on Speech and Emotion ; A conceptual framewor for research, Proceedings on line, 2000. http://www.qub.ac.uk/en/isca/proceedings

An XML-UNL Model for Knowledge-Based Annotation

Jesús Cardeñosa, Carolina Gallardo and Luis Iraola

Facultad de Informática, Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
{carde,carolina,luis}@opera.dia.fi.upm.es

Abstract. Efficient document search and description has radically changed with the widespread availability of electronic documents through Internet. Nowadays, efficient information search systems require to go beyond HTML-annotated documents. Complex information extraction tasks require to enrich text with semantic annotations that allow deeper and more detailed content analysis. For that purpose, new labels or annotations need to be defined. In this paper we propose to use UNL, an interlingua defined by the United Nations University, as a language neutral standard content representation in Internet. The use of UNL would open documents to a new dimension of semantic analysis, thus overcoming the limitations of current text-based analysis techniques.

1 Introduction

XML [1] is an standardized annotation language currently employed for a variety of purposes. For any given domain, the set of tags defined in its DTD attempts to capture the logical content structure of typical documents of the domain. So annotated, documents can be exploited by sophisticated document management systems that provide precise answers to users' queries. One the most promising uses of XML is the possibility of replacing textual document bases by their XML counterparts for document management purposes as well as for content management.

The capability of the XML standard to define the different information items present in a given document facilitates subsequent information extraction operations. This capability makes XML an ideal choice for annotating text corpora.
Annotated corpora have been one of the most useful resources in the last years for the study of linguistic phenomena. This orientation towards linguistic analysis has frequently associated corpus annotation with tasks such as part of speech tagging, chunking and parsing.. The Brown Corpus [2] or the British National Corpus [3] are examples of such annotated corpora. This sort of annotation is useful for many purposes but may be insufficient for information management tasks and for the location of very specific information items.

Corpus annotation poses significant difficulties when the goal is the representation and classification of information expressed in text form. While one could say that lexical and syntactic annotation of textual corpora is a more or less solved problem, semantic tagging is still a challenging goal currently aimed by several research lines.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 300–308.

Semantic corpora annotation has traditionally focused on the tasks of sense disambiguation of linguistic expressions [4] [5], definition of the conceptual relations between a heading verb and the dependent elements within the sentence [6], and frequently on tagging and classification of key concepts and specific elements pertaining to a given domain, like in [7] and [8]. Therefore, content analysis and representation by semantic tagging has in most cases a descriptive character and semantic annotation is mostly driven by the specific terminology of the domain. In multilingual corpora, semantic annotation is more superficial, and practically stops at the linguistic level.

A departure from this approach (domain and language dependency for semantic tagging) is one where textual information is expressed in a language-independent formalism whose semantic relations do not depend on any specific given domain. Such language independent formalisms are known as interlinguas in the field of Machine Translation.

An interlingua is an artificial language able to represent meaning in a language-independent way. Since one of the purposes of XML tagging is semantic annotation of the informational contents of a given document, there is in principle no special objection in applying XML tagging to represent a document written in an interlingua. The interlingua approach is not new and its origins can be traced back to the late eighties, when a number of multilingual machine translation systems were designed and implemented, such as Pivot [9] and Atlas-II [10]. In the nineties, machine translation systems evolved into the so-called knowledge-based machine translation systems, of which Kant [11] and Mikrokosmos [12] are two prominent examples.

The scalability problems of interlingua-based multilingual translation systems almost led to the rejection of the concept of interlingua. However, in 1996 the Institute of Advanced Studies of the United Nations University launched a new research project that rescued the interlingua approach for supporting multilingual content exchange in Internet by means of the use of UNL (Universal Networking Language).

UNL can be viewed as a reincarnation of the traditional concept of interlingua as an intermediate abstract representation common to all natural languages in a multilingual machine translation system. But UNL goes beyond the notion of a classical interlingua: it also serves for representing informational contents in any domain and in a language independent manner. UNL is endowed with an expressive capability similar to a natural language but with the features of a formalized language; its syntax and semantics are well defined, so UNL may be employed in information extraction and reasoning tasks.

2 The UNL System

UNL is an artificial language designed to represent textual content written in any natural language. The specifications of the UNL [13] formally define the language and its components. These are basically the following ones:

Universal words. They conform the vocabulary of the language, i.e., they can be considered the lexical items of UNL. In order to be able to express any concept occurring in a natural language, UNL proposes the usage of English headwords possibly modified by a series of semantic restrictions that eliminate potential ambiguities of those headwords. When there is no English headword suitable to express the intended concept, UNL allows the usage of words coming from other languages. In this way, the interlingua achieves the same lexical richness than natural languages but without their ambiguity. Take, for example, the English word "construction" meaning "the action of constructing" and also the "final product or result of constructing". The basic universal word "construction" will be paired with two different restricted universal words:

$\texttt{construction}_1$ `: construction(icl>action)`
$\texttt{construction}_2$ `: construction(icl>concrete thing)`

where "icl" is the abbreviation for "included".

Relations. There are a set of 41 basic relations that allow for the definition of any possible semantic relation among concepts. They include argumentative (agent, object, goal), circumstantial (purpose, time, place) and logic relations (conjunction and disjunction). For example, in the sentence "The boy eats potatoes in the kitchen", there is a main predicate ("eats") and three relations, two of them are instances of argumentative relations ("boy" is the agent of the predicate "eats", whereas "potatoes" is its object) and one circumstantial relation ("kitchen", the physical place where the action described in the sentence takes place).

Attributes. They express several types of semantic information that modify the relations and/or the universal words employed for expressing the content of a given text. This information includes time and aspect of the event, negation and modality of predication, type of reference of the entities described by the universal words, number and/or gender, etc. In the previous sentence, attributes are needed to express plurality in the object ("potatoes"), definite reference in the both the agent ("boy") and the place ("kitchen") and finally and special attribute denoting which UW is the head of the whole expression (the entry node).

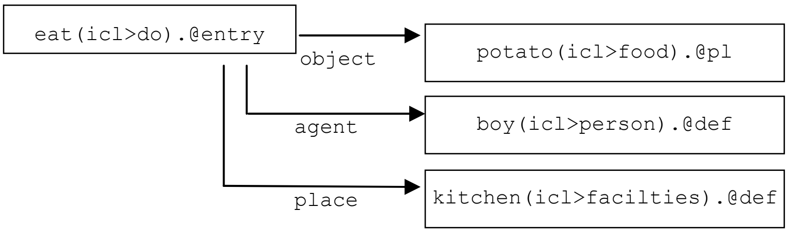


Fig 1. Graphical representation of a UNL expression.

Formally, a UNL expression has the form of a semantic net, where the nodes (universal words) are linked by arcs labeled with the UNL conceptual relations. The graphical representation of the sentence "the boy eats potatoes in the kitchen" in UNL is shown in figure 1, whereas its representation in the UNL syntax is as follows:

```
agt(eat(icl>do).@entry, boy(icl>person).@def)
obj(eat(icl>do).@entry, potato(icl>food).@pl )
plc(eat(icl>do).@entry, kitchen(icl>facilities).@def)
```

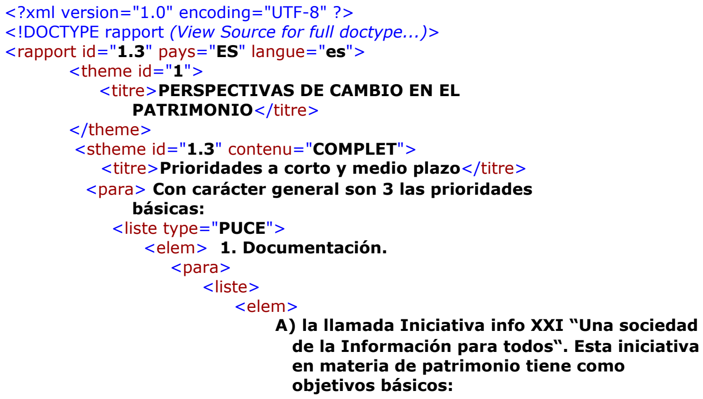
The capabilities of UNL for representing content independently from the source language led the authors to participate in the Herein system and to use UNL for supporting multilingual services in this particular system.

3 The UNL Approach in Herein

The Herein system (IST-2000-29355) [14] is a perfect example of a massively multilingual environment. It constitutes an Internet-based facility for improving cultural heritage management methods at the European level. Among the main tasks of the project, participant countries must compose a report providing detailed information about all aspects regarding cultural heritage.

Due to the large number of countries participating in the project (almost thirty) and the huge variety of topics that comprises cultural heritage (legislation, preservation, dissemination, etc.), there was an urgent need to standardize both the format and the structure of the contents that each country should provide. A definite structure was established and every country involved in Herein had to integrate its particular contents into such structure. Eventually, this structure turned out to be a de-facto standard for the description of the cultural heritage issues of a country.

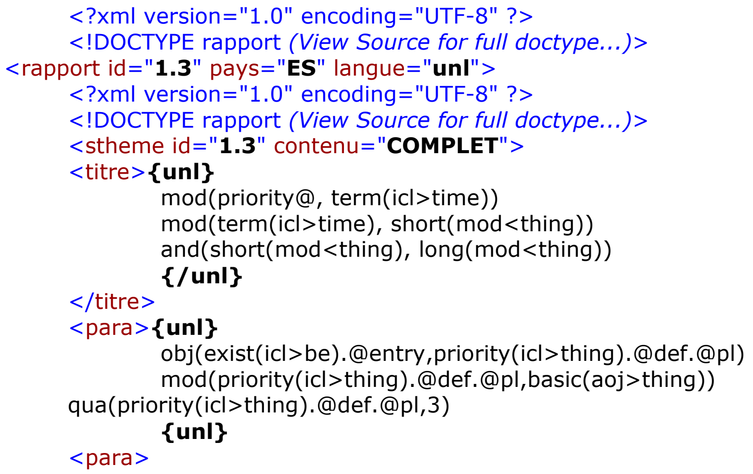
The supporting format chosen for the structured reports on cultural heritage of each participating country was XML. Figure 2 shows the appearance of a typical report in the Herein project: a fragment extracted from the Spanish Report.



```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE rapport (View Source for full doctype...)>
<rapport id="1.3" pays="ES" langue="es">
        <theme id="1">
            <titre>PERSPECTIVAS DE CAMBIO EN EL
                PATRIMONIO</titre>
        </theme>
         <stheme id="1.3" contenu="COMPLET">
            <titre>Prioridades a corto y medio plazo</titre>
          <para> Con carácter general son 3 las prioridades
                básicas:
             <liste type="PUCE">
                 <elem>  1. Documentación.
                     <para>
                         <liste>
                             <elem>
                                 A) la llamada Iniciativa info XXI "Una sociedad
                                   de la Información para todos". Esta iniciativa
                                   en materia de patrimonio tiene como
                                   objetivos básicos:
```

Fig 2. Example of Spanish content in XML structure

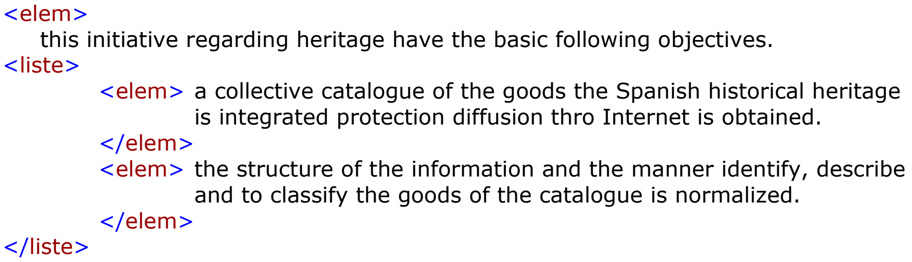
The complete report of the Spanish cultural contents was codified into UNL as an initiative of the Spanish government, representative institution of the Herein contents in the Spanish language, and in collaboration with the Spanish Language Center, representative and responsible of the Spanish language in the UNL program.



```
          <?xml version="1.0" encoding="UTF-8" ?>
          <!DOCTYPE rapport (View Source for full doctype...)>
      <rapport id="1.3" pays="ES" langue="unl">
          <?xml version="1.0" encoding="UTF-8" ?>
          <!DOCTYPE rapport (View Source for full doctype...)>
          <stheme id="1.3" contenu="COMPLET">
          <titre>{unl}
                  mod(priority@, term(icl>time))
                  mod(term(icl>time), short(mod<thing))
                  and(short(mod<thing), long(mod<thing))
                  {/unl}
          </titre>
          <para>{unl}
                  obj(exist(icl>be).@entry,priority(icl>thing).@def.@pl)
                  mod(priority(icl>thing).@def.@pl,basic(aoj>thing))
          qua(priority(icl>thing).@def.@pl,3)
                  {unl}
          <para>
```

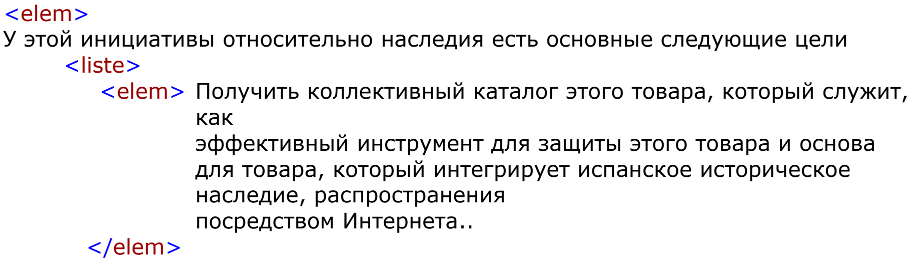
Fig 3. UNL code embedded into an XML document.

The UNL code has been embedded into the XML structure shared by all reports, as if the UNL code were another natural language (see figure 3). The difference lies in the fact that the aforementioned code can be extracted from the XML file and employed by the natural language generator of any language. After generation [15], the corresponding text will be inserted into the XML structure of the document. The result is shown in figures 4 and 5 for the English and Russian language generators.



```
<elem>
   this initiative regarding heritage have the basic following objectives.
<liste>
         <elem> a collective catalogue of the goods the Spanish historical heritage
                is integrated protection diffusion thro Internet is obtained.
         </elem>
         <elem> the structure of the information and the manner identify, describe
                and to classify the goods of the catalogue is normalized.
         </elem>
</liste>
```

Fig 4. Output text of the English generator



```
<elem>
У этой инициативы относительно наследия есть основные следующие цели
     <liste>
         <elem> Получить коллективный каталог этого товара, который служит,
                как
                эффективный инструмент для защиты этого товара и основа
                для товара, который интегрирует испанское историческое
                наследие, распространения
                посредством Интернета..
        </elem>
```

Fig 5. Output text of the Russian generator

The complete integration of UNL into the Herein system is illustrated in figure 6. In this figure, it can be seen how an original XML document about Spanish heritage is the input to an UNL editor once its XML tags have been removed from it and the textual content extracted. The UNL editor is a tool that enables its user to encode Spanish sentences into UNL expressions. The degree of automation depends on the current state of Spanish-UNL dictionaries and its syntactic and semantic analyzers. The output of the UNL editor is a plain document written in UNL (that is, no XML tagging is present). This UNL document goes directly into the language generators, for example the English and Russian language generators. These generators yield the contents of the original XML Spanish document but now in English and Russian respectively. The final step is the "XMLization" of these plain documents according the DTD adopted in the Herein system.

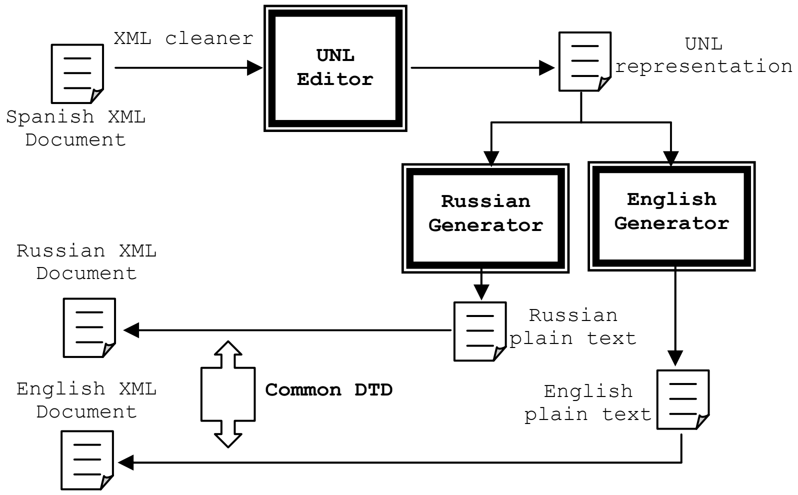


Fig 6. Model for the Integration of UNL into Herein

Within the Herein system, UNL has been integrated with XML mainly for the support and maintenance of multilingual documents. However, the integration of UNL into XML can be further explored in order to take further advantage of the UNL code for semantic annotation.

4 Knowledge-Based Annotation in XML: a Three-Dimensional Approach

Currently, a closer integration of UNL and XML is being studied from a different perspective [16] but within the same framework here described. This innovative work attempts to define environments and architectures that allow the inclusion of XML tags that identify individual UNL elements (i.e. universal words, relations and attributes). This fine-grained semantic representation will pave the way to more intelligent information extraction tasks. This is possibly the most immediate research line that would produce an effective integration of XML and UNL. If we are able to define a suitable XML syntax for representing UNL, and also to semantically annotate the content of a document not only according to an set of domain-specific descriptive terms but also using the semantic relations that connect the concepts present in the document, we will transform a "one-dimensional" textual document into a "three-dimensional" document.

Why a third dimension? We may consider the text as the first dimension of a document. It is the basis of any linguistic analysis and it is certainly the basis of the encoding process in the UNL system. Layout, formatting and hyper-linking constitute a second dimension of the document. This second dimension provides cues about the specific information pieces contained in a document and facilitates searching and extraction. But, if in addition to the first and second dimensions we are able to capture the semantic relations among the concepts present in the document, we may say that a third dimension has been made available, a dimension where the knowledge contained in a given document is made explicit. Document management systems become knowledge management systems by exploiting this third dimension, implementing knowledge-based reasoning procedures able to produce intelligent answers to complex queries.

The integration of the UNL representation will improve the quality and depth of the knowledge expressed by XML tagging. The UNL relations are based on what has been traditionally known as conceptual or thematic relations or simply cases. Along this line, other authors are using these relations as the leitmotiv for semantic annotation [6]. However, at this point some reflections should be made about the nature of UNL, as they back UNL as a firm candidate for the task of representing the knowledge level in any XML document. Key UNL characteristics are:

(a) The set of necessary relations existing between concepts is already standardized [13]. This is the result of intensive research on the thematic roles existing in natural languages by a number of experts in the area of MT and AI.

(b) Similarly, the set of necessary attributes that modify concepts and relations is fixed and well-defined.

(c) The UNL syntax and semantics are formally defined, UNL can be viewed as a formalism for representing knowledge.

In short, UNL has in its favor the standardization of the process of representing knowledge coming from documents written in a natural language. In the following example, we show the approach to be followed along this direction. We will show an example of the abovementioned third dimension applied to a paragraph extracted from the Herein Spanish report (originally in Spanish but here in English for readability reasons):

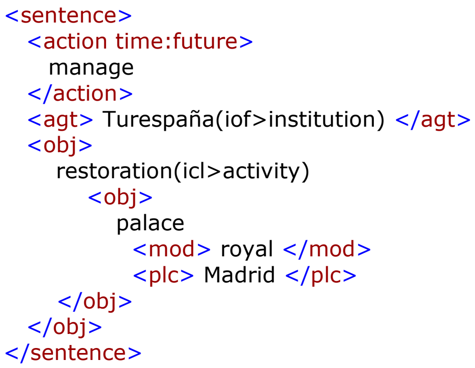
```
<para> The restoration of the Royal Palace of Madrid
will be managed by Turespaña. </para>
```

Its UNL representation is as follows:

```
agt(manage(icl>do).@entry.@future,
    Turespaña(iof>institution))
obj(manage(icl>do).@entry.@future,
     restoration(icl>activity).@def)
obj(restoration(icl>activity).@def,
     palace(icl>building).@def)
mod(palace(icl>building).@def, royal(mod<thing))
plc(palace(icl>building).@def, Madrid(iof>city))
```

The encoded meaning is that of an action carried out by an agent (agt) and described as a managing action performed by the institution named (iof) 'Turespaña'. The object (obj) of the managing action is a restoration activity. It is also specified that the object of the restoration is a palace, a type of building (icl), modified (mod) by the property of being a royal palace and located (plc) in Madrid. Additionally, the time of the action is future.

It is clearly possible to define an XML-based tag language for expressing the elements of a UNL representation: UNL relations could be considered as XML tags, attributes could be represented as XML attributes and universal words may just be textual data enclosed within UNL relation tags. Figure 7 presents the previous UNL representation along these lines.



```
<sentence>
  <action time:future>
    manage
  </action>
  <agt> Turespaña(iof>institution) </agt>
  <obj>
     restoration(icl>activity)
        <obj>
           palace
             <mod> royal </mod>
             <plc> Madrid </plc>
        </obj>
  </obj>
</sentence>
```

Fig 7. UNL representation using XML-based tags.

This representation conforms to a very precise characterization of the semantic relations and the concepts present in the sentence. Therefore, the knowledge implicit in the sentence has been explicitly formalized and integrated within an XML-based document structure.

5 Conclusions

In this paper we have presented a new approach for representing knowledge contained in textual documents using an interlingua. The use of UNL allows and facilitates the integration of knowledge into an XML structure by means of the definition of a set of XML-based tags and attributes suited for the basic elements of a UNL representation. At the moment we are testing the adequacy of UNL representations embedded into XML documents for information extraction tasks. We are also devising an interactive system of queries over contents so represented. Our approach may prove useful for annotating multilingual text corpora with semantic information, thus extending the range of applications of an interlingua originally designed for multilingual generation purposes.

References

1. W3C Recommendation. *Extensible Markup Language*, www.w3.org/TR/2004/REC-xml-20040204/, 2004.
2. Francis W.N. and Kucer H. *Brown Corpus Manual*, helmer.aksis.uib.no/icame/brown/bcm.html, 1979.
3. Guy A., and Burnard L *The BNC Handbook - Exploring the British National Corpus with SARA*. Edinburgh, UK. Edinburgh University Press, 1998.
4. Garside, R. and Rayson, P. Higher-level annotation tools, in Garside, R., Leech, G., and McEnery, A. (eds.) *Corpus Annotation: Linguistic Information from Computer Text Corpora*. Longman, London, 1997.
5. Thomas, J., and Wilson, A. Methodologies for studying a corpus of doctor-patient interaction. In J. Thomas and M. Short (eds.). *Using corpora for language research*. Longman, London, 1996.
6. Kingsbury M., Palmer M., and Marcus M. In Proceedings of the Human Language Technology Conference, San Diego, California, 2002.
7. Ohta T., Tateisi Y., Takai Y. and Tsujii J. A Semantically Annotated Corpus from MEDLINE Abstracts. In the Proceedings of Genome Informatics. Tokyo, Japan. Universal Academy Press Inc., 1999.
8. Vintar S. and Volk M. Semantic Relations in Concept-Based Cross-Language Medical Information Retrieval. In Proceedings of the ECML/PKDD Workshop on Adaptive Text Extraction and Mining (ATEM), Cavtat-Dubrovnik, Croatia, 2003.
9. Muraki, K.: PIVOT. Two-phase machine translation system. Proceedings of the Second Machine Translation Summit. Tokyo, Japan, 1989.
10. Uchida, H.: ATLAS-II. A machine translation system using conceptual structure as an interlingua. Proceedings of the Second Machine Translation Summit. Tokyo, Japan, 1989.
11. Nyberg E. and Mitamura T. The KANT System: Fast, Accurate, High-Quality Translation in Practical Domains. Proceedings of COLING-92: 15th International Conference on Computational Linguistics. Nantes, France, 1992.
12. Beale, S., S. Nirenburg and K. Mahesh. Semantic Analysis in the Mikrokosmos Machine Translation Project. Proceedings of the 2nd Symposium on Natural Language Processing. Bangkok, Thailand, 1995.
13. Uchida, H.: The Universal Networking Language. Specifications. www.undl.org, 2002.
14. HEREIN Project (IST-2000-29355): Final Report. European Commission, 2003.
15. Cardeñosa J., Gallardo C and Tovar E. Standardization of the generation process in a multilingual environment. In Proceedings of the International Conference Convergences 03. December 2003. Alexandria. Egypt, 2003.
16. Wang-Ju. Work in progress accessible from: www-clips.imag.fr/geta/wang-ju.tsai/viewer/Multiple_js.htm, 2004.

A "Pivot" XML-Based Architecture for Multilingual, Multiversion Documents: Parallel Monolingual Documents Aligned Through a Central Correspondence Descriptor and Possible Use of UNL

Najeh Hajlaoui, Christian Boitet

équipe GETA, laboratoire CLIPS,
385 rue de la bibliothèque - BP 53, 38041 Grenoble Cedex 9 - France
Christian.Boitet@imag.fr

Abstract. We propose a structure for multilingual, multiversion documents, built on the model of the web-oriented, cooperative lexical multilingual data base PAPILLON: a document is represented by a collection of monolingual XML "volumes" interlinked by a central volume of "interlingual links". Here, the links relate subdocuments (XML trees) corresponding to each other in monolingual "volumes". We are developing a Java application to enable direct editing of a multilingual document through the web, at the level of monolingual volumes as well as through bilingual or trilingual interfaces inspired by those of commercial "translation workbenches". Another goal is easy integration with machine translation and multilingual generation tools. For this, we add a special UNL volume. In a first stage, we split the UNL-xml document in several monolingual documents, again represented by XML files. Each document contains the text in a particular language, plus the corresponding UNL graphs, and can be modified independently. The interface is easy to build, but realigning the documents after a series of such modifications is a very difficult task.

1 Introduction

Due to Internet, the number of available documents grows dramatically. There is a strategic need for companies to control information written in more than 30 languages (HP, IBM, MS, Caterpillar). This requires the installation of powerful and effective management tools of multilingual "synchronized" documents.
There are techniques of large-grained linking (on the level of HTML pages). However, there are no techniques for structuring multilingual documents so as to allow fine-grained synchronization (at paragraph or sentence level) and even less permitting editability through the Web.

The interest to synchronize at least on the level of the sentences is double:

- for the translation and human revision with the assistance of techniques of HTHM (Human Translation Helped by Machine) and in particular of translation memory.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 309–325.

– for the increase in the number of languages of a multilingual document, it would be useful to synchronize the versions of multilingual documents with a representation such as the multilingual UNL document format, allowing to increase the number of languages of the document in an economic way by calling distant deconverters.

The paper is organized as follows.

In the first part, we put our research in perspective with the UNL project (Universal Networking Language). We show the advantage and the limits of the UNL format, and discuss some aspects related to the management of the information systems.

In the second part, we present a possible solution to manage the correspondences between the linguistic versions of a multilingual document: it consists in splitting a document in UNL format in several monolingual documents.

The third part is devoted to the reconstitution of links broken between the documents, and to a mockup and prototypes of interfaces.

In the conclusion, we show the flexibility of such a structure of multilingual, multiversion documents, and its applicability in several domains.

2 Problems

2.1 Situation of the Problem

There are many multilingual documents, which are modified separately (leaflets, booklet, etc.). After a certain time, we wish to make them coherent [1]. That means finding the correspondences (alignments) and reconstituting a complete and coherent (monolingual) "source" document. For this, modifications in target languages have to be translated into the source language.

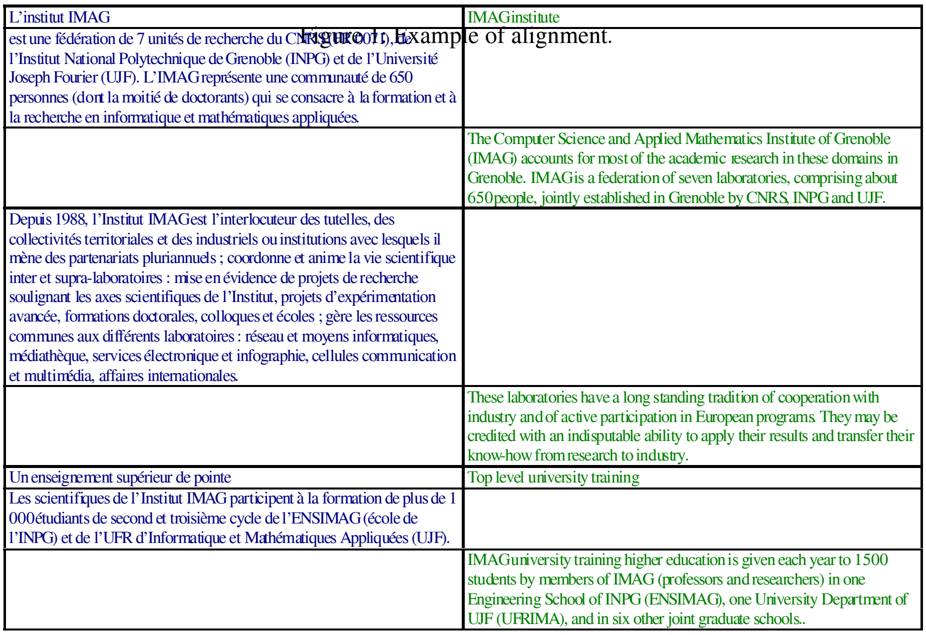
A. Assimi, in his PhD work, treated the case of the non-centralized management of the evolution of multilingual parallel documents.

In the industry, it is frequent that documents are managed on the same platform without being linked at a fine-grained level like that of sentences or paragraphs.

For example, technical documents are usually aligned at the level of HTML pages. Generally, free modification by readers (final users) is not authorized (whereas it is usually permitted for leaflets in Word).

Several problems appear in real life:

1. As shown in Figure 1, alignment (based on sentences considered to be exact mutual translations of each other) may be quite sparse, even with only 2 languages, after only one batch of modifications in one language.
2. There is no explicit link between the monolingual (real) documents constituting the (virtual) multilingual document.
3. In some contexts like the European Heritage web site, a UNL document is also built in parallel, as a simple list of UNL-graphs, with no document structure. The problem can then be abstracted as in Table 1 below.



L'institut IMAG	IMAGinstitute
est une fédération de 7 unités de recherche du CNRS (UR 0071), de l'Institut National Polytechnique de Grenoble (INPG) et de l'Université Joseph Fourier (UJF). L'IMAG représente une communauté de 650 personnes (dont la moitié de doctorants) qui se consacre à la formation et à la recherche en informatique et mathématiques appliquées.	
	The Computer Science and Applied Mathematics Institute of Grenoble (IMAG) accounts for most of the academic research in these domains in Grenoble. IMAGis a federation of seven laboratories, comprising about 650people, jointly established in Grenoble by CNRS, INPG and UJF.
Depuis 1988, l'Institut IMAGest l'interlocuteur des tutelles, des collectivités territoriales et des industriels ou institutions avec lesquels il mène des partenariats pluriannuels ; coordonne et anime la vie scientifique inter et supra-laboratoires : mise en évidence de projets de recherche soulignant les axes scientifiques de l'Institut, projets d'expérimentation avancée, formations doctorales, colloques et écoles ; gère les ressources communes aux différents laboratoires : réseau et moyens informatiques, médiathèque, services électronique et infographie, cellules communication et multimédia, affaires internationales.	
	These laboratories have a long standing tradition of cooperation with industry and of active participation in European programs. They may be credited with an indisputable ability to apply their results and transfer their know-how from research to industry.
Un enseignement supérieur de pointe	Top level university training
Les scientifiques de l'Institut IMAG participent à la formation de plus de 1 000étudiants de second et troisième cycle de l'ENSIMAG (école de l'INPG) et de l'UFR d'Informatique et Mathématiques Appliquées (UJF).	
	IMAGuniversity training higher education is given each year to 1500 students by members of IMAG (professors and researchers) in one Engineering School of INPG (ENSIMAG), one University Department of UJF (UFRIMA), and in six other joint graduate schools..

Figure 1: Example of alignment.

Language 1 (FR)	Language 2 (EN)	……	UNL
φ^{FR}_{1}	φ^{EN}_{1}		γ_1
φ^{FR}_{2}	ϕ		γ_2
ϕ	φ^{EN}_{3}		γ_3
…			
φ^{FR}_{n}	φ^{EN}_{n}		γ_m
…			
$\varphi^{FR}_{N\text{-}FR}$	$\varphi^{EN}_{N\text{- }EN}$		γ_M

Table 1: correspondences between sentences.

φ^{l}_{i} = sentence with identifier i in language l

γ_m= *UNL* graph representing the meaning of one (occurrence of a) sentence

A very simple idea is to seek an identifier for a set of equivalent sentences, with

- *$\varphi_1 \cong \varphi_2$ if and only if UNL (φ_1) = UNL (φ_2)*
- *$\varphi^{l}_{i} \cong \varphi^{l'}_{i'}$ if and only if $\sigma(\varphi^{l}_{i}) = \sigma(\varphi^{l'}_{i'})$*

  σ is the equivalence of the intuitive means but testable by a human translation
- *$\varphi^{l}_{i} \cong \varphi^{l'}_{i'}$ if and only if $\rho(\varphi^{l}_{i}) = \rho(\varphi^{l'}_{i'})$*

  ρ is defined in a restrictive and operational way. Here, ρ = UNL.

A first problem is to calculate links from the UNL graphs to the sentences in each monolingual document. They may be modeled as a function $\Pi : 1..M \times L \rightarrow \mathbb{N}$ or as a relation in $1..M \times L \times \mathbb{N}$.

If we choose the first possibility, a UNL graph (in the parallel UNL document) cannot be linked by Π to more than 1 sentence in any language, which implies that 2 identical UNL graphs can appear in the central list. The idea is that, after some reordering and duplication, the list of UNL graphs can be linked to the list of sentences (the terminal nodes) of the xml structure of each monolingual document, with "no crossing". In other words, Π is then monotonically increasing in its first component.

We might also choose the second possibility, where Π is a relation, so that all occurrences of sentences with the same meaning could be linked to the same UNL graph. Then, the parallel UNL file should represent a set of UNL graphs, with no possible repetition.

However, both these possibilities lead to problems. Let us show it on the first only (Π is a function). Then,

Π (m, l) = n if and only if

1. $\delta (\gamma_m, l) \approx \varphi^l_n$ where δ stands for "deconversion" (from UNL)
2. $\lambda (\varphi^l_n) = \gamma_m$ where λ stands for "enconversion" (into UNL)

Π (m, l) = <u>nil</u> otherwise (γ_m does not correspond to any sentence).

To establish the links between the UNL graphs and the sentences implies then to call all deconverters on all graphs, and to compare the results with the actual sentences. But deconverters are constantly updated, may be unavailable at some time, and sentences may also be modified by hand. Hence, with all probability, only very few links will be established. What would be needed is a process to compare the meaning of a sentence present in a document with that of a sentence produced by deconversion "on-the-fly". But that is a hard and perhaps harder problem!

We can also attack the problem from the other side, that is, we can try to establish links from the sentences to the UNL graphs. This linking is the inverse ψ of Π. Again, ψ can be a function or a relation. In the UNL format, it is a function, which implies that, if a sentence is truly ambiguous and corresponds to several different UNL graphs, one of them has to be chosen in the representation. Let us adopt this restriction.

We have then $\psi : \mathbb{N} \times L \rightarrow \mathbb{N}$, and

ψ (n, l) = m if and only if Π (m, l) = n.

We encounter a similar problem: to compute ψ, we have to "enconvert" each sentence, and compare the result with the UNL graphs in the list. But (1) enconversion is harder than deconversion, and (2) the UNL language allows for more than one way of representing a given interpretation of a sentence.

We should then develop techniques to test the synonymy of 2 UNL graphs… but it is quite certain that any proposed solution will be incomplete, because the problem of deciding whether 2 formal expressions have the same meaning is undecidable as soon as the considered formulas pertain to a rich enough formal system. For example, it is undecidable whether 2 java programs compute the same function.

This shows that the solution consisting in putting some UNL-related or UNL-like representation as a central structure leads to problems. It also imposes the added difficulty to build a correct and complete UNL-xml document.

Hence, our solution will be to design a specific central structure linked to all sentences of all monolingual documents, and to the UNL graphs. A separate problem will be to determine whether some intersection or union of the monolingual document structures should be reflected in the central structure or not.

2.1.1 Evolution of the Versions of a Multilingual Document

We introduce the term "polyphrase" to denote a set of sentences in several languages and UNL graphs, formed from an initial set of such elements, the "kernel" of the polyphrase, deemed to be semantically equivalent.

In most cases, the kernel is simply one sentence in a given language, all other sentences are obtained by translation or corrections, and the UNL graphs by enconversion and then direct edition or coedition.

While the kernel corresponds to exactly one intended meaning, the evolution of the polyphrase may introduce new meanings. To trace them, we need to add a notion of version to the elements of a polyphrase, and by extension to all parts of a multilingual document.

The passage to a new version can happen in many cases:

- correction of errors.
- human revision.
- addition of another language.
- change of order of linguistic objects.
- addition of new polyphrases.

The preceding points are important factors, which influence the increase in the number of versions of a multilingual document, and the unalignment rate of these versions.

2.1.2 Coherence of the Versions

The coherence of the versions is directly related to the concept of alignment. 2 versions in 2 languages will said to be "coherent" if their aligned documents are mutual translations of each other. Alignments should go at least to the level of sentences. In our first mockup (see below), we stop there, but finer units such as segments and words may be quite useful to help human translators or posteditors.

The coherence of the versions of the database is distinct from that of an environment of translation; the graph of dependence is fixed and the ascending translation process respecting alignment generates a coherent version.

A new version then traverses a development cycle until it becomes frozen and/or validated, before entering in a state of "public" availability. It can then be used in a translation memory.

2.2 Advantages of the UNL Language and Limits of the UNL Format

We choose UNL [2] as our interlingua for various reasons:

1. it is specifically designed for linguistic and semantic machine processing,
2. it derives with many improvements from H. Uchida's pivot used in ATLAS-II (Fujitsu), still judged as the best quality MT system for English-Japanese, with a large coverage (586,000 lexical entries in each language in 2001),
3. participants of the UNL project[1] have built "deconverters" from UNL into about 12 languages, and at least the Arabic, Indonesian, Italian, French, Russian, Spanish, and Thai deconverters were accessible for experimentation through a web interface in spring 2003,
4. although formal, UNL graphs (see below) are quite easy to understand with little training and may be presented in a "localized" way to naive users by translating UNL symbols (semantic relations, attributes) and lexemes (UWs) into symbols and lexemes of their language,
5. the UNL project has defined a format embedded in html for files containing a complete multilingual document aligned at the level of utterances, and produced a "visualizer" transforming a UNL file into as many html files as languages, and sending them to any web browser.

The UNL representation of a text is a list of "semantic graphs", each expressing the meaning of a natural language utterance. Nodes contain lexical units and attributes; arcs bear semantic relations. Connex subgraphs may be defined as "scopes", so that a UNL graph may be a hypergraph.

The lexical units, called Universal Words (UW[2]), represent (sets of) word meanings, something less ambitious than concepts. Their denotations are built to be intuitively understood by developers knowing English, that is, by all developers in NLP. A UW is an English term or special symbol (number…) possibly completed by semantic restrictions: the UW "process" represents all word meanings of that lemma, seen as citation form (verb or noun here), and "process(icl>do, agt>person)" covers only the meanings of processing, working on, etc.

The attributes are the (semantic) number, genre, time, aspect, modality, etc., and the 40 or so semantic relations are traditional "deep cases" such as agent, (deep) object, location, goal, time, etc.

One way of looking at a UNL graph corresponding to an utterance in language L is to say that it represents the abstract structure of an equivalent English utterance "seen from L", that is, where semantic attributes not necessarily expressed in L may be absent (e.g., aspect coming from French, determination or number from Japanese, etc.).

The UNL format, whether UNL-html or UNL-xml, gives for the moment a simple solution: a multilingual document is only one large file where the alignment of the various versions (languages and revisions) is done at the level of each sentence. But, in general, two parallel documents in two different languages cannot be aligned at this level. Indeed, a sentence in L1 can correspond to two or three sentences in L2 and conversely (m-n possibility).

[1] http://unl.ias.unu.edu

[2] Universal Word, or Unit of Virtual Vocabulary

Moreover, the order of a list of sentences or paragraphs can vary from one language to another (for example because of a lexicographical sorting). Thus, the idea from where we left in the introduction is good, but must be refined.

2.3 Aspects Related to the Management of Information Systems

The problem of management of correspondences and coherence of MPDs (Multilingual Parallel Documents) still remains open: there is no adequate concrete solution, indeed there is a lack of tools, methods, practices and models to describe, maintain and refine the correspondences between versions of the same document in several languages.

An important point is that the suggested techniques must be usable in practice and as practical as possible in the known information systems. Let us see how the problem is posed on this level.

2.3.1 Centralized Management

In the case of centralized management of documents, the problem is easier to solve as soon as (1) a unique XML format is used for exchanging and storing data, and (2) there is a central place to describe and control the correspondences between linguistic versions. The disadvantage, however, is that the freedom to modify individual versions is limited.

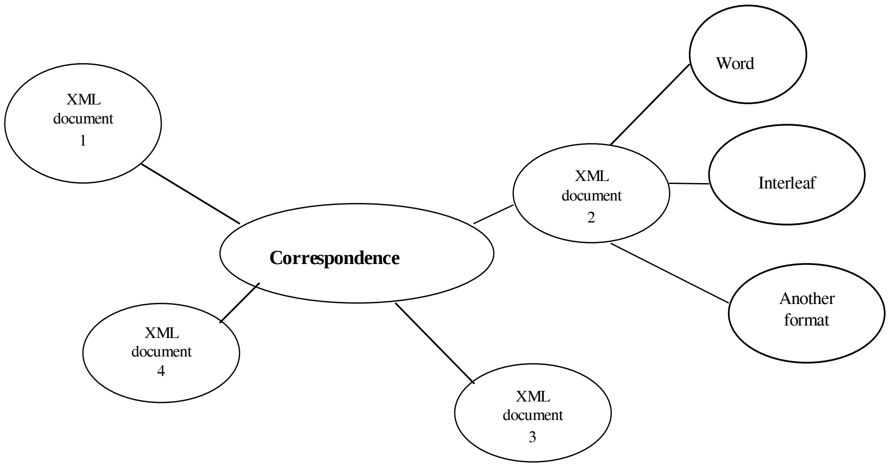


Figure 2: correspondence between centralized documents (XML formats)

Indeed, the life cycle of a multilingual document organized in this way has to be controlled from the start using certain mechanisms of observation and protection of the correspondences.

2.3.2 Non-Centralized Management

There are many cases where the various versions of a document are not centralized, for instance because they have to be processed with different tools on different platforms. To realign them after a series of modifications have been done is quite difficult, and to rebuild a coherent complete original is even more difficult.

On the formatting side, there are m-n possibility of correspondences for several distributed documents, n different formats and 2n filters.

A. Assimi [1] analyzed and solved a part of these and other problems posed by the management of the non-centralized evolution of multilingual parallel documents. He used a structuring of the multilingual texts by a multicolumn table, which is not practicable for documents of big size (technical documentation, catalogues...). In his thesis, he reports that this simple solution worked for certain needs of customers, but was limited to the management of small documents such as the brochure of the IMAG Institute (Informatics and Applied Mathematics at Grenoble), which contains approximately 2000 words, that is 8 standard pages of translation, or 4 pages of Word.

2.3.3 Principe of Solution

Starting from the study made in the two preceding cases, we see the need for designing tools and methods allowing practical management of large multilingual documents. In particular, it is necessary to describe and to maintain linguistic correspondences at a very fine level between n versions in m languages, while allowing new versions to appear in any language independently of others.

For that, the idea is to represent the correspondences between the structural trees of n parallel monolingual documents by a separate structure, of a different type, connecting fragments of trees with as few constraints as possible, as is done for the macrostructure of the multilingual lexical data base PAPILLON.

3 The Versioning Problem and a First Solution

We simply adopt the solution implemented in PAPILLON (storage of the modifications on standby in the form of XSLT transformations in the private space of each contributor) and draw from our preliminary experiment in management of versions for XML documents representing virtual electronic components.

In order to manage the successive versions of a multilingual document, we introduce the concept of status of a version.

3.1 Status of Documents and Versions

The status of any part of a document can be:

- **modifiable**: when its contents can still undergo modifications.
- **frozen**: when its contents cannot be modified but are not yet validated.

– **validated**: when its contents have been validated. A validated part may be put on some sharable reference space.

We define the order: modifiable < frozen < validated.

Suppose a multilingual document has content in n languages (including UNL if present).

The "last version" of any part of this document is the n-uple consisting of the maximum version number of all its polyphrases.

A "version" of a document is any n-uple of version numbers less or equal to the last version (component by component).

The status of a version of a document is the minimum of the statuses of the sub-document corresponding to that version.

3.2 From a Multilingual Document to Several Monolingual Documents

The basic idea is to separate the monolingual documents and to represent their correspondences in an autonomous "pivot" structure. It was also the idea of A. Assimi, but we use it here in a context where the formats to be synchronized are standard XML formats. We find it too in PAPILLON, where each dictionary of lexies (word meanings or monolingual acceptions) is represented by an XML file, as well as the "pivot" or "hub" formed by the axies (links between lexies).

In addition, more and more annotations are introduced into documents for various applications (IR, summary, categorization...). They can be annotations related to the language (like GDA of K. Hashida) or annotations only related to the contents (graphs UNL, semantic categories...).

At this point, we consider two ways of separating the monolingual documents: partial separation and total separation.

3.2.1 Partial Separation

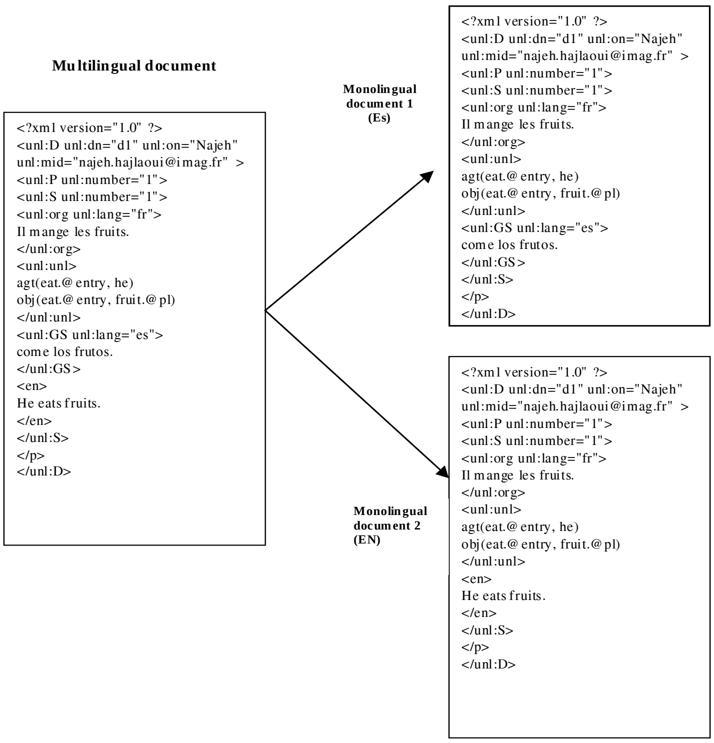
Let us suppose for the moment that we have a multilingual document in UNL-xml format aligned on the level of the sentence. Suppose we want to switch to the non-centralized management situation, for example to let 15 persons edit the same document in 15 languages.

The idea of partial separation is then to split the UNL-xml representation into 15 monolingual documents enriched by the original content (source language) and its UNL representation as shown in the following example.

This makes it possible to make local modifications in each language and thus to introduce different versions. Here, for example, the sentence "He eats fruits" becomes "He is eating fruits" with the corresponding modification of UNL-xml format, and a second version of the English document appears (figure 4).

3.2.3 Total Separation

Here, we split the UNL-xml representation in several monolingual documents by considering the fact that the original is also a monolingual document as well as its UNL representation.

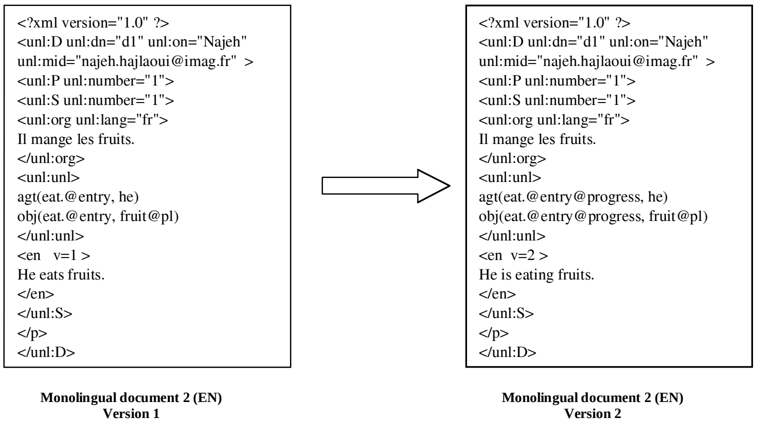


```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit.@pl)
</unl:unl>
<unl:GS unl:lang="es">
come los frutos.
</unl:GS>
<en>
He eats fruits.
</en>
</unl:S>
</p>
</unl:D>
```

```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit.@pl)
</unl:unl>
<unl:GS unl:lang="es">
come los frutos.
</unl:GS>
</unl:S>
</p>
</unl:D>
```

```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit.@pl)
</unl:unl>
<en>
He eats fruits.
</en>
</unl:S>
</p>
</unl:D>
```

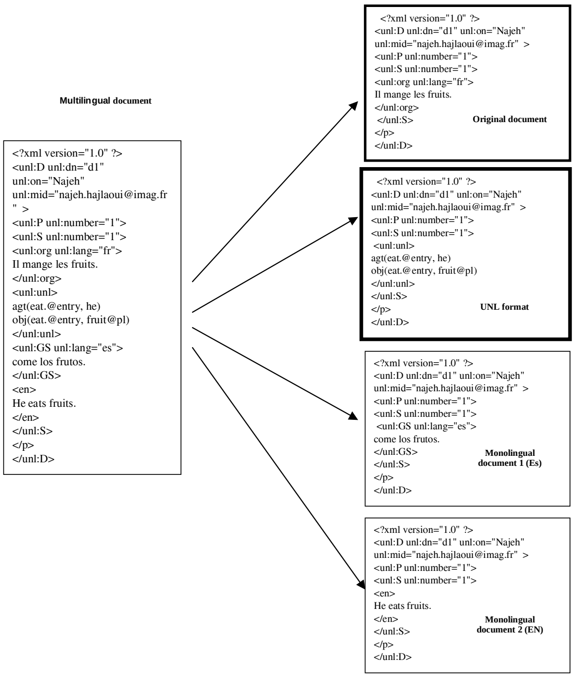
Figure 3: partial separation of a multilingual document.



```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit@pl)
</unl:unl>
<en   v=1 >
He eats fruits.
</en>
</unl:S>
</p>
</unl:D>
```

```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry@progress, he)
obj(eat.@entry@progress, fruit@pl)
</unl:unl>
<en  v=2 >
He is eating fruits.
</en>
</unl:S>
</p>
</unl:D>
```

Figure 4: evolution of monolingual document.



```
<?xml version="1.0" ?>
<unl:D unl:dn="d1"
unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr
" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
<unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit@pl)
</unl:unl>
<unl:GS unl:lang="es">
come los frutos.
</unl:GS>
<en>
He eats fruits.
</en>
</unl:S>
</p>
</unl:D>
```

```
  <?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<unl:org unl:lang="fr">
Il mange les fruits.
</unl:org>
 </unl:S>
</p>
</unl:D>
```

```
  <?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
 <unl:unl>
agt(eat.@entry, he)
obj(eat.@entry, fruit@pl)
</unl:unl>
</unl:S>
</p>
</unl:D>
```

```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
 <unl:GS unl:lang="es">
come los frutos.
</unl:GS>
</unl:S>
</p>
</unl:D>
```

```
<?xml version="1.0" ?>
<unl:D unl:dn="d1" unl:on="Najeh"
unl:mid="najeh.hajlaoui@imag.fr" >
<unl:P unl:number="1">
<unl:S unl:number="1">
<en>
He eats fruits.
</en>
</unl:S>
</p>
</unl:D>
```

Figure 5: total separation of a multilingual document.

This separation of UNL-xml representation can be improved by gathering technical information common to the monolingual documents in the same document of description. That is possible using XML facilities for creating and managing meta-data.

3.3 Discussion

In the first technique of separation

- the autonomous evolution of each linguistic version is possible; that constitutes an important advantage for human revision.
- the source language, the target language and the UNL representation are in the same file, which allows the simple reuse of tools and interfaces of "traditional" MAHT (Machine-Aided Human Translation) systems, there must be a source text and a target text.
- There exist "local" UNL tools which begin to be really used in practice.

In the second technique and since we have only one UNL-xml file, controlled and centralized at the level of sentences, this last file cannot remain strictly parallel with each linguistic version; it has to some extent to reflect modifications. For example, if we replace in the French file a sentence by two sentences, it will be necessary to leave the UNL graph for the old large sentence in the UNL file and to add 2 new UNL graphs.

Consequently, the two preceding techniques are not satisfactory and there remains the problem of the maintenance of the correspondences.

If modifications are done in all the versions, we cannot use the UNL file as "center" also serving to memorize these modifications.

The principle of our solution is inherited from the area of technical document management and from the PAPILLON project. This solution is based on two important points:

- Monotony: never erase anything in any "volume" (an XML file) but add new evolutionary versions.
- Modularity: represent the correspondences in a separate way.

We propose the following diagram:

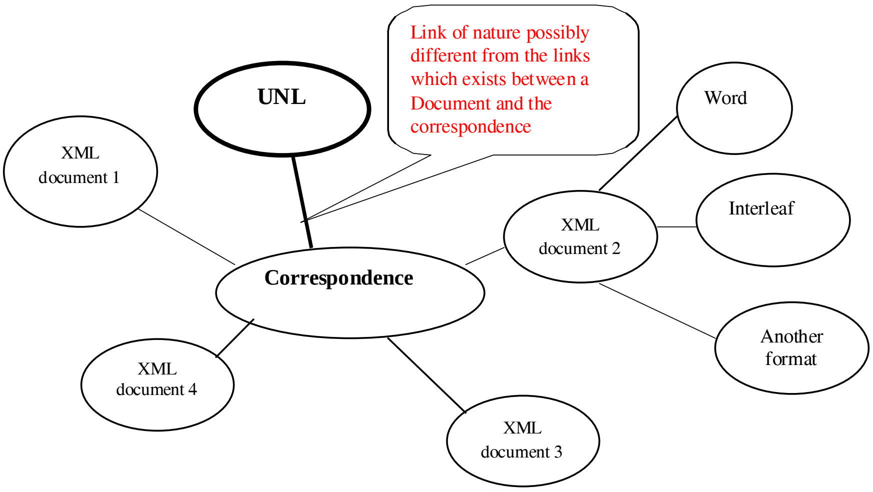

Link of nature possibly different from the links which exists between a Document and the correspondence
UNL
Word
XML document 1
Interleaf
XML document 2
Correspondence
Another format
XML document 4
XML document 3

Figure 6: correspondence between several documents.

4 Second solution: a central representation of all correspondences between monolingual and UNL content

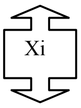
4.1 Logical View

The idea is to represent the correspondences between the various linguistic versions in the form of links in a central structure. These links can be numbers of sentences in the case of a simple local structure such as a large XML file, which includes all the data, the URLs of XML and DTD files representing the versions of each language. It is to some extent a question of following the life cycle of each version, of conserving a complete history of the modifications and applying thereafter the list of the modifications made to the parallel versions to keep alignment.

When a new revision is created, it is necessary to keep a trace identifying the reason for this modification. Moreover, information to be annotated on the object to be replaced in the document is predefined: author, date of operation, optional comment describing the cause of operation.

In what follows, we propose a representation of the correspondence between the linguistic versions which highlights the dependence of the data.

In the figure,



indicates a correspondence link

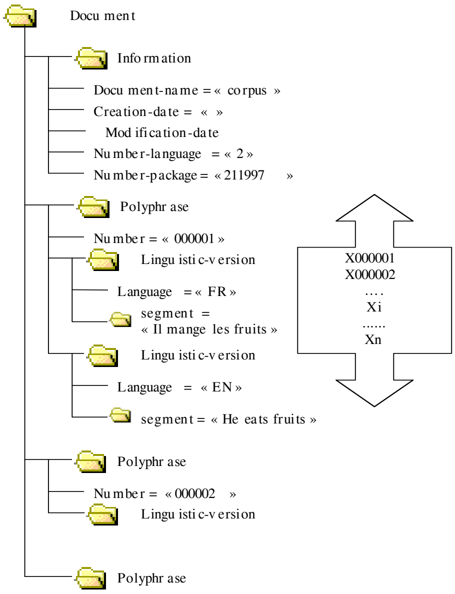


Figure 7: tree XML and representation of correspondences.

The XML tree conforms to the MLD.dtd (Multilingual Language Document). Xn represents a link between the linguistic versions.

For example, X000001 is a link between the French version « Il mange les fruits » and the English version "He eats fruits" constituting the first polyphrase.

We store the set of these links in the XML file, as well as the history of the modifications made to each version.

4.2 Physical view

Data will be stored on a central server in two ways:

- A “Postgres” data base
- File descriptors written in XML, and conforming to a certain DTD, by default our MLD.dtd (Multilingual Language Documents).

The data stored in the database comprises all that relates to the effective management of XML files and access rights on the server. Data tables gather the following information:

- Correspondence between the linguistic versions, their files descriptors in XML, and their DTD.
- URLs of various files (XML, DTD) in order to allow searching and handling them on the server.
- Total information on the level of a version for managing it, and for checking access rights: name, version, author, creation date, planning date in the case of versions under development, translation system used, and some comments.
- Access and modification rights (import and export).
- A version can have two states:
- Private Version : the version is stored on the user workstation, this version can be reloaded and modified.
- Published Version : the version is stored on the server. It results from the decision to publish a Private Version.

4.3 A First Mockup (TraCorpEx project)

After having studied possible architectures and data structures, we have started practical experiments in the framework of the TraCorpEx project. Two parallel corpora in Japanese-English are available [3]. The first comprises 162000 sentences from the CSTAR project and the second 214000 sentences from the PAPILLON project.

To easily manage these corpora using XML, we defined a DTD, MLD.dtd, corresponding to the general structure of multilingual documents. MLD (Multilingual Language Documents) is evolutionary and allows to add other languages to these corpora.

4.3.1 MLD (MultiLingual Documents)

A polyphrase is the set of linguistic versions of the same segment, which have one attribute in common, a unique number. They are also identifiable by other attributes: the language, and for each language the version of the content. In these corpora, the level of alignment is the sentence, but it can go down to a finer level of segments and words. In other corpora, we might go up to the level of paragraph, if sentences are not perfectly aligned.

4.3.2 Interfaces

At this point, the storage format adopted in TraCorpEx is an XML file, which respects MLD.dtd. Upper levels concern the division into corpora, then into sections (import files), then into sentences. Further levels give a hierarchical structure to a polyphrase: language, original and versions, distances, administrative information for tracing etc. At each level, some information is encoded as XML attributes.

```
<!ELEMENT document (information, polyphrase*) >
<!ELEMENT information (#PCDATA) >
<!ATTLIST information   document-name   CDATA
#REQUIRED>
<!ATTLIST information   creation-date   CDATA
#IMPLIED>
<!ATTLIST information   modification-date   CDATA
#IMPLIED>
<!ATTLIST information   number-language   CDATA
#IMPLIED>
<!ATTLIST information   number-polyphrase   CDATA
#IMPLIED>

<!ELEMENT polyphrase (linguistic-version*) >
<!ATTLIST polyphrase   number   CDATA
#REQUIRED>

<!ELEMENT  linguistic-version(segment) >
<!ATTLIST  linguistic-version  language   CDATA
#REQUIRED>
<!ELEMENT  segment(#PCDATA) >
```

This DTD respects the tree structure of the corpora, as well as the dependencies which arise from the translation process, as we go down the tree towards the contents.

It describes a format for multilingual, multiversion documents with m languages and n versions, n>m, and represents at the same time the correspondences between the parallel parts.

A multilingual document is a set of organizational information (name of the document, creation date, last modification date, numbers of languages, numbers of polyphrase) plus a set of polyphrases.

Figure 8 : MLD (MultiLingual Documents)

To add French to these corpora, we have begun to use the commercial MT system Systran-Pro/EF and to revise the results. We plan to run other MT systems and to choose automatically the "best" translation using the distances between the retrotranslations and the orignal English. In case of conflict, we will also use distances between the translations, to group them, and between translations and original, to detect those with more unknown words, left untranslated.

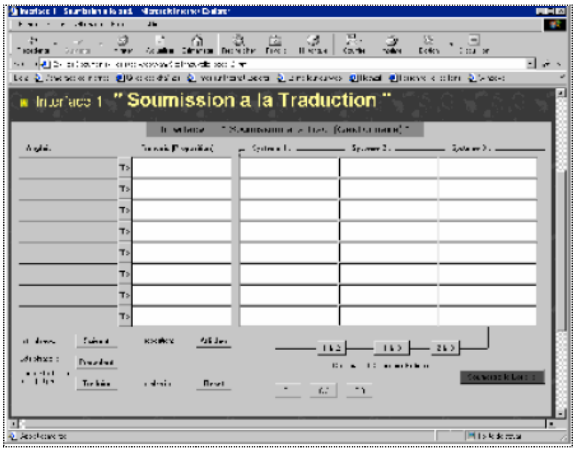
Last but not least, further elaboration, again using string distances, will provide various feedbacks to the developers of the MT systems thus used.

A third interface will be built for the preparation of feedbacks to the developers of the MT systems used. It will allow to calculate and validate the words unknown or badly translated by each system, and to provide translation suggestions from “reference” translations obtained after human revision. It will also provide comparisons between the various systems used, always thanks to the computation of distances at the level of the characters or words.

It also computes distances between English original sentences, so that the document can be used as a translation memory in the following step.

5 Conclusion

The proposed structure of multilingual multiversion documents is technically flexible and modifiable on the initiative of the administrator. It is declared in a hierarchical way in the form of an XML DTD and can be tailored to each corpus of multilingual structured documents aligned at the level of sentences. The hope is that it can contribute to the standardization of multilingual documents, needed to facilitate their management and evolution.

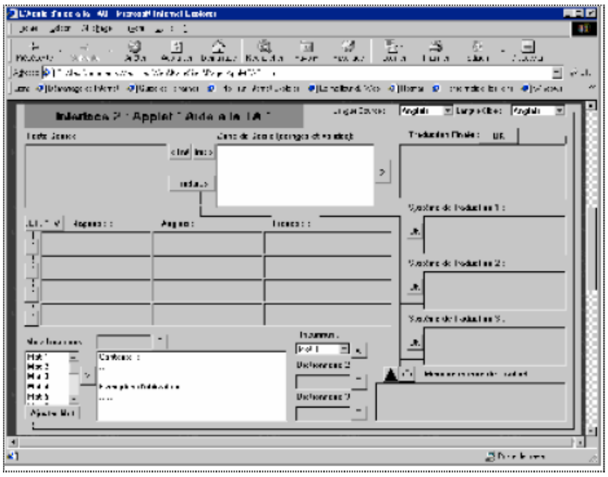


A java program has been developed to calculate the distance between two character strings and to post the result in the form of a matrix and an XML file directly presentable in Word “Track changes” format.

Prototypes of two interfaces have also been produced.

The “preparation” interface allows to submit the English sentences to two or three EF MT systems and to compute the “best” translation of each sentence.

Figure 9 : Interface 1 “preparation”


Interface 2 " Applet " Aide a la TA "
Texte Source
Traduction Finale :
Système de traduction 1 :
Système de traduction 2 :
Système de traduction 3 :
Contexte :
Dictionnaire 2
Dictionnaire 3
Ajouter Mot

The second interface is for human revision of the best suggestion using an English zone: we can correct words or expressions and use the translation memory which is in this case the multilingual document itself.

Figure 10 : interface 2 “revision”

The approach presented here is quite flexible and allows any description of file and directory by XML tags, for multiple applications, among which multilingual information retrieval, multilingual summary, multilingual categorization and of course all types of translation.

References

Al-Assimi A.-B. (2000) *Gestion de l'évolution non centralisée de documents parallèles multilingues.* Nouvelle thèse, UJF, Grenoble, 31/10/00, 200 p.

Al-Assimi A.-B. & Boitet C. (2001) *Management of Non-Centralized Evolution of Parallel Multilingual Documents.* Proc. Internationalization Track, 10th International World Wide Web Conference, Hong Kong, May 1-5, 2001, 7 p.

Blanc É. & Sérasset G. (2001) *From Graph to Tree: Processing UNL Graphs using an Existing MT System.* Proc. The first UNL open Conference, Suzhou, China, 22-26 November 2001, UNDL.

Boguslavsky I., Frid N., Iomdin L., Kreidlin L., Sagalova I. & Sizov V. (2000) *Creating a Universal Networking Language Module within an Advanced NLP System.* Proc. COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL & Morgan Kaufmann, H. Uszkoreit ed., vol. 1/2, pp. 83-89.

Boitet C. & Tsai W.-J. (2002) *Coedition to share text revision across languages.* Proc. COLING-02 WS on MT, Taipeh, 1/9/2002, 8 p.

Boitet C. (2003) *Approaches to enlarge bilingual corpora of example sentences to more languages.* Proc. Papillon-03 seminar, Hookaido university, Sapporo, 3-5 July 2003, 13 p.

Hajlaoui N. (2002) *Gestion des versions des composants éléctroniques virtuels.* Rapport de DEA, CSI, INPG, juin 2002, 80 p.

Sérasset G. & Boitet C. (1999) *UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction.* Proc. MT Summit VII, Singapore, 13-17 September 1999, Asia Pacific Ass. for MT, J.-I. Tsujii ed., pp. 220—228.

Sérasset G. & Boitet C. (2000) *On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter.* Proc. COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL, H. Uszkoreit ed., 7 p.

Tomokiyo M., Al-Assimi A.-B. & Boitet C. (2001) *Multilingual documents management by using Universal Networking Language UNL on Alignment Gestion Tool OGA.* Proc. PACLING'01, Fukuoka, 11-14/9/2001, H. Sakaki ed., 7 p.

Tsai W.-J. (2001) *SWIIVRE- a web site for the Initiation, Information, Validation, Research and Experimentation on UNL (Universal Networking Language).* Proc. First UNL Open Conference - Building Global Knowledge with UNL, Suzhou, China, 18-20 Nov. 2001, GETA, CLIPS, IMAG, G. UNDL ed., 8 p.

Emmanuel Planas Ph.D. Thesis. http://bibliotheque.imag.fr/theses/1998/Planas.-Emmanuel/these.dir/

Multilingual corpora (ITC, Italy). http://tcc.itc.it/people/forner/multilingualcorpora.html

UNL project (Universal Networking Langage). http://www.undl.org/

W3-TR/REC, www.w3.org/TR/REC-xml.

UCL—Universal Communication Language

Carlos A. Estombelo-Montesco and Dilvan A. Moreira

Universidade de São Paulo,
Instituto de Ciências Matemáticas e de Computação
Av. do Trabalhador São-Carlense, 400 - Centro - Cx. Postal 668
São Carlos - SP - Brazil CEP 13560-970
c_estombelo@yahoo.com, dilvan@computer.org

Abstract. For successful cooperation to occur between agents they have to be able to communicate among themselves. To enable this communication an Agent Communication Language (ACL) is required. Messages coded in an ACL should adequately express their meaning from a semantic point of view. The Universal Communication Language (UCL) can fulfill the role of an ACL and, at the same time, be convertible to and from a natural language. UCL design is concerned with the description of message structures, their underlining semantic context and the support for protocols for agent interaction. The key point about UCL is that the language can be used not only for communication among software agents but among humans too. This is possible because UCL is derived from the Universal Network Language (UNL), a language created to allow communication among people using different languages. UCL was defined using the Extended Markup Language (XML) to make it easier to integrate into the Internet. In addition, an enconverter-deconverter software prototype was written to serve as a tool for testing and experimenting with the language specifications.

1 Introduction

The technology of software agents can be an interesting tool for the creation of new models for complex software systems. In the project of software agents, many of the traditional techniques of artificial intelligence can be mixed with techniques from the field of distributed computer systems, theories about negotiation and theories about working teams [2]. Software agents are basically designed to cooperate (either with others or with humans) in a seemingly intelligent way. But for cooperation to occur a communication language is necessary.

What does it mean to be able to communicate with someone? Simplifying it, useful communication requires shared knowledge. While this includes knowledge of language, words and syntactic structures, meaningful communication is even more focused on knowledge about a problem to be solved. To interact with a florist you need some knowledge about flowers.

The widespread use of the Word Wide Web (WWW) and growing Internet facilities have sparked enormous interest in improving the way people communicate using computers. To date, communication among software agents and

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 326–336.

humans has been done under limited conditions: communication is reduced to basic information exchange, ignoring the richness and flexibility implied by human language.

However to deal with any human language would be very difficult. To solve this problem, communication systems can use an Agent Communication Language (ACL) based in a simplified form of human language, which could be converted from and to natural language.

2 Objectives

The main objective of this work is the specification of a new ACL, called UCL—Universal Communication Language, that focus on the specification of the semantic model and structure of the messages it represents. It also adds support for message transmission over the Internet and can be translated into or generated from natural language (English or other language).

UCL is derived from the Universal Network Language (UNL) [6] and implemented using the language XML (Extensible Markup Language) [1]. XML is a W3C (World Wide Web Consortium) standard language, like HTML, this means an easy integration with the Internet.

Another goal of this paper is to show a working UCL enconverter-deconverter prototype using the tool *Thought Treasure* and its associated ontology.

3 Communication Among Agents

In the communication process among agents, it is indispensable an appropriate understanding of what will be communicated through the exchange of messages. A good representation of the knowledge domain, shared by the agents, can collaborate for a better understanding of the context where a message exchange takes place. As a consequence, it is important to explore concept classifications and their hierarchical structures for knowledge domain representation. The concepts in the knowledge domain have to be shared by the agents exchanging messages and be reusable in more than one context.

The specification of an ACL has to deal with the description of the message structure, his semantic model and the interaction protocols [4]:

- The message format defines the communicative acts primitives and the parameters of the message (as sender, receiver, etc.). The message content describes facts, actions, or objects in a content language (KIF, Prolog, etc).
- The semantic model of an ACL should allow for messages with a concise meaning and no ambiguity.
- The interaction protocols are projected to facilitate the communication among agents. Protocols are optional, but, in case they are used, the communication among agents should be consistent with the chosen protocols.

3.1 Ontologies for Communication

'Ontology' is a term used to refer to the common sense of some domain of interest. The ontology can be used as a uniform framework to solve communication problems.

An ontology necessarily links or includes some type of "general vision" regarding a certain domain. This "general vision" is frequently conceived as a group of concepts (for example: entities, attributes, processes), their definitions and their interrelations. That is called a conceptualization.

A conceptualization can be concretely implemented, for example, in a software component, or it can remain abstract, being the implied concepts of a person. The use adopted in this work for ontology is that it is an explicit idea, or a representation (of some part) of a conceptualization.

An explicit ontology can take a variety of forms, but necessarily they will include a vocabulary of terms and some specification of their meanings (for example: definitions).

The level of formality for a vocabulary varies considerably. This variation can be shown in the following four points of view:

- Highly informal: expressed freely in natural language.
- Semi-informal: expressed in a restricted form and structure in natural language. Better clarity for ambiguity reduction.
- Semi-formal: expressed in an artificial language defined formally.
- Strictly formal: defined meticulously with formal semantics, theorems and proofs.

A shared ontology is necessary for communication between two agents. Unfortunately UNL does not have a public available ontology. For this reason, the ontology embedded in the tool *Thought Treasure* was used to implement the enconverter-deconverter prototype.

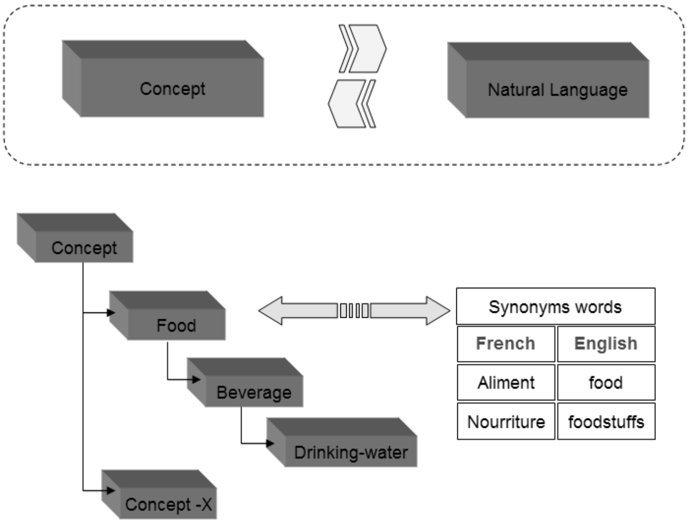
3.2 The Tool *Thought Treasure* (TT)

Thought Treasure (TT) is is a powerful tool for processing natural language, developed by Erik T. Mueller [5]. It is capable of interpreting natural language, as well as extending its ontology-based knowledge base. TT has a compiler for natural language that allows it to extract information of sentences.

TT has a database with 25,000 concepts organized in a hierarchical way. For example, Evian is a flat-water type, which is a drinking-water type, which is a food type and so on.

Each concept has one or more word translations what forms a total of 55,000 words and sentences of the English and French language. For instance, as it is observed in the Fig. 1, the association with the concept food in the English language are the words *food* and *foodstuffs* and in French *aliment* and *nourriture* (among others).

In addition, TT has approximately 50,000 assertions related to concepts such as: a green-pea is a seed-vegetable, a green-pea is green, the green-pea is part of pod-of-peas, and pod-of-peas is found usually at a store of foodstuffs.



Synonyms words	
French	English
Aliment	food
Nourriture	foodstuffs

Fig. 1. Association of the ontology with a natural language.

4 UCL—Universal Communication Language

The language UCL represents information in the same way UNL does, but using syntax based in XML. XML is a meta-language, a simplified form of SGML, which developers can use to create new languages based in tag elements. The new tags, created to represent the new language elements, can be described in a special file called DTD (Document Type Definition). UNL is a formal language for representing the meaning of natural language sentences and exchange information over a network. Information that is written in a native natural language is "enconverted" into UNL and stored in a server. This information can be "deconverted" into other languages to be read by each native reader. Thus, UNL can play the role of an interface between different human languages to exchange information.

UNL represents information expressed in sentences as a set of relations between meanings, expressed by words, and a syntactic structure that makes up the sentence. The vocabulary of UNL consists of:

- Universal Words (UWs), to represent word meaning.
- Relation Labels, to represent relationships between UWs
- Attribute Labels, to express further definitions or additional information for the UWs that appear in a sentence.

In UNL, the information about a sentence includes its meaning, tense and aspect information (how the speaker grasp the event), intention of utterance, speaker's feeling or judgment upon contents, and sentence structure. In the language, the meaning of a sentence is represented by the description of the relationships between UWs and its structure is described by attaching attribute labels to these UWs.

4.1 UCL Goals

The language UCL is to be used for high-level communication among agents through the exchange of messages. Some characteristics that guided the definition of the language were:

- To aid the communication involving agents giving importance to the semantics of the message;
- To be easy to use;
- To facilitate its integration into the Internet environment writing it in XML.

The language UCL represents the information in sentences (that can form messages) that involves a syntactic structure with a group of concepts, relationships and attributes similar to UNL:

- Universal Words (UW),
- Relationship labels,
- Attribute labels.

To define a language based in XML a specific DTD file is used. This DTD is essentially a free context grammar, like the extended BNF form (*Backus Naur Form*) used to describe computer languages [3].

As in UNL, a Universal Word (UW) is the minimum unit that represents a concept, which denotes a specific meaning in a message. When a concept needs to be defined in more detail Relationship Labels and Attribute Labels are used. In addition, UCL uses a shared ontology, from the TT tool, to add meaning to the UWs. All agents participating in a communication process should share this ontology.

In a UCL sentence, each defined UW has an identifier label (id) that is used to identify a particular concept inside a sentence. A sequence of alphanumeric characters forms this labels. The label head corresponds to the place where the name of the concept will be defined. The concepts used are always related to the ontology being used (TT ontology). It is at this point that a sentence in UCL is connected to the ontology for a specify knowledge domain.

In UCL, messages possess a certain meaning involving concepts. This composition of concepts is represented by groups of binary relationships, which allow different relationships involving the concepts. The relationship labels used come from UNL. Figure 2 shows an English sentence and its translation to UCL.

5 Enconverter-Deconverter Implementation

UCL is defined in the meta-language XML, to work with it a XML parser should be used. As the enconverter-deconverter is written in the language Java, the Java

Sentence: UNL is a common language that would be used for network communications.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sentence SYSTEM "Sentence.dtd">
    <sentence>
        <uw id="uw00" head="language">
            <icl direction="to"> <uw head="abstract thing"/> </icl>
            <tense attribute="present"/>
            <focus attribute="entry"/>
        </uw>
        <uw id="uw01" head="UNL">
            <icl direction="to"> <uw head="language"/> </icl>
            <focus attribute="topic"/>
        </uw>
        <uw id="uw02" head="common">
            <aoj direction="to"> <uw head="thing"/> </aoj>
        </uw>
        <uw id="uw03" head="use">
            <icl direction="to"> <uw head="do"/> </icl>
            <tense attribute="present"/>
        </uw>
        <uw id="uw04" head="language">
            <icl direction="to"> <uw head="abstract thing"/> </icl>
            <tense attribute="present"/>
            <focus attribute="entry"/>
        </uw>
        <uw id="uw05" head="communication">
            <icl direction="to"> <uw head="action"/> </icl>
            <convention attribute="pl"/>
        </uw>
        <uw id="uw06" head="network">
            <icl direction="to"> <uw head="thing"/> </icl>
        </uw>
        <relation label="aoj" uw-id1="uw00" uw-id2="uw01"/>
        <relation label="mod" uw-id1="uw00" uw-id2="uw02"/>
        <relation label="obj" uw-id1="uw03" uw-id2="uw04"/>
        <relation label="pur" uw-id1="uw03" uw-id2="uw05"/>
        <relation label="mod" uw-id1="uw05" uw-id2="uw06"/>
    </sentence>
```

Fig. 2. Definition of a UNL sentence.

API for XML Processing (JAXP) Version 1.1 from Sun, was used (other Java XML parsers could have been used).

As said before, UCL uses the ontology available on the TT tool (written in C). This tool includes program libraries to manipulate concepts of the ontology, to do consultations on the concepts network, and to analyze their hierarchy. An instance of TT can run as a server in a network and communicate with a Java program running in another process. A Java communication API is supplied with TT to handle the low level details of this communication.

The enconverter-deconverter prototype uses the Java communication API to contact a running instance of TT and use its functionality. Those include natural language treatment, ontology queries, etc. A high level Java interface was written to communicate with the TT server (through the API) and implement the high level functions needed by the prototype. This interface is called *UclLanguage*.

Figure 3 presents a diagram with the sequence of events that happens when the prototype makes use of the interface *UclLanguage* to generate UCL messages.

The process begins when a user enters a natural language sentence into the prototype. The prototype calls the method *understood* of the interface *UclLanguage*. The natural language sentence is interpreted (using TT) and some possible semantic interpretations are returned. The user chooses the most appropriate interpretation. The chosen interpretation is converted to TT format (method *takeAttofConcept*) and then to UCL format (method *convertTTtoUCLwrite*). The UCL format can be shown on the screen or saved in a file.

The reverse process, to transform a UCL message in natural language is easier. The prototype uses the method *deconvertUCLtoTT* to convert the UCL message in a list of TT concepts. Then it uses the method *deconverterTTtoLN* to transform this list of concepts in a natural language sentence, which represents the original UCL message. Figure 4 shows the prototype converting a sentence to UCL and then back to English (and French).

Figure 5 illustrates the use of UCL (using one TT server) in the communication process between two software agents.

6 Conclusions

The definition of the Universal Communication Language (UCL) includes all theoretical concepts of the Universal Networking Language (UNL). This was done to preserve the representative power of this language. The Web community currently regards XML as an important step toward semantic integration. Developing the language UCL using XML yielded some important benefits. The most important is the reuse of existing tools for creating, transforming, and parsing UCL documents.

The UCL enconverter-deconverter prototype shows the need for a shared ontology for the implementation of a successful enconverter-deconverter. UCL was developed to be used as a rich Agent Communication Language (ACL), which would make it easier for humans to communicate with and program software

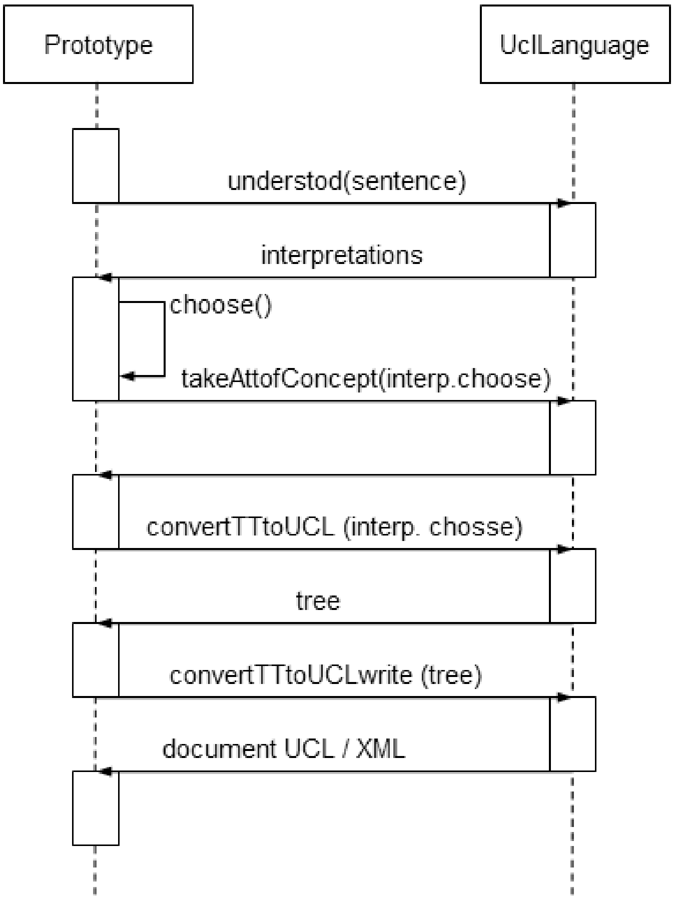

Prototype
UclLanguage
understod(sentence)
interpretations
choose()
takeAttofConcept(interp.choose)
convertTTtoUCL (interp. chosse)
tree
convertTTtoUCLwrite (tree)
document UCL / XML

Fig. 3. Diagram with the sequence of events during enconvertion.

Example: Monkey eats bananas

```
======= Input Natural Language ==========
Example: Monkey eats bananas.

============ Choose Option ==============
<0>An ape eats a banana.

Option: 0
============ Message UCL  ===============
<?xml version="1.0" encoding="UTF-8"?>
<sentence>
    <uw id="uw2" head="present-indicative">
        <icl direction="to">
            <uw head="present-tense" />
        </icl>
        <focus attribute="entry" />
    </uw>
    <uw id="uw4" head="eat">
        <icl direction="to">
            <uw head="ingest" />
        </icl>
    </uw>
    <uw id="uw5" head="ape">
        <icl direction="to">
            <uw head="mammal" />
        </icl>
    </uw>
    <uw id="uw7" head="banana">
        <icl direction="to">
            <uw head="fruit-tropical" />
        </icl>
    </uw>
    <relation id="uw1" label="icl" id1="uw2" id2="uw6" />
    <relation id="uw6" label="icl" id1="uw3" id2="uw7" />
    <relation id="uw3" label="agt" id1="uw4" id2="uw5" />
</sentence>

======== Deconverter Message UCL  ===========
=>Debug : [present-indicative [eat ape banana ]]

English: An ape eats a banana.
French : Un singe croque la banane.
```

Fig. 4. Prototype converting a sentence to UCL and back to English.

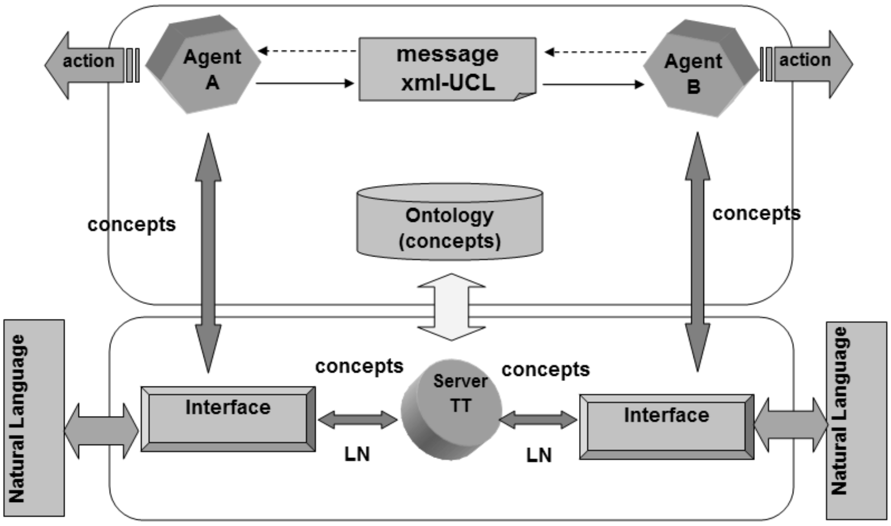


Fig. 5. Architecture of a system that uses the UCL language.

agents (using multiple natural languages). But UCL can be used in the same role as UNL.

The prototype also points out the need for an open shared ontology for UNL. UNL relation and attribute labels have some ontological knowledge already embedded in them. This makes impossible to map all possible UNL (and consequently UCL) constructs into *Thought Treasure* (TT) ontology based representation. The prototype cannot be expanded into a full featured UCL enconverter-deconverter. For the time being this prototype is good enough to help the development of a prototype UCL interpreter for software agents.

The full power, of the approaching of using UCL as an ACL and programming tool for software agents, will only be realized when an open shared ontology for UNL and enconverters-deconverters, for many natural languages (using this shared ontology), are available. One will be able to program a software agent using his own native language and share this program with many other people, which will see and interact with the program in their own native languages.

Finally, UCL is still a proposal, but we hope that others in the Web community will help to shape its final format.

7 Acknowledgements

The authors would like to thank the CNPq—Brazilian National Council for Research for the financial support for this work.

References

1. Connolly, D.: *Extensible Markup Language (XML).* February (2000). Available on-line: http://www.w3.org/XML/
2. Dignum, F., Greaves, M. (ed.): *Issues in Agent Communication.* (Lecture notes in computer science; Vol 1916: Lecture notes in artificial intelligence) Berlin: Springer, (2000).
3. Grosof, B. N.; Labrou, Y.: *An Approach to using XML and a Rule-based Content Language with an Agent Communication Language.* IBM Research Report. RC 21491 (96965), 28 May (1999). Available on-line: http://www.research.ibm.com
4. Mamadou, T. K., Shimazu, A., Tatsuo, N.: *The State of the Art in Agent Communication Languages.* Japan Advanced Institute of Science and Technology, Japan, (2000).
5. Mueller, E.T.: *Natural Language processing with ThoughtTreasure.* New York: Signiform (1998). Also available on-line: http://www.signiform.com/tt/book/
6. Ushida, H., Zhu, M., Senta, T.D.: *The UNL a Gift for a Millennium.* UNU/IAS, ISBN:4-906686-06-0, November (1999). Also available on-line: http://www.unl.ias.unu.edu/publications/index.htm

Knowledge Engineering Suite: A Tool to Create Ontologies for Automatic Knowledge Representation in Intelligent Systems

Tânia C. D. Bueno,[1] Hugo C. Hoeschl,[1] Andre Bortolon,[1] Eduardo S. Mattos,[1]
Cristina Santos,[1] Ricardo M. Barcia [2]

[1] Instituto de Governo Eletrônico, Inteligência Jurídica e Sistemas – IJURIS
Rua Lauro Linhares, 728, sala 105, Florianópolis BRASIL – CEP 88036-002
http://www.ijuris.org
tania@ijuris.org;{hugo, bortolon, mattos, cristina}@wbsa.com.br

[2] Virtual Institute of Advance Studies – VIAS
Florianópolis, BRASIL
rbarcia@uol.com.br

Abstract. The present work is focused on the systematization of a process of knowledge acquisition for its use in intelligent management systems. The result was the construction of a computational structure for use inside the institutions (Intranet) as well as outside them (Internet). This structure was called Knowledge Engineering Suite, an ontological engineering tool to support the construction of ontologies in a collaborative environment and was based on observations made at Semantic Web, UNL (Universal Networking Language) and WordNet. We use both a knowledge representation technique called DCKR to organize knowledge, and psychoanalytic studies, focused mainly on Lacan and his language theory to develop a methodology called Engineering of Mind to improve the synchronicity between knowledge engineers and specialists in a particular knowledge domain.

1 Introduction

The importance of the Knowledge Based Systems is in the fact that they provide the computer with some peculiar characteristics of human intelligence, such as the capacity to understand natural language and simulate reasoning in uncertainty conditions. Defining the relevant information to be inserted into a Knowledge Based Systems is the great problem in the development of intelligent systems, mainly because the process is basically experimental and depends greatly on the ability of the knowledge engineer. In particular, a great difficulty is related to the definition of the terminology used to nominate the concepts and the relations [1]. Besides the great number of methods to do the knowledge acquisition, we can't find one that deals with the understanding and learning of the people involved, both specialists and knowledge engineers.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 337–346.

More recently, the notion of an ontology is being so popular in fields such intelligent information integration, information retrieval on the Internet, and knowledge management. The reason is in part due to what they promise: a shared and common understanding of some domain that can be communicated across people and computers [2]. Different developments of a worldwide range have a reference in cooperative work as a WordNet, Semantic Web and UNL (Universal Networking Language) through the construction of ontologies using collaborative tools. The use of ontological engineering tools or metatools to support the knowledge engineering process allows the organization of a knowledge base established on the relationship between relevant expressions from a context. Ontologies, as a basis for automatic generation of knowledge acquisition tools, simplify the tool specification process by taking advantage of ontologies defined as part of the knowledge engineering process [3]. Nevertheless, experience shows that often the bottleneck of building sharable ontologies lies more in the social process than in the technology [4]. For this reason, we develop a methodology to the process of knowledge acquisition to allow the specialist and the knowledge engineer to work in synchronicity, in cooperative networked organizations. We call this methodology Engineering of Mind. This synchronization process initiates with the understanding of human intelligence, its unconscious manifestations and its relationship with the words, since, according to Lacan, every human investigation is linked irreversibly in the inner space created by language. In the present development, we create a tool to support the knowledge engineering process by assisting developers in the design and implementation of ontologies in a specific domain.

In earlier works, we use a methodology called DCKR (Dynamically Contextualized Knowledge Representation [5]. DCKR allows the construction of a knowledge base, improving the construction of the domain ontology, and the automatic representation of cases in knowledge-based systems, either in the legal area [6], or in knowledge management domain [7].

In the next section, the methodology for the knowledge synchronization is described. This methodology allowed an exceptional coherence among the semantic relations of what is called ‘indicative expressions’, mainly by the support of all this computational structure during the process. This allowed the knowledge engineer and the specialist to develop much more than the knowledge representation of the domain, but abilities as inherent conscience, discipline, persistence, and empathy.

2 The Knowledge Representation in Knowledge Based Systems

We use a special process to extract and represent the knowledge for the knowledge based systems. The main intention of this process is to allow an automatic process of text indexing, on the basis of a controlled vocabulary and a dictionary of normative terms, constructed through the relevance of the definite terms persuasively, called normative key-terms [8]. Due to the necessity to turn the acquisition process quicker, it was necessary to evolve the process, using IR techniques (Information Retrieval) to associate the relevance of the terms with the frequency of the words added to the controlled vocabulary and the dictionary of normative terms; this approach gave origin to a methodology of knowledge representation called DCKR - Dynamically Contextualized Knowledge Representation [9]. DCKR is a methodology of representation of knowledge whose approach is centered in a dynamic process acquisition of the knowledge of texts, defined through elaboration of a controlled vocabulary and a dictionary of terms, associated to an analysis of frequency of the words and indicative expressions of the context.

UNL, Semantic Web and Wordnet

In the knowledge acquisition for elaboration of the knowledge base of intelligent systems we chose the use of methodologies that use web environments and cooperative development. Today, there are three great worldwide developments that use the Internet for the development of ontologies, the UNL, Semantic Web and Wordnet.

UNL (Universal Networking Language) [10] is a language for computers to share information through a net. It is meant for representing the natural, independent language of its language, so that computers process the text and represent it in different languages.

The WordNet [11] is a lexicon reference system whose design is inspired in psycholinguistic theories on the human lexical memory. The nouns, verbs, adjectives and adverbs of the English language are classified only, being organized in sets of synonyms, each one representing a lexical concept. The sets of synonyms are through relations different to each other.

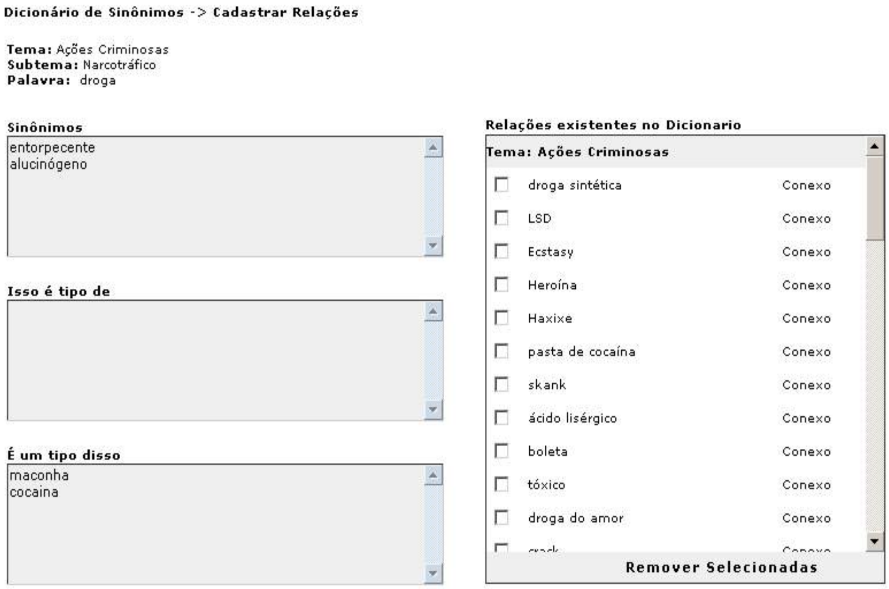
The Semantic Web [12] is an extension of the current Web, in which the information has a very well defined meaning, allowing the computers to process the information contained in the pages and to understand it, executing operations that facilitate the work for the users.

The three initiatives are meant to facilitate the automatic processing of the information contained in documents, allowing the computers to execute more intelligent operations and to retrieve information in more efficient way.

3 The Knowledge Engineering Suite

The Knowledge Engineering Suite is an Ontological Engineering Tool for collaborative networked works on the Web. Built to facilitate the knowledge sharing between the knowledge engineering team and the specialist team. The Suite allows to build relationships between complex terms, considering its concepts in the specific domain of the application. These relations are based on AI techniques [13], theories of language, Semantic Web, WordNet, and UNL.

The creation of an infrastructure for acquisition of the knowledge for cooperative work on the Web is an efficient and effective tool for the acquisition of the knowledge in intelligent systems. Many different techniques of Knowledge Acquisition exist; the Knowledge Engineering Suite (see figure 1) is used with the DCKR methodology. Where tools as the Frequency Extractor, Semantic Extractor and the Knowledge Engineering Suite have been associated to this methodology to assist in the task.



Dicionário de Sinônimos -> Cadastrar Relações

Tema: Ações Criminosas
Subtema: Narcotráfico
Palavra: droga

Sinônimos
entorpecente
alucinógeno

Isso é tipo de

É um tipo disso
maconha
cocaina

Relações existentes no Dicionario

Tema: Ações Criminosas

□	droga sintética	Conexo
□	LSD	Conexo
□	Ecstasy	Conexo
□	Heroína	Conexo
□	Haxixe	Conexo
□	pasta de cocaína	Conexo
□	skank	Conexo
□	ácido lisérgico	Conexo
□	boleta	Conexo
□	tóxico	Conexo
□	droga do amor	Conexo
□	crack	Conexo

Fig. 1. Editing Module - Ontology construction (insertion and consistency checking).

This application works with extractors of automatic standards in contribution with knowledge engineers and specialists in the approached domain as specifications found in methodology DCKR - Dynamically Contextualized Knowledge Representation, which consists of a dynamic process of analysis of the general context that involves a thematic focused. The Suite is an editor of ontologies structured in a form to allow a cooperative work on the Web between the team of knowledge engineering and the team of specialists.

This computational environment of shared access has two main objectives: organization and representation of the knowledge, and update of the Knowledge Base.

Basically, four modules compose it, they are:

1. Register. It allows the elaboration of a contextualized dictionary, for the selection of topics and sub-topics for the classification of the indicative expressions. In this environment the user defines the topic and sub-topic in which it will insert a new indicative expression. A domain can be categorized in innumerable topics and sub-topics;

2. Search. It informs about other terms already registered in the base, which have some phonetic similarity with the term typed. This tool allows the verification of possible typing errors , besides preventing the registration of the same term more than once. It is a search system based on similarity. It supplies the user with a list of similar indicative expressions present in the knowledge base in alphabetical order after consultation made by the user. It is used in the registers, in the edition and the administration module;

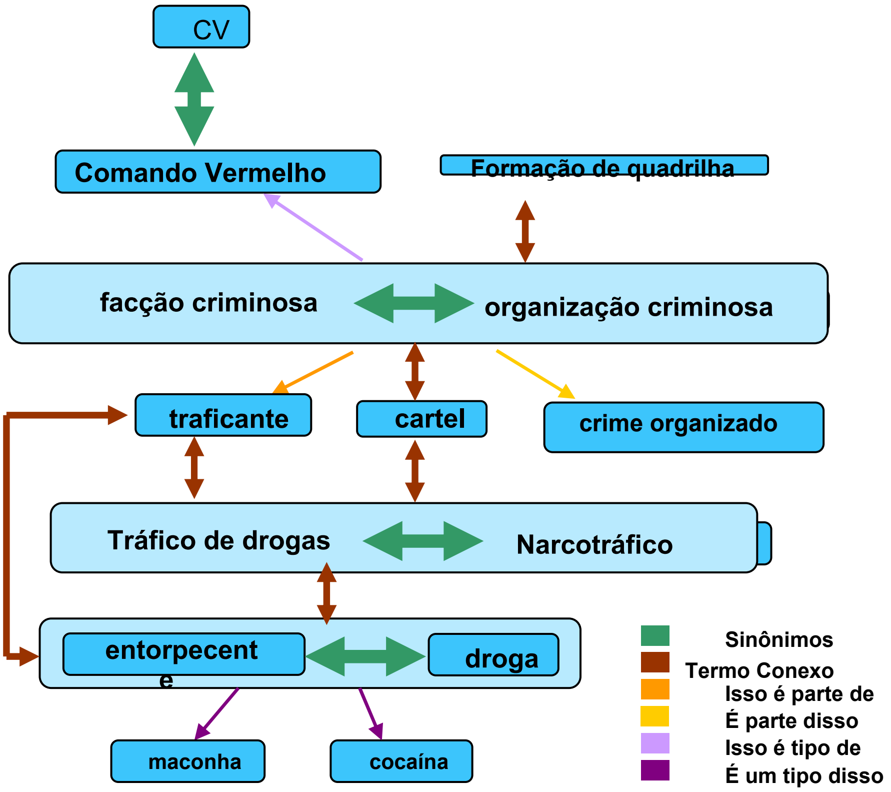


Fig. 2. The semantic relations of indicative expressions.

3. Relationship Editor. It allows the building of the relationship tree, always considering the similarity between all the terms registered and the ones already existing in the base. These relationships allow the system to expand the search context. The organization of the tree allows the dynamic definition of the weights of the indicative expressions according to the entrance of the user. The fields with all the relationships available to be formed are presented. They are the following: *-synonyms*; *-Related terms*; *-This is type of*; *- It is a type of this*; *- This is part of*; *- It is part of this.* The editor presents the existing relationships and allows excluding them (see figure 2). Each relationship has a weight related to the defined indicative expression in the search by the user.

4. Administration Environment. The knowledge integration and the validation between words is made in accordance with the context of themes and subthemes. This topic is organized in three levels: - High Level - it allows to insert themes and subthemes, to validate exclusions, to include and to exclude users, to verify productivity of each user and to verify descriptions of the dictionaries, themes, subthemes and indicative expressions; - Medium level- it allows to verify productivity and historical data; and, Low level- it allows to verify descriptions.

However, all this structure and methodology had not been enough to turn the cooperative work efficient and effective. It was necessary a more holistical approach, which allows a greater coherence between the relations of the expressions, mainly in the elaboration of the related terms where the participation of the specialist is almost exclusive. It is important to highlight that this structure of contextualized ontologies allows an automatic information indexing by the system and an knowledge acquisition that gives more qualitative answers in the retrieval process.

4 Elaborating the synchronicity in a collaborative networked organization

The different unfolding of the human inventivity, even being so diversified, have the same origin, the unconscious mind and the human perceptions; from the fact of distinct constructions eventually to lead the thoughts to one same reference. Because of that, we elaborate a methodology that let the immediate perception of the specialist arise, without the pretension to reach all the knowledge, but with clear objectives, for example, to eliminate the common resistances of the people to the technological innovations, standing out the importance of the management of the human capital. [14].

During the development of tasks of knowledge engineering, it was observed that the efficiency of the acquisition process had a direct relation with good relationship between the knowledge engineer and the specialists of the domain, no matter the quality or content of the interviews, or the efficient application of the support tools. Thus, keeping this relationship in perfect synchrony is a key factor for the success of the system and a challenge for which the stages defined in the present work serve as a model of relative success.

The *Common-sense* tells us that the immediate perception (intuition) has greater effectiveness on the best solution for a problem than the application of rules of the propositional logic. Although, the most accepted proposal is people try to solve deductive problems applying rules such as of the propositional logic. According to Lacan [15], if we consider that the unconscious is structured as a language, it is possible to reconstruct the unconscious associations between the words, thus disclosing, a context.

There are elements, like the cognitive complexity and the capacity to learn, that supplies the underlying individual traces in which the specialized knowledge and abilities are based, similarly the sociability and the confidence supply the anchors to develop and to keep a net of relationships. Thus, identifying that non-cognitive knowledge is also important knowledge of the institutions and, for this reason, they must be part of the capital of these organizations, becomes necessary to look for a way to identify it and to represent it in the knowledge based systems. Therefore, this complex net of communication between the diverse areas of talent that will go to supply necessary flexibility, versatility and adaptability intelligences to happen.

All the languages are structurable as an articulating system. But their character, their coherence is in an articulated system, which is unique. Thus, the cognitive point of view concerns the symbolic acquisitions, those have as a foundation the meanings whose support is, generally, natural language, or, at times, specialized languages, as the formal ones. The attachment of these elementary meanings in a wide team requires synchronous thoughts.

This synchronization process initiates with the understanding of human intelligence, its unconscious manifestations and its relationship with the words, therefore, in accordance with Lacan [16], every human investigation is tied irreversibly in the interior of the space created by the language. But, for the victory of this dynamics of cerebral gymnastics, it is primordial that the person is in a positive attitude. The brain only registers, learns and ramifies when it is open to what is new.

4.1 Engineering of Mind Methodology

There are many different techniques of Knowledge Acquisition. We created the Engineering of Mind (see figure 3) to help developing the following process (DCKR methodology): 1. Inventory of the entire domain (classification of all sources of digital information that will be in the system database). 2. Application of the word frequency extractor based on the database inventoried; 3. Comparison between extractor results with the specialist's needs. 4. Construction of a representative vocabulary of the domain, by the specialist and knowledge engineers. 5. Application of the semantic extractor on the database; using the representative vocabulary (indicative expressions). 6. Definition of a list of words based on evaluation of the result of the frequency of the indicative expressions found in the inventory. 7. Construction of the ontologies in the Knowledge Engineering Suite based on this controlled vocabulary. 8. Definition of synonyms, related terms, homonyms, hyponyms, hypernyms and meronyms.

The acquisitions of the knowledge carried through by the team of engineers of the knowledge, in the area of its specialization [5] [6] got a bigger effectiveness than the acquisition carried through for the same team in diverse dominium of its specialization [7], where some obstacle of communication had taken the necessity of a new implantation of the acquisition process. That is, it did not have synchronization problems, therefore the deep knowledge of the specialists of the area of the technique of AI that was being applied in the system modeling (e.g., Case-Based Reasioning) allowed a transference of knowledge for the computational language of a very positive form for the final target of the systems.

It was observing the elements presented in the two processes that were possible to systemize a series of questions, improving the speed and quality of the knowledge represented in the system.

Associated to these comments, very uncommon procedures to the process of knowledge acquisition had been adopted, such as programming techniques neuro-linguistics and meditation to defragment the emotional memory of the specialist and to facilitate the learning process. This process was due to the existence of the following problems: 1. Resistance to the system; 2. Difficulty to reproduce the process of decision; 3. Little quality of the handled knowledge.

However, the focus object is not the area of application of the system (domain), but the specialist(s) and the knowledge engineer(s) that (will) exactly work in the definition of the target of the system and in the formation of the knowledge base of this system. To identify and to separate knowledge conditions are essential, therefore both (specialists and engineers) will have to learn and to train the learning process and, for this, it will need to surpass the comfort zone. Knowledge Engineering is over all knowledge exchange.

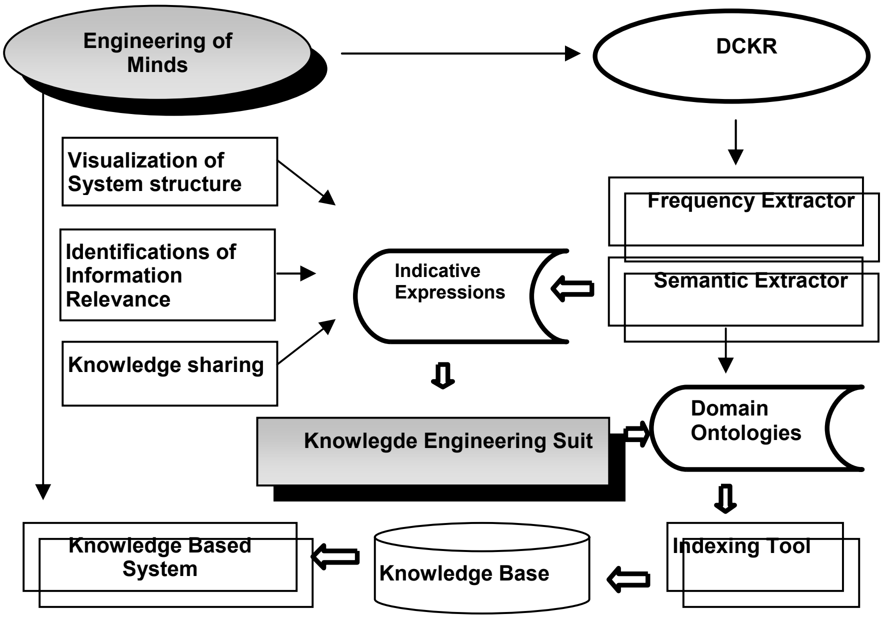


Fig. 3. Engineering of Minds Methodology applied to construction of ontologies in Knowledge Based Systems.

The importance that the existing knowledge has for the new acquisitions comes from the basic role that they play inside of the construction of the representations and of the idea of that the acquisition passes, necessarily, by these representations. This is the importance of this phase, there is an exchange of knowledge, the specialist starts to know the form as its knowledge could be organized, that is, the basic concepts of the technique of used Artificial Intelligence in the representation of the knowledge, thus it will be able to contribute with more effectiveness and will have greater interest in participating in the process. As to the specialist, the exchange will lead to a more immediate perception of the target of the system, and will increase the interest in if going deep the study of the domain. Both will be prepared to deal with this overload and to obtain the ability necessary to plan or to choose a perspective that it determines then that elements of the situation must be treated as important and which can be ignored. When perceiving that of the vast information, the knowledge if only restricts to some of the number of characteristics and possibly excellent aspects, to take a decision one turns easier.

The continuous sharing of the established visions becomes the specialists and engineers better to work in cooperation in the construction of the ontologies of the domain. This productive process is continuous and can establish changes in elapsing of the implantation of the system.

5 Conclusions

The systematization and organization of teams of domain specialists together with the team of Knowledge Engineering started to be the great challenge in the system development of knowledge management. The cooperative work between the teams does not only need the deep knowledge on the domain application to the system, but also on the organization of its knowledge base. The creation of a computational environment in the Web allowed to a greater sharing of information and better results among the teams in the construction of knowledge based systems.

The Knowledge Engineering Suite enables a cooperative work among people in different places, structuring a continuous knowledge base and easy visualization (knowledge tree), through relationship nets and supplied an exceptional coherence among the semantic relations of what is called 'indicative expressions', mainly by the support of all this computational structure during the process. This allowed the knowledge engineer and the specialist to develop much more than the knowledge of the domain, but abilities as proper conscience, disciplines, persistence, and empathy.

Acknowledgements The authors would like to thank the support of the WBSA – Sistemas Inteligentes S/A. Special thanks to researchers and students of the Juridical Institute of Intelligence and Systems - IJURIS: Aline Junckes and Valdir Luiz Junior, who contributed to our research.

References

1. Resende, Solange Oliveira. Sistemas Inteligentes: fundamentos e aplicações. Barueri, SP: Manole,2003.
2. Duineveld, A. J. et al, 1999. WonderTools? A comparative study of ontological engineering tools. *Twelfth Workshop on Knowledge Acquisition, Modeling and Management.*Voyager Inn, Banff, Alberta, Canada.
3. Eriksson, H. et al, 1999. Automatic Generation of Ontology Editors. *Twelfth Workshop on Knowledge Acquisition, Modeling and Management.*Voyager Inn, Banff, Alberta, Canada.
4. Benjamins, V.R., 1998. The ontological engineering initiative $(KA)^2$, *Formal Ontology in Information systems*. IOS Press, Amsterdam.
5. Hoeschl, Hugo. C. Hoeschl, Bueno, Tania. C. D., Barcia, Ricardo. M., Bortolon, Andre., Mattos, Eduardo Da Silva. Olimpo: Contextual structured search you improve the representation council of UN security with information extraction methods In: å. Artificial International conference on inteligence and law, 2001, St. Louis. ICAIL 2001 Proceedings. New York: ACM SIGART, 2001, p.217 – 218.
6. Bueno, Tânia Cristina D'Agostini. O Uso da Teoria Jurídica para Recuperação em Amplas Bases de Textos Jurídicos. 1999. 94 f. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina, Florianópolis, 1999.
7. Ribeiro, Marcelo Stopanovski. KMAI, da RC²D à PCE. Gestão do conhecimento com inteligência artificial, da representação do conhecimento contextualizado dinamicamente à pesquisa contextual estruturada. [2004]. Dissertação (Mestrado em Engenharia de Produção) – Universidade Federal de Santa Catarina, Florianópolis, 2003.
8. Bueno, Tânia C. D. et al, 1999. JurisConsulto: Retrieval in Jurisprudencial Text Bases using Juridical Terminology. *Proceedings of the Seventh International Conference On Artificial Intelligence And Law*. ACM, New York.

9. Hoeschl, Hugo. C. et al, 2003. Structured Contextual Search For The Un Security Council. *Proceedings of the fifth International Conference On Enterprise Information Systems*. Anger, France, v.2. p.100 – 107.
10. UNL. Universal Networking Language. Available in: www.unl.ias.unu.edu/unlsys/index.html. Access in: 19 jan. 2004.
11. WORDNET. Available in: http://www.cogsci.princeton.edu/~wn/. Access in: 19 jan. 2004.
12. Semantic Web. Available in: http://www.w3.org/2001/sw/. Access in: 19 jan. 2004.
13. Kolodner, J. Case-Based Reasoning. Morgan Kaufmann, Los High, CA. 1993.
14. Gratton, Lynda, Ghoshal, Sumantra. Managing Personal Capital Human: new ethos will be the "Volunteer" Employee, The European Management Journal, vol 21, n° 1 pp1-10, February, 2003.
15. Lacan, Jacques. Os seminários de Lacan. Disponível em CD Room, 2000.
16. Miller Jacques-Alain, 1988. *Percurso de Lacan: uma introdução*. Jorge Zahar Editor Ltda, 2ª edição, Rio de Janeiro.

Using Semantic Information to Improve Case Retrieval in Case-Based Reasoning Systems

J. Akshay Iyer and Pushpak Bhattacharyya

Indian Institute of Technology Bombay
{akshay,pb}@cse.iitb.ac.in

Abstract. Conventional Case-Based Reasoning (CBR) systems rely on word knowledge to index and search cases from its memory. On being presented with a problem, the Case-Based Reasoning system tries to retrieve a relevant case based on the words that appear in the problem sentence without considering their respective senses. Drawbacks of such systems become more evident in cases where the input is in the form of a sentence in a natural language. Ignoring semantic information in this case may not result in retrieval of desired case or may result in retrieval of an undesired case. In this paper we present a method that tries to improve the precision of retrieval by also taking into account semantic information available to us about the words in the problem sentence. Towards this goal, Universal Networking Language (UNL) is made use of, which provides a semantic representation of natural language text to capture sentence structure. Lexical resource like WordNet is used for finding semantic similarity between two concepts. Using an existing commercial Case-Based Reasoning system as basis for comparison, we demonstrate that considering such semantic information helps in improving case retrieval.

1 Introduction

Case-Based Reasoning (CBR) Systems are one of the most widely used systems in the field of problem solving and planning. A number of such systems are developed and reported [5, 8, 11]. Typically, a number of cases are stored in memory and upon being presented with a problem, a set of relevant cases is retrieved and presented as a solution to the problem [7]. One of the fundamental issues in such systems concerns this retrieval process. Information from the input problem is extracted out and this information is used to index (or search) in the memory to locate the desired case. In systems where a problem is input in Natural Language form, the issue becomes more profound. Traditionally, a number of statistical methods are used for extracting information from the input problem and using it in turn for identifying cases that are relevant to the problem. However, since such methods do not employ any natural language understanding, they fail in situations when mere knowledge about words is not sufficient.

In this paper we propose a method by which we could use information, both semantic and syntactic, from natural language text to compare and retrieve relevant cases. The rest of the paper is organized as follows. Section 2 discusses the shortcom-

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 347–358.

ings of traditional methods and sets out the motivation for using sentence structure and semantic knowledge in case-searches. In order to capture sentence structure we propose to use Universal Networking Language (UNL) [4], whose introduction and generation process are given in section 3 and 4 respectively. In section 5 we present our algorithm to measure sentence similarity that simultaneously takes into account the structural similarity as well as the similarity of concepts involved. This is done using UNL and WordNet [6]. The results obtained from our reference Case-Based Reasoning System and the results obtained through our method are compared and described in section 6. We conclude the paper in section 7 with a note that using semantic information helps in improving the case retrieval in Case-Based Reasoning Systems.

2 Role of Semantic Knowledge in Case Searches

In order to understand and appreciate the role and importance of semantic knowledge and sentence structure in case-retrieval process, we need to understand the working of systems that do not use this information and rely only on the word knowledge. *CHEF* [5] is a CBR system developed at Yale University by Hammond. The input to the system is a set of goals to be satisfied by a single integrated case. The cases in this system describe recipes for various dishes. *CHEF* stores its cases in memory indexed by various features like dish-type, ingredients etc. The input to the system is the type of dish that is to be prepared and a list of desired ingredients which are used as keywords to search for relevant cases in its index. Since the system uses only these features as inputs, the system is limited to the jargon of culinary only. The system offers no flexibility in that the user is expected to follow a set representation for input. Any input that falls outside the representation will not be able to produce desired results. Also, a user is not allowed the freedom to annotate his input with any remarks or comments that may be useful while preparing a plan for the dish.

CONSULT [11], developed by *Tata Consultancy Services*, is a more generic Case-Based Reasoning System. Each case in *CONSULT* pertains to a single problem and contains questions that are posed to the user for an interactive diagnosis of the problem [14]. Based on the inputs given by the user, a relevant case is retrieved and output to the user. Here, every case contains a *Title* field that describes the problem whose solution is contained therein. The user enters a problem in natural language text and a case is retrieved from the memory to perform further diagnosis. The problem of searching for a case that contains a problem similar to the one input by the user hence gets reduced to finding cases whose *Title* is the most similar to the problem input by the user. A set of questions are posed to the user, the answers to which are compared to the ones listed out under the relevant cases that are retrieved. A question-answer pair typically behaves as an attribute value pair. The answers provided by the user to the questions posed are compared to these values and a match is found. Our efforts have been directed toward devising an approach that will make the initial case retrieval based on the similarity between a problem statement and the case *Titles* more fruitful.

Let us look into an example that illustrates this. We consider a case-based system that is modeled on the *CONSULT* system. Let us consider three cases, whose respective *Title* fields contain the following

– My computer in office is not running
– Cannot run MS Office on my computer
– My machine is not working

Though the first two sentences are talking about two different problems, it is difficult to know this difference until we consider the meanings of the words present. The two sentences share most of their words and hence would seem very similar to each other to a system that follows conventional methods like stemming, gramming [9], etc. It might present both the cases as being relevant to a single problem. On the other hand, the first and third cases, though seemingly different at the word level, are highly related to each other. A system ignoring the meaning of words will not be able to capture this similarity.

We also need to appreciate the importance of sentence structure in sentence similarity measure. In our method, sentence structure similarity measure denotes whether similar concepts are playing similar roles in the sentences being compared.

A sentence is represented using an interlingua called Universal Networking Language (UNL) [4]. Information in every sentence is captured at three levels: the concepts that are involved, the role they play in the sentence and attributes that describe their properties. The role of concepts in the sentence with respect to each other is represented using UNL relations and it is these relations that we consider to capture sentence structure. In the next section, we present a brief introduction to UNL and how it extracts and represents information out of a natural language text.

3 Universal Networking Language

Information contained in natural language text sentences needs to be captured effectively and exhaustively to be useful for understanding and processing. Universal Networking Language (UNL), proposed by United Nations University [4], represents natural language in the form of a semantic network where the concepts form the nodes of the graph and the relations among these concepts form the links among them (see Figure 1). UNL represents information sentence by sentence. This hypergraph is also represented as a set of directed binary relations, each between two concepts present in the sentence. Concepts are represented as character strings called Universal Words (UW).

The knowledge within a document is represented in three dimensions:

– **Universal Words (UW)**: describe concepts that are present in a document. Since concepts are universal and are expected to be independent of any one language, the Universal Words also are language independent. The Universal Words are accompanied by restrictions that describe the sense of the word, given by the UW, in a given context. For example, *drink* can describe either *putting liquids in the mouth*, *liquids that are put in the mouth*, *liquids with alcohol* or *absorb* etc. But a concept with a restriction like *drink(icl$>$liquor)* describes the sense of *drink* as

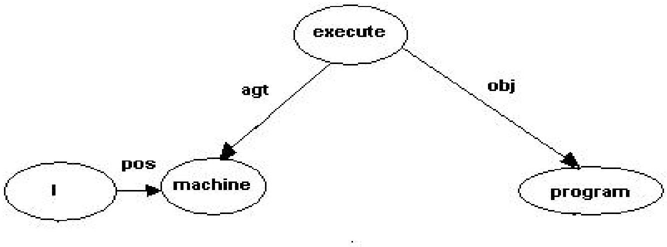


Fig. 1. UNL Graph for ***"MY MACHINE IS EXECUTING PROGRAMS"***

a noun standing for a type of *liquor*. The *icl* represents an *IS-A* kind of relationship between the concept and what follows *icl*. Therefore, each Universal Words represents a unique sense.

- **UNL Relations**: describe the relations between the concepts involved in the sentence and the roles (e.g. *subject* or *object* in case of nouns) that they play in conveying the meaning of a sentence. UNL uses a standard set of 41 relations to capture this knowledge. Each relation describes a kind of role that concepts can play towards the overall meaning of a sentence. Let us illustrate this with the help of an example- *My printer is not working*- the UNL for which is given below.

  agt(work(icl>function).@not.@present.@progress
  .@entry, printer(icl>machine))
  pos(printer(icl>machine), I(icl>person))

  The relations in UNL are binary defined as *rel(UW1, UW2)*. Here, not only the facts that *printer* is a kind of machine and *work* is a kind of action, but also, the relation between these two concepts, *printer* being an **agent** (*agt*) of the action of **not working**, are presented. Also, *I* is shown to be related to *printer* as its **possessor** (*pos*).

- **UW attributes**: capture and represent properties of concepts like tense of a verb, and speaker's perspective, attitude etc. In the above example, the use of *.@present* and *.@progress* with the UW for *work* describe the action of **working** as happening in the present time. Speaker's attitude like affirmation, contradiction, exclamation are also represented using these attributes.

  To illustrate this, consider the previous example. The presence of an attribute *.@not* indicates that the process of **working** is not happening. *.@entry* is a special attribute that indicates the main predicate of a sentence. This attribute is attached to the node from where generation of target natural language begins. In the following sections, we briefly describe the process of generation of UNL from a source natural language text, also known as *EnConversion*.

4 Generation of UNL from Source Natural Language

The process of conversion of a natural language text into its equivalent UNL representation is called *EnConversion* and the machine that performs this is called *EnCon-*

verter [12]. EnConverter (EnCo) is a language independent parser that performs morphological, syntactic and semantic analysis synchronously [3].

EnCo analyzes the source text sentence by sentence. It makes use of a knowledge rich lexicon of concepts and an exhaustive rule base for analysis. The EnConverter's function can be compared to that of a multi-headed Turing machine. The input sentence is converted into a node-list representation where a token (word or blank) forms a node. The EnCo works on this node-list through its windows (or heads), namely Analysis Windows (AW) and Condition Windows (CW). There are two Analysis Windows, but there can be any number of Condition Windows. EnCo checks the nodes under its two AWs, and the nodes that appear under its CWs. Based on the attributes of these nodes, it performs actions as described in the rule-base [1]. The various actions performed could be deletion, exchange, composition, forming a relation etc. Since there are only two Analysis Windows and at a time, operations are performed only on the nodes that appear under them, all UNL relations that are generated are binary relations. For example, consider a sentence, *Machine is executing*. The initial node-list that would be generated is shown.

/>>/Machine is executing/>>/

Here, >> and << indicate sentence head and sentence tail markers respectively. The EnCo picks up relevant entries and attributes for these words from a UW dictionary. In this case, the concept for **machine** will have attributes *N, INANI etc.*, whereas the attributes for **execute** will be *VRB, VOA-ACT etc.* A sample rule in rulebase is given below.

>(SHEAD){N::agt}{VRB:::}P10;

The above rule states that if there is a *noun* under the Left *AW* preceded by a sentence head indicator *SHEAD* (checked by a *CW*) and followed by a *verb* under the Right *AW*, then the *noun* is related to the *verb* as the **agent** of the action indicated by the relation *agt* in the rule.

In addition to forming relations between two nodes, EnConverter can add or delete attributes from a node. For example, in a sentence *Program will run*, the word *will* does not appear in the final UNL representation but EnCo adds an attribute *.@future* to the predicate of the sentence that is *run*. It is the UNL relations that we consider in our system when comparing two sentences for structural similarity.

5 Measuring Sentence Similarity

As mentioned previously, similarity between two sentences is measured on two counts: how similar are the concepts involved in the two sentences and how similar roles do the concepts play in the sentence? Since relations describe the roles that concepts play in the meaning of the sentence, similar structure sentences will have similar relations in their respective UNL representations. For example, *My machine is running* and *The printer is not working*, though different in meaning, have a very similar structure by virtue of the role *machine* and *printer* play (*i.e. agt*) in the meaning of their respective sentences. We now present the algorithm used in the CONSULT system followed by our algorithm to measure the similarity among sentences.

5.1. Sentence Similarity in CONSULT

Consult uses k-Nearest Neighbor (kNN) algorithm to determine the similarity between cases. The kNN computes a weighted Euclidean distance between two cases. Each case is represented as a set of attribute-value pairs. One such attribute of a case is the case *Title* and its value is a string that describes the problem to which the case is related. A case-similarity search involves matching of the case *Titles* too. We now describe briefly the string matching that is undertaken in CONSULT [11].

Let the input string be
Sytem hanged while doing a btch program.

Let the case Title of a case be
Software crashes when I run batch process.

It may be noted that the two strings share not a single word and that the words *System* and *batch* are misspelled in the input string. Sentence similarity in CONSULT proceeds in the following steps:

– **Stemming**: In this step, the input string is taken and all the words, that are derived or inflected, are reduced to their root forms. Hence, the input string now looks like

  *Sytem **hang** while **do** a btch program.*

The words that were changed are indicated in boldface.

– **Synonym Rewriting and Auto Correction**: In this step, common spelling errors are rectified and variants of a word that mean the same (i.e. synonyms) are reduced to a standard form. In our example, the word *Sytem* is auto-corrected to *System* and *btch* to *batch*. Also, the words *System* and *do* are reduced to their standard forms of *Software* and *run* respectively. At this stage, our input string becomes
  ***Software crash** while **run** a batch program*

– **Stripping of Noise Words**: Noise words (also called stop-words), that do not add anything significant to the overall meaning of the sentence, are excluded from the sentence before matching. In current example, the words "while" and "I" in the input string are dropped. Thus, our input sentence now looks like
  Software crash run batch program

– **Gramming**: After the first three steps, it may be noted that the two strings seem very similar to each other. Each string is now broken down into an unordered set of strings of fixed length or grams. This sequence of grams is then used for comparison. Similarity is computed as a function of the cardinality of the intersection at the gram level.

Thus, as is evident, CONSULT does not even attempt to utilize word meanings or the role that they play in the given problem sentence. However, this approach is prone to failure in the example three sentences mentioned in Section 2.

5.2. Comparing Concept Similarity

Taking UNL representation for one *Case Title* at a time, we compare each of the concepts occurring in it's UNL representation with the concepts that appear in the UNL representation for the problem sentence. The similarity score

is computed using the method proposed by Resnik [10] where similarity of two concepts is determined by the information that they share indicated by the most specific concept that subsumes them both in a concept hierarchy. Resnik used WordNet [6] (see the appendix) for this hierarchy of concepts. For every concept, its likelihood of occurring in the document is calculated by counting the number of instances of itself and the concepts subsumed by it in the document. Therefore, the more general a concept, the more number of occurrences it will have. Probability (or likelihood) of occurrence of a concept is given as

$$P(c) = Nc / N \tag{1}$$

where *Nc* is the number of times a concept *C* occurs in the document and *N* is the total number of words in the document. Using the Information Content Theory [13], the *Information Values* associated with each concept *C* is negative log of the likelihood of occurrence of the concept.

$$IC(c) = -\ln(P(c)) \tag{2}$$

We too used WordNet to arrange concepts in a hierarchy and assign them *Information Content Values* in the manner proposed by Resnik. However, in Resnik's method, the sense of the concepts being matched is not known. Therefore, a similarity score is measured for all senses of the two concepts and the maximum among them is chosen. While this may work in most of the cases, it is not always very effective. Use of UNL Universal Words helps us restrict our attention to only one sense of a concept and therefore produces the most useful similarity score.

If there are N_1 and N_2 nodes (or words) in the two sentences S_1 and S_2 respectively, then the concept similarity measure is calculated as

$$\frac{\sum_{n_1 \in S_1} \sum_{n_2 \in S_2} SimScore(n_1, n_2)}{(N_1 * N_2)} \tag{3}$$

The sum of all the similarity scores over all pairs of concepts, that are matched for two sentences, is taken and averaged over the number of comparisons made. This is done to ensure the number of occurrence of a concept does not affect, influence or mislead the final similarity score.

5.3. Comparing Sentence Structure Similarity

Given all the cases in our Case-Based Reasoning system, we begin by obtaining the UNL representations for the *Titles* in each case. We consider the graph representations of UNL for our system. However, we represent a labeled edge in a UNL graph as sequence of two links; one from the initial *concept node* to a *relation node* that is labeled after the relation and another link from the latter to the destination *concept node* of the original link as shown in Figure 2. The structural similarity of two sentences is obtained by calculating the total number of subgraphs that their respective UNL graphs share. The method of calculation of common subgraphs is based on the method proposed by [2]. The smallest unit of subgraph in this context is either an edge with exactly one *relation node* or a *concept node*. The common subgraph calculation for a pair of graphs being compared is performed by calculating, for all pairs of nodes taken from the two graphs, the sum of number of common subgraphs rooted at the given nodes.

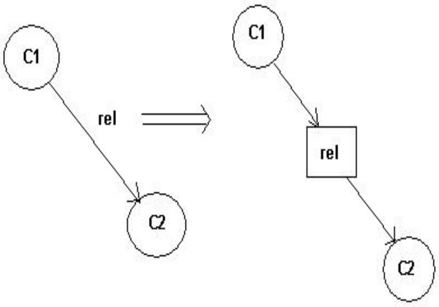


Fig. 2. Modification of Links in UNL Graphs

The recursive formula for common-subgraph calculation is given in [2].

$$C(n_1,n_2)=\prod\nolimits_{(x,y)\in sim(n_1,n_2)}(C(x,y)+2)-1 \tag{4}$$

where *sim(n₁,n₂)* is the set of common descendants of n_1 and n_2.

$$sim(n_1,n_2)=\left\{(x,y)\mid \begin{array}{r} x\in children(n_1) \\ y\in children(n_2) \\ label(x)\approx label(y)\end{array}\right\} \tag{5}$$

The condition *label(x)* ≈ *label(y)* in the above definition denotes similarity in the words' underlying concepts as computed by our concept matching algorithm.

Using the above definitions, the total number of common subgraphs between two graphs G_1 and G_2 is

$$C(G_1,G_2)=\sum\nolimits_{n_1\in G_1,n_2\in G_2,label(n_1)\approx label(n_2)}C(n_1,n_2) \tag{6}$$

The structural similarity between two sentences (or graphs) is now computed as

$$C(G_1,G_2)/N(G_1)*N(G_2) \tag{7}$$

where $N(G_1)$ and $N(G_2)$ are the total number of subgraphs in G_1 and G_2 respectively.

In the end, the concept and structural similarity scores obtained using the two methods are combined together to give us a cumulative similarity score.

6 Results

6.1. Experimental Setup

Our experiments based on the ideas and algorithms presented in section 5 were carried out on a case base of 120 cases. The cases dealt with problems faced by users in varied domains like *printer-related problems*, *DOS/Windows-related problems*, *Internet-related problems* and *DB2-related problems*. A set of 10 queries from each domain was taken and input to our system. The results obtained thus were compared with the ones that were obtained from CONSULT. We illustrate the performance our system with the help of a few examples.

Since our system uses both concept similarity and structural similarity, there are certain advantages and disadvantages with this approach. For example, let us consider the first query in the *DOS* category that was input to the system, *Windows creates a lot of temporary files*. This query did not have any corresponding relevant case in the Case-Base. The CONSULT and our system, however, returned the same case as the seemingly most relevant one, *I am not able to create a file in MS-DOS*. Obviously, the similarity here was established only by means of term matching. In addition to cases that matched due to presence of *files*, our system could also return results that talked about *programs*, *mails* etc. since they too are documents and are similar to *files*. This way, we could also include those cases that could have been relevant to this query but could have been ignored. As an illustration, an input query *My printer is not printing in Windows 98* also returns *Printer not writing to LPT1*. Note that our system could find a similarity between the concepts of *print* and *write* in the given context.

UNL Attributes also play an important role in concept matching. A system that relies only on terms and their grams will not be able to distinguish between sentences that are differentiated by the presence of a *not*. As another illustration, an input query *Unable to rename a file in DOS* returns *I am unable to create a file in DOS* due to its high structural similarity with respect to the relations that are shared by *unable*, *file* and *DOS* with the main verb of the each sentence, as well as concept similarities.

Sometimes, precision of case-retrieval may suffer due to concept similarities. This is illustrated in the following example.

– The input sentence was: My HP printer is not working

Table 1. Case Matching Results – Sentence 1

Sentence	Score (Our Method)	Score (CONSULT)
My mouse is not working in MS-DOS	0.53	–
My printer does not print the whole page	0.43	50
The printer is not writing to LPT1	0.40	35
I cannot read the print on my page when it is printed	–	30

As mentioned previously, this is an instance where concept similarity generates undesired results. A *mouse* is considered similar to *printer* since they both are devices and the two sentences' structural similarity is very high resulting in a close match.

– The input sentence was: I am unable to download pictures from the Internet

Table 2. Case Matching Results – Sentence 2

Sentence	Score (Our Method)	Score (CONSULT)
I am not able to download exe files from the Internet	0.38	60
I am not able to hear music from the Internet	0.31	30
I am unable to view the Internet connection icon on my desktop	0.35	30
Internet Explorer quits opening web pages while surfing	0.33	30

Note the high structural similarity of the first three cases with the query sentence. Both structural similarity and concept similarity give a high score to the first case in this example.

– The input sentence was: I am unable to surf the web

Table 3. Case Matching Results – Sentence 3

Sentence	Score (Our Method)	Score (CONSULT)
I am unable to browse the Internet	0.86	–
I am unable to print entire screen image	0.51	–
I am unable to install my Lexmark printer	0.42	9.09

The above example illustrates the power of our system. Given that *surf* is a common word used to describe the activity of *browsing* the Internet and *web* being synonymous with *Internet* or a computer network, we are able to identify the similarities between these concepts here and produce a match. CONSULT, however, was not able to come up with a match in this case since it could not find any common terms. The other two cases that are retrieved are similar to the original query by way of structural similarity with very little concept similarity.

– The input sentence was: I cannot create tables in DB2

Table 4. Case Matching Results – Sentence 4

Sentence	Score (Our Method)	Score (CONSULT)
We cannot create DB2 database for our project	0.62	50
I am not able to create index on table	0.56	50
I am not able to create table in database	0.53	50
I cannot use database	–	50

The first case here matches the query due to its strong similarity in its structure with respect to *create* and also the presence of *DB2*. The second and third too are quite similar to the query. CONSULT also provides *I cannot use database* as a retrieved case which our system does not pick.

We ran our system on a case-base of 120 cases. The precision for our system as well as that of CONSULT was calculated. We define *precision* as the number of relevant cases among those that are retrieved by the system. Overall, our system provided a higher precision than CONSULT in all domains.

7 Conclusions

In this paper, we presented a **novel method that uses semantic information to improve relevant case retrieval in Case-Based Reasoning systems**. As is observed, conventional methods of sentence similarity measures that do not take word meanings into account, fail miserably in scenarios where two different words in two sentences may be talking about the same concept. Such systems are also prone to failure in scenarios where presence of same words in two sentences convey different meanings altogether. This was highlighted by some example sentences given in section 2. These problems are duly and effectively handled by our system because it not only considers words in a sentence, but also, their correct senses. The capabilities of this system are taken another step forward by taking into account the similarity in sentence structure. In short, use of additional semantic information, obtained through UNL in our case, helps us to better evaluate similarity of sentences.

Appendix: WordNet

WordNet is a lexical resource that organizes words and concepts based on their similarity in meaning [6]. It divides the lexicon into five categories; noun, verbs, adjectives, adverbs and functional words, each of which follows a different semantic organization. It organizes concepts in terms of word meanings. A word meaning is represented by a set of all the word forms that can be used to express it called a *Synonym Set* or *Synset*. These synsets designate meaning to a word. The organization of WordNet describes a number of semantic relations between concepts represented as pointers between these *Synsets*. Some of the semantic relations found in WordNet are:

- **Synonymy**: defines a relation between concepts that mean the same. By synonymy, we mean that usage of one Synset can be replaced by the other without changing the meaning of the concept.
- **Antonymy**: is relation that is formed between word forms and not word meanings. This is because, an opposite of ***x*** is not always *not-**x***.
- **Hypernymy**: is a semantic relation that, along with *hyponymy*, defines a *IS-A* hierarchy between two concepts. This relation is transitive and asymmetrical and generates a hierarchical semantic structure. This is what is used by Resnik, and subsequently by us, to generate *Information Content Values* for concepts that occur in our Case-Base.

References

1. Shah C., Parikh J., Soni T.,"Conversion of English Langage Texts to Universal Networking Language", *B.E. Dissertation,Dharamsinh Desai Institute of Technology, Nadiad,* 2000.
2. Collins M., Duffy N., "Parsing with a Single Neuron: Convolution Kernels for Natural Language Problems", *Technical Report, University of California and Santa Cruz,* 2001.
3. UNL Centre/UNDL Foundation, "EnConverter Specification Version 3.3", April 2002.

4. Uchida H., Zhu M., Della S.T., "UNL: A gift for a millenium", *The United Nations University*, 2000.
5. Hammon K., "Explaining and Repairing plans", *Journal of Artificial Intelligence Research,* 1990.
6. Miller G. A.,Beckwith R.,Fellbaum C.,Gross D.,Miller K. J., "Introduction to WordNet: An Online Lexical Database", *Technical Report, Princeton University,* 1993.
7. Mitchell T., "Machine Learning", *McGraw Hill Companies Inc,* 1983.
8. Bhattacharyya P.,Choudhury S.R.,Gupta S.S., "Man power Planning with Case Based Reasoning: The Selector", *International Conference on Knowledge-based Computer Systems (KBCS),* December 1998.
9. Porter M., "An algorithm for suffix stripping", *Readings in Information Retrieval, San Fransisco, US,* 1997, 313-316.
10. Resnik P., " Semantic similarity in a taxonomy: An infomation-based measure and its application to problems of ambiguity in natural language", *Journal of Artificial Intelligence Research,* 1999.
11. Chakraborti S.,Balaraman V.,Vattam S.S., "Using CBR inexact search for intelligent data retrieval", *International Conference on Knowledge-based Computer Systems (KBCS),* 2000.
12. Pairkh J., Dave S., Bhattacharyya P., " Interlingua based english hindi machine translation and language divergence", *Journal of Machine Translation,* September 2002.
13. Shannon C., "A Mathematical Theory of Communication", *The Bell System Technical Journal,* July, October 1948, 379-423, 623-656.
14. Tata Research, Development and Design Centre, "CBDM – A Case-Based Development Methodology", *Technical Report,* 2000.

Facilitating Communication Between Languages and Cultures: a Computerized Interface and Knowledge Base

Claire-Lise Mottaz Jiang,[1] Gabriela Tissiani,[2]
Gilles Falquet,[1] Rodolfo Pinto da Luz [3]

[1] CUI, University of Geneva, Switzerland,
(Claire-Lise.Mottaz, Gilles.Falquet)@cui.unige.ch
[2] CNPq Researcher at CUI, University of Geneva, Switzerland
Gabriela.Tissiani@cui.unige.ch
[3] Instituto UNDL Brasil, Florianópolis, Brazil
Luz@undl.org.br

The Universal Networking Language (UNL) deals with communication, information, knowledge, language, epistemology, computer sciences, and related disciplines. This interdisciplinary endeavor calls for theoretical and applied research, which can result in a number of practical applications in most domains of human activities. Specially, it can help solving some of the most critical problems emerging from current globalization trends of markets and geopolitical interdependence among nations. This paper presents a project that aims to contribute with UNL KB (UNL Knowledge Base) theoretical and practical. The goal is to make possible people from various linguistic and cultural backgrounds to participate at UNL KB construction in a distributed environment.

1 Introduction

This paper presents a project that will be developed by the following partners: Information System Interfaces (ISI) Research Group at University of Geneva, the UNDL Foundation, and the United Nations Institute for Training and Research (UNITAR). This project is part of the Geneva International Academic Network Programme (GIAN). It involves creation of ontologies, for the Universal Networking Language (UNL) Knowledge Base (KB).

The project argues that the construction of these KB ontologies will contribute to the United Nations initiative of creating the multilingual infrastructure on UNL. Its infrastructure is meant to facilitate communication among natural languages on the Internet and includes development of a broad knowledge base from diverse linguistic sources and cultural backgrounds [10].

The UNL multilingual infrastructure is an interdisciplinary undertaking that involves both linguistic and engineering aspects. Its main components are (1) a formal, language-independent, non-ambiguous artificial language (UNL) and (2) a system that manages the interfaces between natural languages and the UNL over computer networks. The UNL itself comprises a vocabulary - a list of concepts, called "univer-

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 359–369.

sal words" represented in a language-independent way as character strings—and a knowledge base that explicit the relations between universal words (UNL KB).

The UNL KB construction raises several challenging problems, because of its particularities (size, high number of contributors, distributed environment, linguistic and cultural issues). Examples of problems to be tackled include:

- How to coordinate multiple, distributed (remote) contributors?
- How to deal with multilingual and multi-cultural issues in order to create a "global" knowledge base?
- Which infrastructure is needed to enable a distributed, asynchronous work and still end up with a coherent knowledge base?
- How to maintain the knowledge base to ensure its validity over time?

The overall goal of this project is to create a framework and tools to support the development and the evolution of the UNL knowledge base. This project includes both applied and theoretical research. As there is no known straightforward engineering solution to this set of problems, theoretical studies will be carried out to support a practical realization.

2 Related Work

The proposed project will use several research fields such as studies on knowledge bases, knowledge representations, and ontologies.

2.1 Knowledge Bases and Knowledge Representation Models

For the present project, we are particularly interested in works that propose models and languages for the representation of concept definitions. In the artificial intelligence field, bases containing knowledge about concepts are usually called "ontologies" or "terminological knowledge bases". Many languages and models have been devised to describe terminological knowledge. Formal terminological knowledge representation systems include for example KL-ONE, CLASSIC, LOOM, OIL, OCML, and OWL. These systems are based on first order logic, description logic, or on frame systems. Research in this area is growing fast to respond to the "semantic web" initiative of the W3C consortium. It focuses on engineering of formal ontologies (how to create and maintain them) and on ontologies use to integrate heterogeneous resources.

2.2 Ontology Engineering Tools

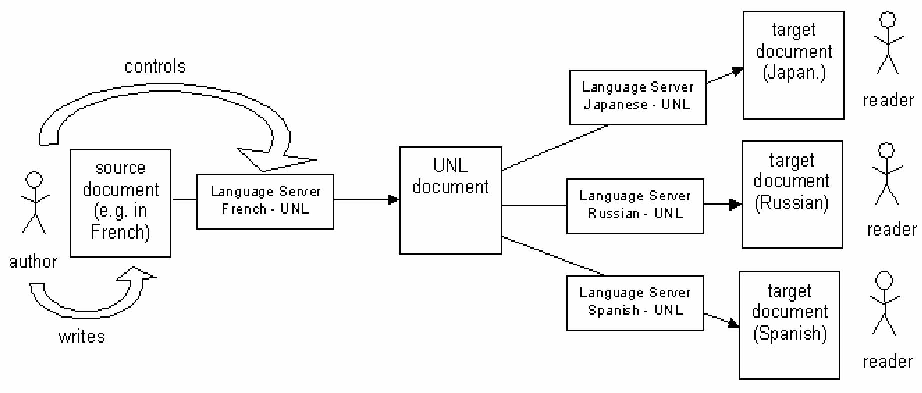
Since the mid 1990s, many ontology engineering systems have been developed (Protégé, OntoEdit, WebODE, etc.). These systems can be classified in two categories, depending on type of knowledge representation language they rely on:

- description logics based (for example Ontosaurus)
- frame based (Ontolingua, Protégé, OntoEdit, etc.)

Different presentation styles are used to display the content of the ontology. The first form is indented text, which is used in many HTML interfaces. In order to hide the formalism and provide user-friendly tools, graphical interfaces have been developed, such as node-link diagrams (graphs), tabular views, or hyperbolic trees. Graphs have been extensively used to visualize knowledge structures in artificial intelligence (semantic networks, conceptual graphs and database design (from entity-relationship diagram to object and class diagrams).

3 UNL System and the UNL KB

Today, a vast proportion of information available on the Internet is written in only a few languages, among which English ranks first. Computer translation systems have been used to provide users with a means to read information written in languages they do not understand. However, two important problems remain unsolved: 1) these systems usually function only with "dominant" languages, such as English, French, Spanish, Russian or Chinese; 2) moreover, they perform well only in specific conditions.


controls
author
source document (e.g. in French)
writes
Language Server French - UNL
UNL document
Language Server Japanese - UNL
Language Server Russian - UNL
Language Server Spanish - UNL
target document (Japan.)
target document (Russian)
target document (Spanish)
reader
reader
reader

Fg. 1. UNL System.

The purpose of introducing the Universal Networking Language (UNL) in communication networks is to achieve accurate exchange of information among different languages. The UNL is an artificial language for computers, unlike Esperanto, which is also artificial, but for humans. In UNL, concepts and sentences are represented formally and non-ambiguously, through logical expressions. The UNL is language-independent; it provides the possibility to work at the semantic level, and enables the construction of a comprehensive "library" from various types of knowledge expressed in diverse indigenous sources.

The UNL multilingual infrastructure emerges from the convergence of linguistic and epistemic research with electronic and digital media, computers, and communication networks. The result comprises the following elements:

– the UNL specifications
– the Universal Word (UW) Dictionary

  Universal Words represent in logical expressions the meaning of the words in natural languages, normally represented in alphabet characters or ideograms. An UW denotes a concept.

– the UNL Knowledge Base (binary semantic relations between Universal Words).

The goal of this project is to theoretically study the distributed construction of very large knowledge bases and to provide a framework and tools to build the UNL KB.

3.1 How Does The UNL System Work?

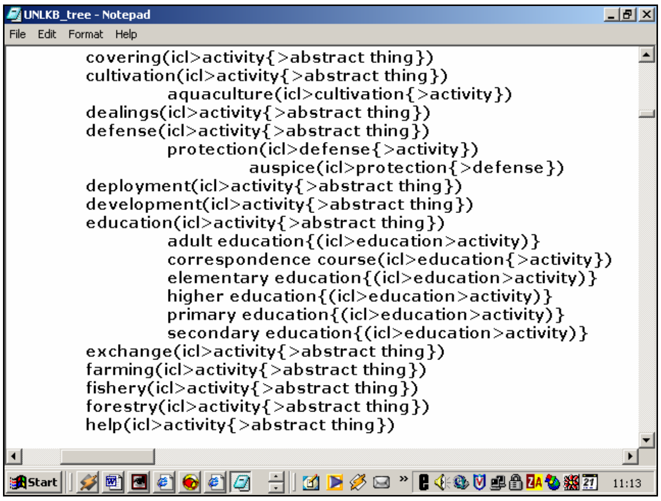
In order to produce an equivalent UNL for a natural language document, one can use the UNL editor of his/her corresponding Language Server. This process is called “enconversion” and it can be either completely automatic, or interactive or completely manual. Finally, the UNL viewer is used by the reader of the document to "deconvert" the UNL text into his or her natural language, by using the UNL viewer of hi/her appropriate Language Server [10, 11].

3.2 The Structure of the UNL KB

The UNL KB structure and mechanism is based on a hierarchy formed of binary relations between UWs. Every UW must be defined in the KB and linked to the other related existing UWs [9].

The UNL Knowledge Base stores UWs that are inter-linked to each other by one of the relations present at UNL Specification. The UNL KB defines an UW by its relations with other existing UWs. Therefore, this requires also the designation of which level it should be situated as well as under which subordinate UW it should be set (on icl case). To create a new UW it is necessary to label the desired concept and also define its relation list (UW) [9] [10]. This list comprises of a relations set with other UWs required to define a concept such as a label. Subsequently, to build up the new UW in the UNL KB, it is indispensable to position it in the existing hierarchy. Finally, all new UW input in KB gate (web application that allows the access to the KB) must be homologated by the UNDL Foundation before its final inclusion into the KB.

As this whole process is text based, it requires a great human effort to manage it. Moreover, the specific view of the UNL expertise is required to update the KB. Even thus, it is not easy to manage these processes. It requires a great human effort to coordinate the multiple remote contributions to the KB, as well as an easy interface to do so.



```
covering(icl>activity{>abstract thing})
cultivation(icl>activity{>abstract thing})
        aquaculture(icl>cultivation{>activity})
dealings(icl>activity{>abstract thing})
defense(icl>activity{>abstract thing})
        protection(icl>defense{>activity})
                auspice(icl>protection{>defense})
deployment(icl>activity{>abstract thing})
development(icl>activity{>abstract thing})
education(icl>activity{>abstract thing})
        adult education{(icl>education>activity)}
        correspondence course(icl>education{>activity})
        elementary education{(icl>education>activity)}
        higher education{(icl>education>activity)}
        primary education{(icl>education>activity)}
        secondary education{(icl>education>activity)}
exchange(icl>activity{>abstract thing})
farming(icl>activity{>abstract thing})
fishery(icl>activity{>abstract thing})
forestry(icl>activity{>abstract thing})
help(icl>activity{>abstract thing})
```

Fig. 2. The present UNL KB

4 The Complexity of the UNL KB

The main reasons for considering dealing with UNL KB a complex task are: 1) the inclusion of new UWs and 2) the management of the remote contributions into the database.

The first one is related to the creation complexity of new UWs, which includes the difficulty to relate them to existing ones. Thus, this process comprises the new UW position at the KB as well as the description of its binary relations. To avoid ambiguities, new concepts must be described according to different contexts to assure the various meanings among the languages.

The second related problem is associated to management of new inputs into the KB. It is difficult to supervise all contributions that come into the "KB gate", to be farther transferred to the KB. Even if the process of homologating new UWs is necessary, the control of shared work can become less complicated with an efficient interface.

Thus we believe that a graphic user interface can facilitate these procedures through the visually representation of the KB textual schema. Indeed, it depends on the use of proficient presentation techniques to allocate the UWs.

It is also important to consider that this interface must allow the representation of the same UW described by different points of view. For instance, it can help the organization of the different UWs' definitions according to the diverse domain's dictionaries.

In this case, the search for existing UWs can be enhanced using a focused investigation for concept's definitions instead of a traditional search for "headwords". Therefore, it will make easier comparisons between UWs and it will add the classification of similarly concepts represented by them. By investigating the similarities between concepts, one can avoid the repetition of similar UWs. Under these settings, the UNL KB can include also a database documentation to support the managing of new UWs.

Finally, it is believed that those problems could be overcome by adoption of new KB formalisms. The proposed research builds on research works in several fields such knowledge representation models and languages, knowledge engineering, hypertexts and human-computer interaction.

5 The Proposal

The project addresses the problem of developing and maintaining a voluminous knowledge base (the UNL KB) in a distributed environment. The problem will be attacked along different axis, like the design and prototype of infrastructure, user interfaces for the UNL KB [10].

The first part of the project will focus on designing and prototyping a storage infrastructure for the UNL KB. The first step will design a model to store UNL definitions in a relational database. Then, this model will be extended to include "management information", such as versions, validation, point of views, access rights, roles, that are needed to support the knowledge base construction process.

This part of the project comprises not only enlargement of the UNL KB but also the design and prototype of various user interfaces for it. These will support different tasks and users, for instance authors, reviewers, or checkers. The first step will be to provide navigational and graphical interfaces to explore the knowledge base and to be familiarized with it.

The second step will be to create interfaces to add, edit, and delete objects in the knowledge base. These include auxiliary interfaces to support its maintenance. In this phase, we will use a wide range of paradigms, such as forms, hypertext, high-density graphical objects (hyperbolic trees, fisheye views, etc), or 3D objects [6] [7] [10]. We will also study which kind of inferences are possible using the UNL KB, not only to present richer information for users, but also to help contributors to expand further the UNL KB and to check consistency of their contributions. It will help to improve the knowledge representation structure and to provide a collaborative environment; hence, the data structure could be set up to represent comments, issues, arguments, decision, etc. For instance, in a multi viewpoint approach, every definition can be related to a viewpoint. Thus through a hypertext interface, the KB can be more comprehensible to the user since s/he will be able to view corresponding annotations when browsing or navigating the ontology. The possibility of navigation when building the knowledge base can help the whole process of UW development. For instance, it can be applied to create an interface that allows the user to investigate the UWs syntax by a visual approach.

In the last phase, we will establish a methodology for collaborative building of the UNL KB. This methodology will include the manual addition of new concepts from diverse natural languages as well as the importation of concepts from existing ontologies. In particular, it will describe the concept review and validation process, including resolution of definition conflicts.

Each part of the project will include a theoretical study, as the issues raised by a distributed construction of a large knowledge base are not yet fully understood and applicable solutions are still missing.

This work aims at progressing towards a complete environment that truly supports the process of ontology development (in contrast to ontology editors whose only goal is to enable the "encoding" of an already well-specified ontology).

6 Interfaces Specification

In another project, we applied the Lazy interface specification language to create visualization and manipulation interfaces for ontologies expressed in description logics [4]. As the UNL formalism shares some similarities with description logics, we intend to use the same kind of techniques to develop tools for the UNL KB. In this section we present a few examples of such ontology interface.

Since there are many tools to develop database applications and interfaces, it seems natural simply to store ontology in a database and to build ontology-engineering tools with database development tools. However, these tools are intended to develop "typical" database applications and are not specifically targeted at knowledge or ontology management applications. In particular, they usually provide form-based or table-based views of the data.

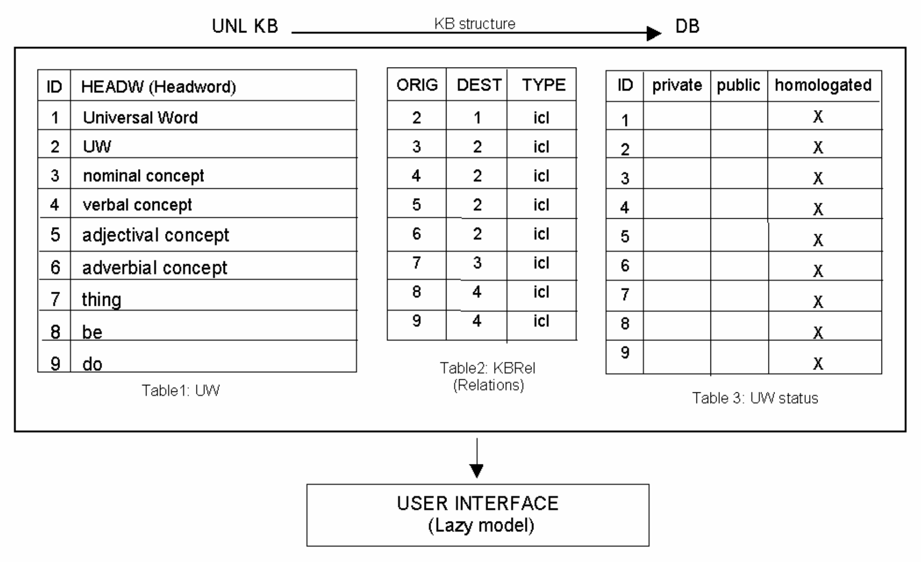
A hypertext view is a derived (computed) hypertext that represents the contents of some underlying information source. The idea is to provide the user with an easy to use hypertext interface that enables him/her to navigate within the information source. Thus it replaces database querying or other complex access mechanisms with just hypertext link following.

The Lazy language was designed to specify and implement hypertext views on relational and object databases. The language has been applied to generate different Web applications. Since the language is purely declarative, hypertext views can be specified without any programming.

The hypertext specification language is based on the concept of node schema. A node is comprised of - a set of parameters - a content specification (made of element specifications) - link specifications - a selection condition (what database objects to select). The instantiation of a node consists in interpreting a node schema for a given set of arguments and on at current state of database. An instance of a node schema is obtained by first selecting the objects (e.g. relation tuples) of the data collection(s) that satisfy the selection expression. Then the content and link specifications are evaluated on each selected object to generate node contents and links to other nodes.

In the Lazy model there are three types of links: reference links (the well known web links), inclusion links (to include the content of another node at this location), and expand in place links (clicking on such a link will open the target node at the link location, i.e. within the source node). These last two types of links are essential to build usable interfaces. We will see in the next sections, examples of the first prototype of this project, illustrating the usefulness of these types of links. For instance, by adding an expand-in-place link to the same node schema, we immediately obtain a typical "class browser" view.

```
node uwWithIcl[u]
  { <b>(headw), " " ,
    expand href iclOf[u]("&lt;icl")
  }
from UW selected by UW.id = u
```



ID	HEADW (Headword)
1	Universal Word
2	UW
3	nominal concept
4	verbal concept
5	adjectival concept
6	adverbial concept
7	thing
8	be
9	do

Table1: UW

ORIG	DEST	TYPE
2	1	icl
3	2	icl
4	2	icl
5	2	icl
6	2	icl
7	3	icl
8	4	icl
9	4	icl

Table2: KBRel (Relations)

ID	private	public	homologated
1			X
2			X
3			X
4			X
5			X
6			X
7			X
8			X
9			X

Table 3: UW status

Fig. 3. Database schema for the UNL KB

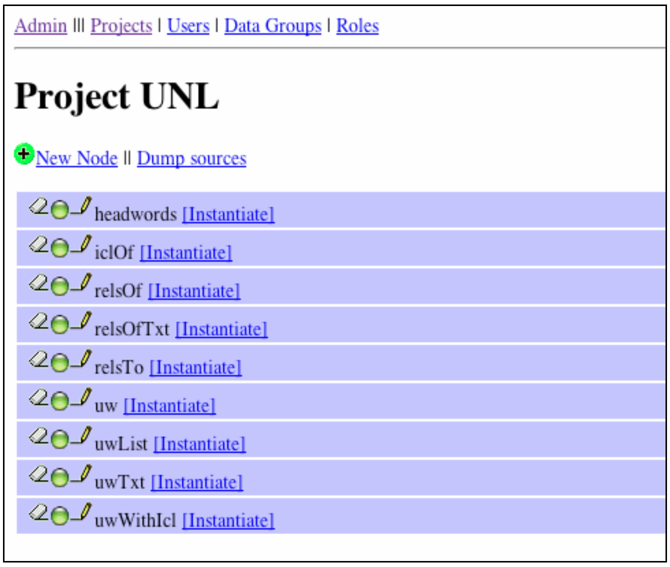
7 UW Validation Process

Since the UWs homologation has been doing manually, the new UWs cannot be validated everyday, as the original UNL project proposes. In order to make it easier, we will add some workflow information to the KB. Each UNL concept can be stored into a relational database described as the proposed DB structure (figure 3). By adopting three different statuses for each UW- private, public and homologated – we can allow contributors to insert new UWs, even if they are not yet validated.

The figure 4 shows an hypertext view from part of the first prototype for the proposed UNL KB. It is made of nodes instances connected through hyperlinks. The figure 5 shows the navigational evolution of the node “uwWithIcl”, where all UNL concepts appear linked by reference and expand links.

8 Dealing with Collaborative Issues

Ontology or knowledge base building is usually a collaborative task that includes both domain experts and ontology experts. In the case of UWs, there will be always many different point of views of the same concept according to the various domains



Admin ||| Projects | Users | Data Groups | Roles

Project UNL

New Node || Dump sources

headwords [Instantiate]

iclOf [Instantiate]

relsOf [Instantiate]

relsOfTxt [Instantiate]

relsTo [Instantiate]

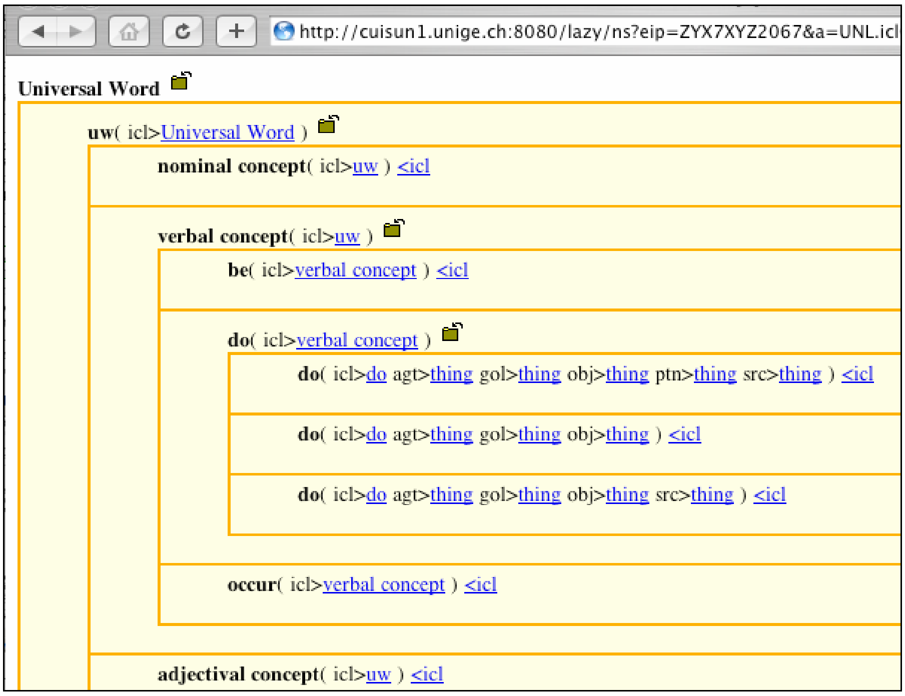
uw [Instantiate]

uwList [Instantiate]

uwTxt [Instantiate]

uwWithIcl [Instantiate]

Fig. 4. First nodes of the proposed UNL KB interface.



http://cuisun1.unige.ch:8080/lazy/ns?eip=ZYX7XYZ2067&a=UNL.icl

Universal Word

uw(icl>Universal Word)

nominal concept(icl>uw) <icl

verbal concept(icl>uw)

be(icl>verbal concept) <icl

do(icl>verbal concept)

do(icl>do agt>thing gol>thing obj>thing ptn>thing src>thing) <icl

do(icl>do agt>thing gol>thing obj>thing) <icl

do(icl>do agt>thing gol>thing obj>thing src>thing) <icl

occur(icl>verbal concept) <icl

adjectival concept(icl>uw) <icl

Fig. 5. Evaluation of the link "Universal Word"

and culture backgrounds. Thus, debates and conflicts on concept definitions are inevitable. Conflicts can occur either on meaning or on form of a definition, which includes the difficult of defining the headwords to write its definition [1] [4]. As happens with description logics ontologies, we believe that the development environment should not only include the UNL KB itself but also various kinds of "management" information. In other words, all sort of information that helps to build the KB (such as a viewpoint/conflict management mechanism, documents, notes, external ontologies, and databases). In fact, the use of a database as a storage infrastructure and use of Lazy to specify the interfaces enables to build an extensible environment easily, simply by adding relevant tables and interface nodes (figure 6).

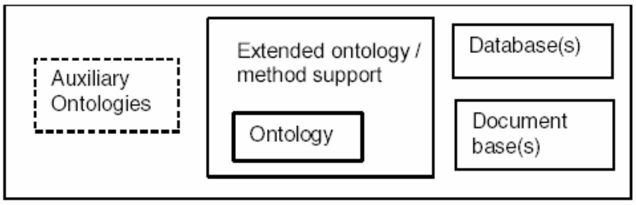


Fig. 6. Ontology environment

9 Final Considerations

The first outcome of this project will be a theoretical study of the different aspects of the distributed construction of large-scale knowledge bases. This study will be materialized in the form of several publications about specific issues and their resolution.

The second, and most important, contribution will be software applications, technical specifications, user manuals, etc. to help developers and contributors to work in a distributed environment. The applications will be packaged as "open software" and will be deployed by the UNDL Foundation to run the UNL System.

In parallel, educational and training activities will prepare a larger number of people to get involved in the collaborative development of the UNL knowledge base.

Finally, a website will be created, firstly to facilitate the wide distribution of the various software applications and documentation, but also to provide an online demonstration, for educational purposes. This website will act as a showcase and it is expected that it promotes the use of UNL and encourage new participants to take part in its development.

References

1. Falquet, G., Mottaz Jiang, C.-L., "A Framework to Construct Hypertext Interfaces for Ontology Engineering", submitted to the IJCAI03 workshop on "Knowledge Management and Organizational Memories", 2003.

2. Falquet, G., Mottaz Jiang, C.-L. "A Model for the Collaborative Design of Multi Point of View Terminological Knowledge Bases" in R. Dieng and N. Matta (Eds) Knowledge Management and Organizational Memories, Kluwer, 2002.
3. Falquet, G., Mottaz Jiang, C.-L. "Navigation hypertexte dans une ontologie multi-points de vue", in Proc. NimesTIC'01 conf., Nîmes, France, December 2001.
4. Falquet G., Mottaz C.-L., "Conflict Resolution in the Collaborative Design of Terminological Knowledge Bases". in Knowledge Engineering and Knowledge Management: Methods, Models, and Tools, R. Dieng and O. Corby (Eds.),Lecture Notes in Artificial Intelligence 1937, Springer-Verlag, 2000
5. NG, G. K. C. Interactive Visualisation Techniques for Ontology Development, PhD thesis, University of Manchester, UK, 2000.
6. OntoWeb Consortium, A Survey of Ontology Tools, 2002, www.ontoweb.org/download/deliverables/D13_v1-0.zip.
7. OntoWeb Consortium, A Survey on Methodologies for Developing, Maintaining, Evaluating and Reengineering Ontologies, 2002, www.ontoweb.org/download/deliverables/D1.4-v1.0.pdf
8. Schneiderman, B. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 3rd edition, Addison-Wesley, Reading, Massachusetts, 1998.
9. Papers presented at the International Conference on Universal Knowledge, Goa, 2002. http://www.cfilt.iitb.ac.in/icukl2002/ papers/index_of_papers.html
10. Uchida H., Zhu M. and Della Senta T., The UNL, a Gift for a Millennium, Institute of Advanced Studies, United Nations University, 1999
11. Uchida H., Zhu M., The Universal Networking Language beyond Machine Translation, presented in the International Symposium on Language in Cyberspace, Seoul, 2001

Using WordNet for linking UWs to the UNL UW System

Luis Iraola

Facultad de Informática, Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Madrid, Spain
luis@opera.dia.fi.upm.es

Abstract. This paper presents the work done with the Spanish-UNL dictionary compiled at the Spanish Language Centre in order to enrich the universal words it contained with the supplementary semantic information required to produce a master entries dictionary. Focusing on a subset of the Spanish-UNL dictionary, namely on the substantives it contains, the work has consisted in automatically enrich the universal word associated with each substantive with the semantic information required to link the universal word to the Universal Word System. For this process, WordNet has been employed as an external source of semantic information and used in addition to semantic features already present in the dictionary. The results achieved are not final and further work is required for a fully automatic, high quality semantic enrichment of the current entries. However, the work done shows the fruitfulness of the approach and its outcome has contributed to the creation of a master entries dictionary.

1 Introduction

A UNL dictionary in which language entries are associated with universal words (UW for short) can be viewed as a repository of UWs and as such does not organise its contents in any way. It links a set of UWs with lexical items of a specific language, each entry having no relation with any other. The necessity of establishing certain relations between UWs arises when considering several desirable features of the UNL system:

– Setting the combinatory possibilities of each UW with respect to any other UW regarding the conceptual relations that may link them and the attributes they may accept.
– Enabling a "fall-back" generation mechanism for those UWs that are not linked with head words in a given language at a given time. Those UWs would be replaced with semantically close but linked UWs so allowing generation to continue.

In order to support these features, a network with the set of UWs as nodes and semantic relations as arcs has been proposed. Such network is called the UNL UW System [1, 2]. Therefore, and in order to build the UW System, UNL Language Centres have to modify their respective UNL language dictionaries for including such new information. Once modified, the new master entries dictionaries will be the repository from which current language dictionaries will be produced as well as the UNL Knowledge Base will be created.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 370–379.

The current UW System consists of several hierarchies to which UWs are linked via inclusion relations (‘icl’) with broader meaning UWs. At the top level, the UW System distinguishes between entities (‘thing’), actions originated by an agent (‘do’), actions that happen without the intervention of an agent (‘occur’), states (‘be’), modifications of actions, events or states (‘how’) and modifications of entities. Each of these maximally general concepts (‘thing’, ‘do’, ‘occur’, ‘be’ and ‘how’) is the root of a hierarchy.

The hierarchy under ‘thing’ is by far the most elaborated one, containing distinctions between concrete and abstract things, functional and spatial entities and so on. Every UW denoting and entity must be located somewhere under the “thing” concept, and for doing this expert knowledge of the UW System and of the lexical meaning of the UW to be linked is required. Experts in the UW system and in English lexicography may manually establish the appropriate semantic links between each UW and the UW System. However, given the amount of entries that need to be processed (in the order of tens of thousands) any alternative that allows us to automate at least part of the task deserves to be explored.

2 Using Wordnet For Linking Uws

In order to automate the task of locating under “thing” the UWs associated with Spanish substantives, a procedure involving the use of WordNet [3] has been devised and put into practice. The procedure relies onto two central insights:

- WordNet 1.6 covers the practical totality of the English lexicon. English substantives in particular are organised in a hierarchy using the hyponym relation, which has been found very similar to the relation of semantic inclusion (‘icl’) employed by UNL. Besides, a first inspection found substantial similarities between the WordNet hierarchy and the most general concepts employed in the UW System regarding the organisation of the meanings of substantives.
- Given the polysemous nature of most English substantives, these words appear in more than one synset. However, these synsets frequently share a common hypernym in the hyponym hierarchy and this common ancestor can be related to a concept under ‘thing’ in the UW System.

2.1 Some Examples

The UW ‘arrogance’ (associated to the Spanish substantive ‘arrogancia’) has been located in the WordNet 1.6 hyponymy hierarchy in the following place:

```
1 sense of arrogance
Sense 1
arrogance, haughtiness, lordliness
  => pride
   => trait
    => attribute
     => abstraction
```

The synset identified with the meaning of ‘arrogance’ is linked in a chain of hypernyms with different nominal concepts until we reach the root node ‘abstraction’. The distinction between concrete and abstract things plays also a key role in the UW System, and besides that, the intermediate node ‘attribute’ is also an organising concept in the UNL hierarchy. Therefore, using the hyponymy relation defined in WordNet we can automatically link the UW ‘arrogance’ with ‘attribute’.

The UW ‘conquest’ (associated with the Spanish substantive ‘conquista’) has been registered in WordNet 1.6 as polysemous, each of its three senses located in a different place under the hyponymy hierarchy:

```
3 senses of conquest
Sense 1
conquest, conquering, subjection, subjugation
  => capture, gaining control, seizure
   => acquiring, getting
    => deed, feat, effort, exploit
     => accomplishment, achievement
      => action
       => act, human action, human activity
Sense 2
conquest
  => success
   => attainment
    => accomplishment, achievement
      => action
        => act, human action, human activity
Sense 3
seduction, conquest
  => success
   => attainment
    => accomplishment, achievement
     => action
      => act, human action, human activity
```

The three senses of ‘conquest’ share common hypernyms from the node ‘accomplishment, achievement’ upward. Immediately upon this common node we find ‘action’, which is also a concept employed in the UW System for organising entities. Therefore, we can locate ‘conquest’ under ‘action’ since whatever the exact sense of ‘conquest’ is the intended one for its association with the Spanish headword, it must be under ‘action’ necessarily.

Linking ‘conquest’ with ‘action’ is certainly a high level link, one that does not precise the specific kind of action we are dealing with. However, this high level distinction is all that we need at the present since the UW System does not make finer distinctions under ‘action’.

The word ‘book’, when considered a substantive, is highly polysemous according to WordNet 1.6:

```
8 senses of book
Sense 1
book
  => publication
   => work, piece of work
```

```
    => product, production
     => creation
      => artifact, artefact
       => object, physical object
        => entity, something
Sense 2
book, volume
  => product, production
   => creation
    => artifact, artefact
     => object, physical object
      => entity, something
Sense 3
record, recordbook, book
  => fact
   => information, info
    => message, content, subject matter, substance
     => communication
      => social relation
       => relation
        => abstraction
Sense 4
script, book, playscript
  => dramatic composition, dramatic work
   => writing, written material
    => written communication, written language
     => communication
      => social relation
       => relation
        => abstraction
Sense 5
ledger, leger, account book, book of account, book
  => record
   => document
    => communication
     => social relation
      => relation
       => abstraction
Sense 6
book
  => section, subdivision
   => writing, written material
    => written communication, written language
     => communication
      => social relation
       => relation
        => abstraction
Sense 7
daybook, book, ledger
  => journal
   => book, volume
    => product, production
     => creation
      => artifact, artefact
```

```
         => object, physical object
          => entity, something
Sense 8
book
  => product, production
   => creation
    => artifact, artefact
     => object, physical object
      => entity, something
```

From these eight nominal senses, those numbered 1, 2, 7 and 8 share hypernyms from the node ‘product, production’ upwards, while those numbered 3, 4, 5 and 6 do so from the node ‘communication’. Given that there is no common hypernym for the eight senses of ‘book’, we can not link as easily as in the two previous examples the UW ‘book’ to the UW System. In this case, it is required to disambiguate which of the two chains of hypernyms should be used to link the UW or else if both of them are pertinent.

2.2 Linking Procedure

To summarise, and according to the previous examples, using WordNet for linking UWs associated with substantives requires the completion of the following steps:

1. To pair the high level concepts employed by the UW System with those present in the higher levels of the hyponym hierarchy of Wordnet.
2. To search the UWs in WordNet synsets and to analyse the hypernyms to which the synsets are linked to. In this process, four cases may arise:

   (a) The UW is not present in any synset. In this case the use of WordNet is of no help and the UW should be linked to UW System by other means. Given the large coverage of WordNet, this case should not be frequent.

   (b) The UW is monosemous. In this case, the chain of hypernyms is traversed until a node paired with an UW System concept is found.

   (c) The UW is polysemous and all its synsets share a common ancestor in their respective hypernym chains. In this case we proceed from that common node as in the previous case.

   (d) The UW is polysemous but there is not a single common ancestor for all the hypernym chains. In this case, extra information is needed for deciding which synset or set of synsets is chosen for linking the UW to the UW System.

2.3 Pairing WordNet 1.6 with the UW System

All first and second level concepts placed under ‘thing’ in the UW System have been paired with their counterparts in the WordNet 1.6 hyponym hierarchy. Frequently, the pairing is biunivocal, e.g. the UNL concept ‘abstract thing’, is paired with the WordNet node ‘abstraction’. Occasionally, a UNL concept is paired with several synsets in WordNet. For instance, the UNL concept ‘information’ has been found related to several of the senses catalogued in WordNet for this word:

```
5 senses of information
Sense 1
information, info
  => message, content, subject matter, substance
   =>  communication
    => social relation
     => relation
      => abstraction
Sense 2
data, information
  => collection, aggregation, accumulation, assemblage
   => group, grouping
Sense 3
information
  => cognition, knowledge
   => psychological feature
Sense 4
information, selective information, entropy
  => measure, measurement
   => magnitude
    => property
     => attribute
      => abstraction
Sense 5
information
  => accusation, accusal
   => charge, complaint
    => pleading
     => allegation, allegement
      => claim
       => assertion, averment, asseveration
        => declaration
         => statement
          => message, content, subject matter
           => communication
            => social relation
             => relation
              => abstraction
```

At least the first four senses of ‘information’ are related to the UNL concept, so it has been paired with the four synsets. The pairing has been done manually. The entity hierarchy proposed in [1] has been examined and WordNet counterparts have been found for all UNL concepts placed on the first and second levels below ‘thing’. A total of 73 high level concepts of the current UW System have been paired with their corresponding synsets in the WordNet substantive hierarchy.

All UWs associated with Spanish substantives in the Spanish-UNL dictionary have been automatically annotated with information coming from WordNet 1.6. Specifically, for each UW, its basic UW has been annotated with the set of synset identifiers in which the basic UW appears as a substantive. The set is empty for those basic UWs not found in WordNet, it contains a single identifier for monosemous basic UWs or several identifiers for polysemous basic UWs. For each synset identifier, the chain of hypernyms linking the synset with a top node of the WordNet hierarchy of substantives has been also retrieved. In the case of polysemous basic UWs, hypernym chains sharing a common ancestor have been collapsed into a single chain starting from the common ancestor.

If more than one hypernym chain remains after collapsing chains with common ancestors, the UW is considered semantically ambiguous and extra information is required for selecting a single chain. Two information sources are exploited: the semantic restrictions that may occur along with the basic UW for creating the UW and certain semantic features that may be present in the Spanish substantive.

2.4 Disambiguation by Means of Semantic Restrictions

If the ambiguous UW has semantic restrictions, we may disambiguate it processing the restrictions in very much the same way as the basic UW: we annotate the restrictions with their corresponding hypernym chains and look for a common ancestor between one of these chains and one of those resulting from annotating the basic UW.

Example. The UW ‘Malay(icl>language)’ is ambiguous after annotating its basic UW ‘Malay’. Two hypernym chains ending in ‘Malay’ share no common ancestor:

1. abstraction>relation>social_relation>communication> language>natural_language>Austronesian>Malayo-Polynesian>Western_Malayo-Polynesian>Malay
2. entity>life_form>person>person_of_color>Asian> Malay

If we now take the basic UW employed as restriction (‘language’) and annotate it with its hypernym chains, we end up with two chains after grouping chains with common ancestors:

(a) abstraction>relation
(b) psychological feature>cognition

Chain (1) shares a common ancestor (‘relation’) with chain (a), while chain (2) shares no ancestor neither with (a) nor with (b). Therefore, we can select chain 1) for locating 'Malay(icl>language)’ in the UW System and discard chain (2).

2.5 Disambiguation by Means of Semantic Features

When cataloguing Spanish substantives, two semantic features have been set for all of them because they are correlated with certain syntactic phenomena. The features ‘human’ and ‘animate’ are set to true for those substantives referring to human beings and animate entities respectively. This information is employed for reducing the number of hypernym chains in the following way: if the ‘human’ feature is set to true, all chains that do not include the node ‘person’ are discarded, if the ‘animate’ feature is set to true, all chains that do not include the node ‘living thing’ are discarded. In addition to these semantic features, the syntactic feature ‘countable noun’, employed for distinguishing mass nouns, is also taken into account: if ‘countable noun’ is set to true, all chains that do not include the node 'physical object’ are discarded, if it is set to false, then all chains not including the nodes 'abstraction’ or ‘substance’ are discarded.

Example. The UW ‘translator’ has been initially annotated with the hypernym chains:

1. abstraction>relation>social_relation>communication> written_communication>writing>coding_system>code>software>program>translator
2. entity>life_form>person>communicator>negotiator> mediator>interpreter

Given that its associated Spanish substantive (‘traductor’) sets the feature ‘human’ to true, we can disambiguate and select chain 2) because it contains the node ‘person’.

As for the order in which these information pieces is employed for disambiguation, semantic restriction are explored first, and only when they do not render a single hypernym chain the semantic features ‘human’ and ‘animate’ are taken into account. Eventually, the syntactic feature ‘countable noun’ is considered as a last resort.

2.6 Locating the UWs in the UW System

All UWs annotated with a single hypernym chain have been located in the UW System by linking them to the most specific chain node that is paired with a UNL concept.

Example. The UW ‘Indonesian’ is annotated with the following hypernym chain: entity > life_form > person > person_of_color > Asian > Indonesian.

Starting from its most specific node and moving upwards, the intermediate node ‘person’ is the first one that is paired to its homonymous UNL concept. Therefore, ‘Indonesian’ is located in the UW System by the following link: ‘Indonesian’—icl→ ‘person’

3 Results

14,911 UWs associated to Spanish substantives have been processed by the method just described. The initial annotation of these UWs with hypernym chains produced the following results:

Table1. Initial annotation results.

UWs not found in WordNet 1.6	1,447 (9.7%)
UWs Annotated with a single chain	7,863 (52.7%)
UWs Annotated with several chains	5,601 (37.5%)

The disambiguation mechanisms have been able to resolve 2,480 UWs (44,2%) from the total of 5,601 initially ambiguous UWs. Every non-ambiguous UW (7,863 plus 2,480) has been located in the UW System by linking it with one of the 73 high level concepts.

The final figures concerning the task of locating by automatic means the UWs associated to Spanish substantives in the Spanish-UNL dictionary rendered these final results:

Table 2. Final linking results.

UWs linked to the UW System	10,343 (69.3%)
UWs not linked because of ambiguity	3,121 (20.9%)
UWs not linked because not found	1,447 (9.7%)

3.1 Analysis of the results

Approximately ten percent of the UWs associated to Spanish substantives have not been found in WordNet 1.6. An analysis of this ten percent shows that most of these UWs fall into the following categories:

1. Proper names (‘Alphonso’, ‘Louis’, ‘IAS’)
2. Discrepancies in capitalisation (‘Internet’ versus ‘internet’ in WordNet 1.6)
3. Discrepancies in the use of separators (‘leather jacket’ and ‘sister-in-law’ versus ‘leatherjacket’ and ‘sister_in_law’ in WordNet 1.6).
4. Use of inflected forms as UWs: ‘begs’, ‘studies’, ‘gratting’.
5. Use of phrases as UWs: ‘garden wall’, ‘day pupil’, ‘small tail’ o ‘student music group’.

UWs included in categories 2 y 3 may be easily solved by a more flexible searching mechanism. Phrasal UWs may be syntactically analysed and their heads used instead of the whole phrase for linking purposes. Proper names require other resources such as lists of personal proper names and institutions while UWs included in category number 4 require careful examination.

Generally speaking, the UWs that remained ambiguous lack of semantic restrictions or have a very general restriction such as ‘icl>thing’. These UWs may be disambiguated manually or their restrictions completed or make more precise. As for the quality performance of the disambiguation mechanisms, an initial inspection of the results allows to put forward the following considerations:

1. Using the semantic restrictions (when they are not extremely general) and the semantic features ‘human’ and ‘animate’ largely produces the selection of the correct hypernym chain.
2. Disambiguation based on the ‘countable noun’ feature is less reliable.

Disambiguation based on grouping chains sharing a common ancestor may lead to very general links in the UW System, since in the worst case the common ancestor is the top concept ‘thing’ and then the UW is linked to this general concept instead than to a more specific one, which is contrary to the main goal of the UW System.

4 Conclusions

This paper has presented a simple and effective way of using a well-known, freely available lexical resource such as WordNet for automating at least partially the creation of the UW System. Taking advantage of the conceptual similarities between WordNet and the UW System, we have mapped the upper levels of the UNL entity hierarchy to the upper levels of the hyponym-hypernym relation defined in WordNet. This has open the possibility of automatically link a substantial part of the UWs associated with substantives in the Spanish-UNL dictionary with the concepts of the UW System.

The results obtained encourage a further development of this approach, deepening the mapping between UNL concepts and the WordNet hierarchy and exploring novel ways of disambiguating hypernym chains.

References

1. Uchida, I. "Master Dictionary specifications. Version 1.0", *UNDL Foundation*, October 2000.
2. Uchida, I. "The UNL UW System. Version 1.0", *UNDL Foundation*, January 2001.
3. Fellbaum, C. *WordNet: An Electronic Lexical Database*, MIT Press, 1998.

Automatic Generation of Multilingual Lexicon by Using Wordnet

Nitin Verma and Pushpak Bhattacharyya

Department of Computer Science and Engineering, I.I.T. Bombay,
{nitinv,pb}@iitb.ac.in

Abstract. A lexicon is the heart of any language processing system. Accurate words with grammatical and semantic attributes are essential or highly desirable for any application- be it machine translation, information extraction, various forms of tagging or text mining. However, good quality lexicons are difficult to construct requiring enormous amount of time and manpower. In this paper, we present a method for automatically generating multilingual Universal Word (UW) dictionaries (for English, Hindi and Marathi) from an input document-making use of English, Hindi and Marathi WordNets. The dictionary entries are in the form of Universal Words (UWs) which are language words (primarily English) concatenated with disambiguation information. The entries are associated with syntactic and semantic properties- most of which too are generated automatically. In addition to the WordNet, the system uses a word sense disambiguator, an inferencer and the knowledge base (KB) of the Universal Networking Language which is a recently proposed interlingua. The lexicon so constructed is sufficiently accurate and reduces the manual labor substantially.

1 Introduction

Construction of good quality lexicons enriched with syntactic and semantic properties for the words is time consuming and manpower intensive. Also word sense disambiguation presents a challenge to any language processing application, which can be posed as the following question: *given a document* ***D*** *and a word* ***W*** *therein, which sense* ***S*** *of* ***W*** *should be picked up from the lexicon?*. It is, however, a redeeming observation that a particular **W** in a given **D** is mostly used in a single sense throughout the document. This motivates the following problem: *can the task of disambiguation be relegated to the background before the actual application starts? In particular, can one construct a* ***Document Specific Dictionary*** wherein single senses of the words are stored?

Such a problem is relevant, for example, in a machine translation context [1]. For the input document in the source language, if the *document specific dictionary* is available a-priory, the generation of the target language document reduces to essentially syntax planning and morphology processing for the pair of languages involved. The WSD problem has been solved before the MT process starts, by putting in place a lexicon with the document specific senses of the words.

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 380–391.

In this paper we describe a methodology for automatic generation of *document specific UW dictionaries* (particularly for English, Hindi, and Marathi)- by making use of the *English, Hindi* and Marathi WordNets} [2, 5, 6, 8]. The methodology described in this paper for generating document specific English-UW dictionaries has an improved performance for adjectives and adverbs over [3].

Section 2 briefly describes the UNL system. The format of L-UW dictionary is described in section 3. Section 4 illustrates the method of *document-specific* English-UW dictionary generation. The method of generating Hindi-UW dictionary by using the *Hindi WordNet* is described in section 5. Section 6 gives the future directions for improving the performance of multilingual lexicon generation system.

2 Universal Networking Language

UNL [7] is an interlingua for machine translation [1] and is an attractive proposition for the multilingual context. In this scheme, a source language sentence is converted to the UNL form using a tool called the *EnConverter* [7]. Subsequently, the UNL representation is converted to the target language sentence by a tool called the *DeConverter* [7]. The sentential information in UNL is represented as a hyper-graph with concepts as nodes and relations as arcs. The UNL graph is a hyper-graph because the node itself can be a graph, in which case the node is called a *compound word* (CW). Figure 1 represents the sentence *John eats rice with a spoon.*

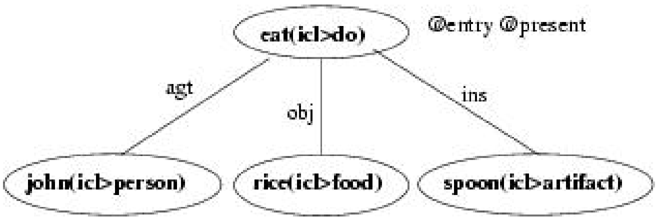


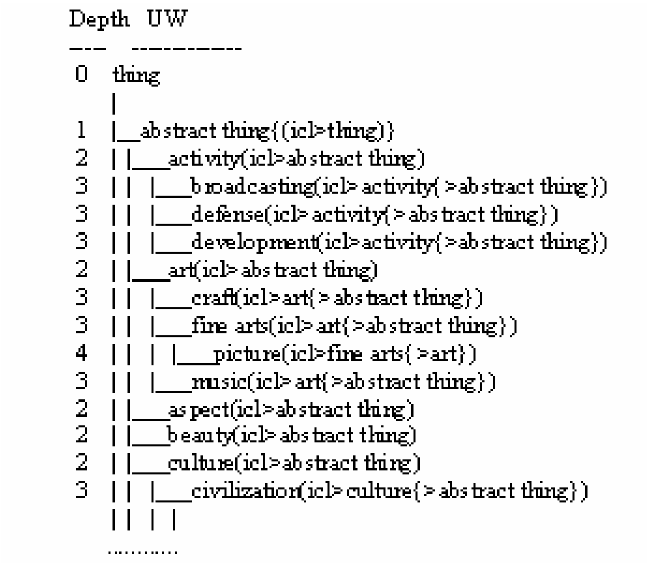
Fig. 1. UNL Graph of *John Eats Rice With A Spoon*.

In the above graph the arcs denoting *agt* (agent), *obj* (object) and *ins* (instrument) are the relation labels as defined in the UNL specification. This graph is represented as a set of directed binary relations between two concepts present in the sentence. The relation *agt* stands for *agent*, *obj* for *object* and *ins* for *instrument.* The binary relations are the basic building blocks of the UNL system, which are represented as strings of 3 characters or less each. There are 41 relations in the UNL system.

In the above figure the nodes such as *eat(icl>do)*, *John(iof>person)*, *rice(icl>food)* and *spoon(icl>artifact)* are the *Universal Words (UW)*. These are language words with *restrictions* in parentheses. *icl* stands for *inclusion* and *iof* stands for *instance* of. UWs can be annotated with attributes which provide further information about how the concept is being used in the specific sentence. Any of the three restriction labels, *viz., icl, iof* and *equ*, is attached to an UW for restricting its sense. For example, two senses of *state* will be represented in the UNL system in the following way:

- *state(icl>express)* to express something clearly and carefully.
- *state(icl>country)* a politically organized body of people under a single government.

A UW is created using the *specifications* of the *UNL Knowledge Base (KB)*. UNL KB organizes the UWs in a *hierarchy*. A *part* of the UW hierarchy for *nouns* in the UNL KB is shown in Figure 2 which is self-explanatory.



```
Depth  UW
---   ----------
 0   thing
     |
 1   |__abstract thing{(icl>thing)}
 2   | |___activity(icl>abstract thing)
 3   | |  |___broadcasting(icl>activity{ >abstract thing})
 3   | |  |___defense(icl>activity{>abstract thing})
 3   | |  |___development(icl>activity{>abstract thing})
 2   | |___art(icl>abstract thing)
 3   | |  |___craft(icl>art{>abstract thing})
 3   | |  |___fine arts(icl>art{>abstract thing})
 4   | |  |  |___picture(icl>fine arts{>art})
 3   | |  |___music(icl>art{>abstract thing})
 2   | |___aspect(icl>abstract thing)
 2   | |___beauty(icl>abstract thing)
 2   | |___culture(icl>abstract thing)
 3   | |  |___civilization(icl>culture{>abstract thing})
     | |  |  |
     ………..…
```

Fig. 2. Hierarchy of *noun* UW's in the UNL KB (a snapshot).

For verbs, the hierarchy is not so deep. All the verbs are organized under three categories, *viz.*, *do*, *occur* and *be*. The first two are *aktionstat* verbs and the last one is the set of *stative verbs*. The adjective, adverb and preposition hierarchies too are quite shallow. The adjectives that are both *attributive* and *predicative* are given the restriction *(aoj>thing)*, where *aoj* is a semantic relation denoting *attribute* of the object and *thing* denotes a nominal concept. The adjectives which are only *predicative* are given the restriction *(mod>thing)* where *mod* is the *modifier* relation. The adverbs are uniformly expressed through *(icl>how)*.

3 L-UW Dictionary

The dictionary maps the *words* of a natural language to the *universal words* of the UNL system [9]. For example:

- [kuttaa]"dog(icl>mammal)" (**...** attributes **...**)
- [bhaukaa]"bark(icl>do)" (**...** attributes **...**)

- are the entries in a Hindi-UW dictionary [10]. Similarly:
- [dog]"dog(icl>mammal)" (**...** attributes **...**)
- [bark]"bark(icl>do)" (**...** attributes **...**)

are the entries is an English-UW dictionary. When the sentence *The dog barks* is given to an UNL-based English-Hindi MT system, the Uws *dog(icl>mammal)* and *bark(icl>do)* are picked up. These are disambiguated concepts different from other senses of *dog* and *bark*, for example the *pursue* sense of *dog* (*dog(icl>do)*) and the *skin of the tree* sense of *bark* (*bark(icl>skin)*). *If the L-UW dictionary contains only document specific UWs, the analyser and the generator systems do not commit error on account of WSD*.

The *attributes* attached to each entry in the L-UW dictionary are the *lexical, grammatical,* and *semantic* properties of the language specific words (*NOT of the UWs*). The syntactic attributes include the word category- *noun, verb, adjectives, adverb etc*. and attributes like *person* and *number* for nouns and *tense* for verbs. The *Semantic Attributes* are derived from an *ontology*. Figure 3 shows a part of the *ontology* used for obtaining semantic attributes [9].

4 Automatic Generation of English-UW Dictionary

For generating the document specific English-UW dictionary we use the *English WordNet, a WSD System*, the *UNL KB* and an *inferencer*. The approach is *Knowledge Based* [12]. The UNL KB as shown in figure 2 is stored as a *mysql* database. The table *UNL-KB-table* in figure 4 shows a part of this storage structure for nouns.

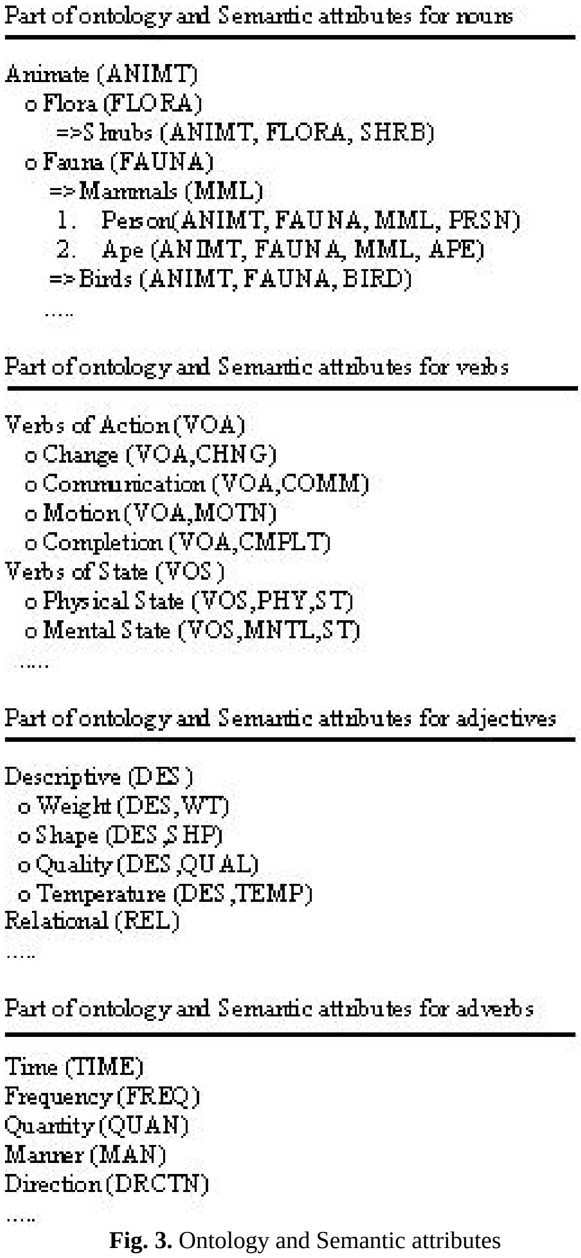
The *word sense disambiguator* (integrated with our lexicon generation system) uses a method called as *Soft Word Sense Disambiguation* [4]. In soft word sense disambiguation method, the *sense disambiguation system* does not commit to a *particular sense* but it gives us a *set of senses* which are not necessarily orthogonal or mutually exclusive. The senses are expressed by the WordNet synsets and are arranged according to their relevance in the given context. A detailed description of *soft word sense disambiguation* method is given in [4].

Soft word sense disambiguation system attaches a *confidence value* (relevance score or probability) with every relevant sense of a word present in the document. In the final English-UW dictionary the entries with the low confidence value of their sense are disabled by placing a semicolon at their beginning. Everything after a semicolon (in a particular line) is ignored by the EnConverter automatically by the lexicon generation system and the one with the highest score is kept enabled. This method still keeps the dictionary *document-specific* and gives a *flexibility* to the lexicographers to enable an appropriate sense in the dictionary generated.

The steps involved in the generation of *document specific English-UW dictionary* are as follows.

4.1 POS Tagging and Sense Disambiguation

The document is passed to the word sense disambiguator [4], which gives us a *part of speech* and *sense tagged* document. The output of this step is a list of entries in the



```
Part of ontology and Semantic attributes for nouns

Animate (ANIMT)
  o Flora (FLORA)
      =>Shrubs (ANIMT, FLORA, SHRB)
  o Fauna (FAUNA)
     => Mammals (MML)
       1.  Person(ANIMT, FAUNA, MML, PRSN)
       2.  Ape (ANIMT, FAUNA, MML, APE)
     => Birds (ANIMT, FAUNA, BIRD)
     .....

Part of ontology and Semantic attributes for verbs

Verbs of Action (VOA)
  o Change (VOA,CHNG)
  o Communication (VOA,COMM)
  o Motion (VOA,MOTN)
  o Completion (VOA,CMPLT)
Verbs of State (VOS)
  o Physical State (VOS,PHY,ST)
  o Mental State (VOS,MNTL,ST)
 .....

Part of ontology and Semantic attributes for adjectives

Descriptive (DES)
 o Weight (DES,WT)
 o Shape (DES,SHP)
 o Quality (DES,QUAL)
 o Temperature (DES,TEMP)
Relational (REL)
.....

Part of ontology and Semantic attributes for adverbs

Time (TIME)
Frequency (FREQ)
Quantity (QUAN)
Manner (MAN)
Direction (DRCTN)
.....
```

Fig. 3. Ontology and Semantic attributes

format **Word:POS:WSD**, where POS stands for *part of speech* and WSN indicates *WordNet sense number*. The *syntactic* attributes are obtained at this stage.

4.2 Generation of UW's

The WN and UNL KB are used to generate the restriction for the word. If the word is a noun, the WN is queried for the hypernymy for the marked sense. All the Hypernymy ancestors $\mathbf{H_1}$, $\mathbf{H}_2$, **...**, $\mathbf{H_n}$ of **W** up-to the *unique beginner* are collected. If $W(icl{>}H_i)$ exists in the UNL KB, it is picked up and entered in the dictionary. If not, $W(icl{>}H_1)$ is asserted as the dictionary entry.

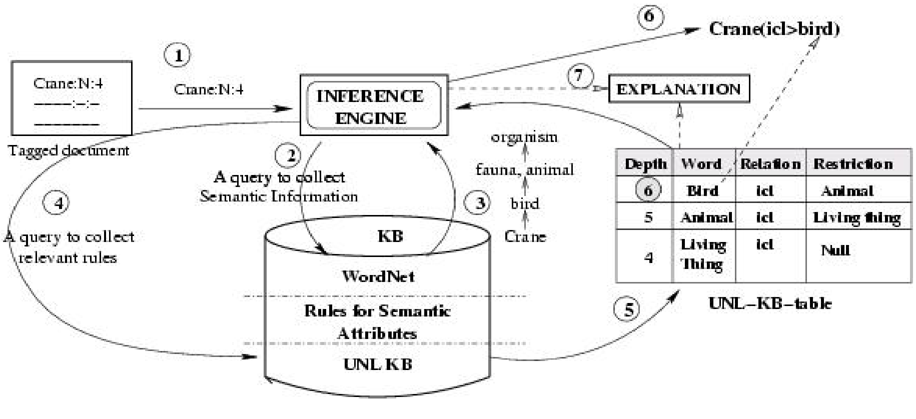
For example, for *crane* the *bird*-sense gives the hypernyms as *bird, fauna, animal, organism* and finally *living_thing*. *crane(icl>bird)* becomes the dictionary entry in this case. Figure 4 illustrates this process.

For verbs, the hypernymy ancestors are collected from the WN. If these include concepts like *be, hold, continue etc.*, then we generate the restriction *(icl>be)* (case of *be* verb). If not, the corresponding *nominal word* (for example, the nominal word for the verb *rain* is *rain* itself) of the verb is referred to in the WN. If the hypernyms of the nominal word include concepts like *phenomenon, natural_event etc.*, then we generate the restriction *(icl>occur)* signifying an *occur* verb. If both these conditions are not satisfied, then the restriction *(icl>do)* is generated.

For adjectives, use is made of the *is_a_value_of* semantic relation [8] in the WN. For example, for the adjective *heavy* the above relation links it to *weight*. If this relation is present then the restriction *(aoj>thing)* is generated. Else we generate *(mod>thing)* (please refer back to Section 2).

For adverbs, (icl>how) is by default generated, as per the specifications of the UNL system.

4.3 Creating the Semantic Attributes



Depth	Word	Relation	Restriction
6	Bird	icl	Animal
5	Animal	icl	Living thing
4	Living Thing	icl	Null

Fig. 4. Universal Word Creation: an example

The semantic attributes are generated from a *rule-base*, linking the lexico-semantic relations of the WN with the semantic properties of the word senses. To take an example, **if** the *hypernym* is *organism*, **then** the attribute *ANIMT* signifying *animate* is generated. We have more than 5000 such rules in the rule base. The tables in the figure *rules* shows sample of such rules for all the POS words.

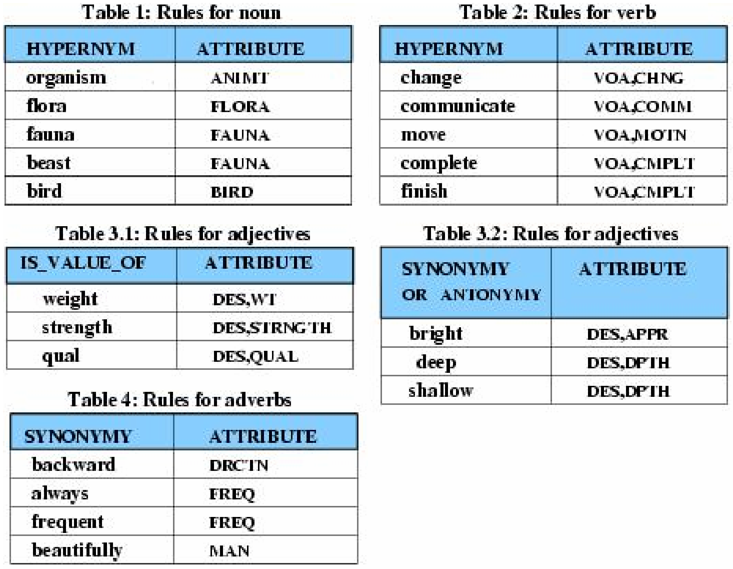


Table 1: Rules for noun

HYPERNYM	ATTRIBUTE
organism	ANIMT
flora	FLORA
fauna	FAUNA
beast	FAUNA
bird	BIRD

Table 2: Rules for verb

HYPERNYM	ATTRIBUTE
change	VOA,CHNG
communicate	VOA,COMM
move	VOA,MOTN
complete	VOA,CMPLT
finish	VOA,CMPLT

Table 3.1: Rules for adjectives

IS_VALUE_OF	ATTRIBUTE
weight	DES,WT
strength	DES,STRNGTH
qual	DES,QUAL

Table 3.2: Rules for adjectives

SYNONYMY OR ANTONYMY	ATTRIBUTE
bright	DES,APPR
deep	DES,DPTH
shallow	DES,DPTH

Table 4: Rules for adverbs

SYNONYMY	ATTRIBUTE
backward	DRCTN
always	FREQ
frequent	FREQ
beautifully	MAN

Fig. 5. Rules For Generating Semantic Attributes

For nouns, Table 1 (Rules for noun) in figure 5 is used to generate semantic attributes. The first entry corresponds to the rule: IF hypernym = organism THEN generate ANIMT attribute. Semantic attributes for verbs are obtained in the same way by using Table 2 (Rules for verbs).

For adjectives, Tables 3.1 and 3.2 are used. The first entry in the table 3.1 corresponds to the rule: **IF** *input_word* IS_A_VALUE_OF(*weight*) **THEN** the attributes *DES,WT* signifying *weight* (classified in *descriptive* category) are generated. The first entry in the Table 3.2 is interpreted as: **IF** *input_word* is (SYNONYM_OF(*bright*) OR ANTONYM_OF(bright)) **THEN** the attributes *DES,APPR* (*descriptive, appearance*) are generated.

4.4 Experiments and Results

We have tested our system on documents from various domains like agriculture, science, arts, sports *etc*. each containing about 1000 words. We have *measured* the *performance* of this system by calculating its *precision* in every POS category. The precision is defined as:

$$precision = \frac{number\ of\ entries\ correctly\ generated}{total\ entries\ generated}.$$

Figure 6 shows the results.

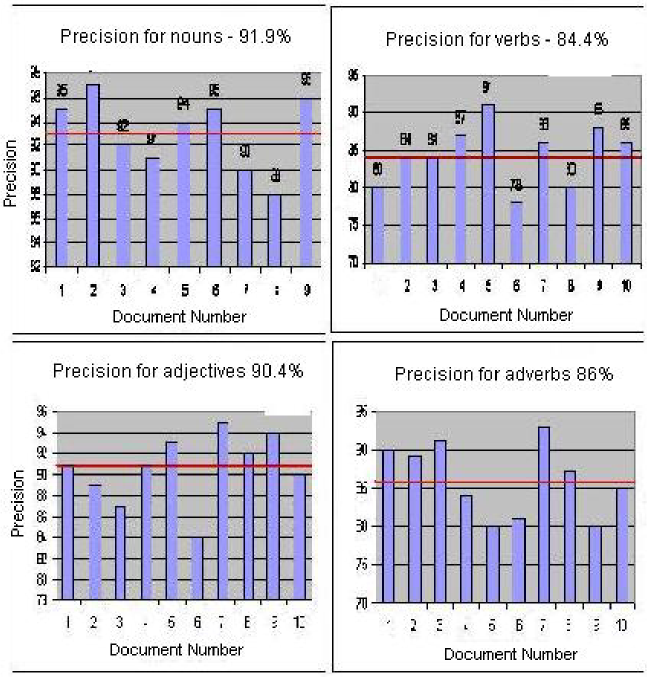


Fig. 6. Experiments And Results

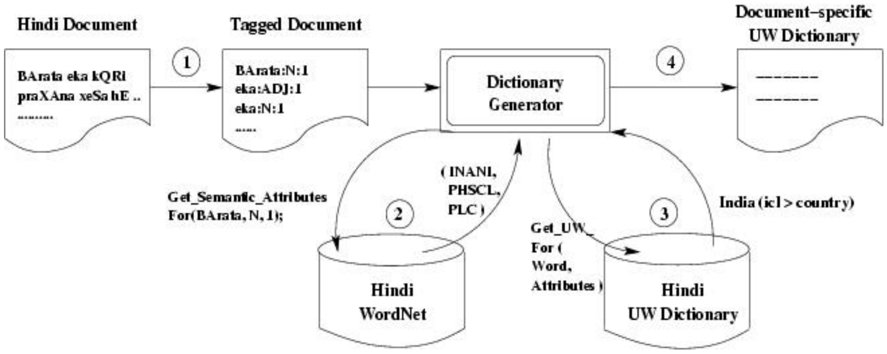


Fig. 7. Document-Specific Hindi-UW Dictionary Generation

The average precision for nouns is **93.9**%, for *verbs* **84.4**%, for *adjectives* **90.6**% and for *adverbs* **86**%. The system is being routinely used in our work on machine translation in a tri-language setting (*English*, *Hindi* and *Marathi*). It has reduced the burden of lexicography considerably. The incorrect entries- which are not many- are corrected manually by the lexicographer. A snapshot of document specific English-UW dictionary generated (the entries with the low score value are disabled *automatically* by placing a semicolon at the beginning) after running our system on a document containing the following paragraph is shown below.

Modern agriculture depends heavily on engineering and technology and on the biological and physical sciences. Irrigation, drainage, conservation, and sanitary engineering- each of which is important in successful farming- are some of the fields requiring the specialized knowledge of agricultural engineers.

– [Modern] {} "modern(icl>character)" (N, INANI, PROP, ACT, COMM, ABS) <E,0,0>; SCORE=0.917893
– ; [Modern] {} "modern(icl>person)" (N, PROP, ANIMT, FAUNA, MML, PRSN, PHSCL) <E,0,0>; SCORE=0.901949
– [agriculture]{}"agriculture(icl>industry)"(N,INANI,EVENT,ABS)<E,0,0>; SCORE=0.931336
– ; [agriculture] {} "agriculture(icl>business)" (N, INANI, EVENT, ABS) <E,0,0>; SCORE=0.90433
– [depend] {} "depend(icl>be(aoj>thing{,obj>thing}))" (VRB, CONT, VOS-PHY-ST) <E,0,0>; SCORE=0.937532
– ; [depend] {} "depend(icl>trust{>be}(aoj>thing))" (VRB, VOA-COGN, VOA-POSS, VOS-MNT-ST) <E,0,0>; SCORE=0.923279
– [engineering] {} "engineering(icl>subject)" (N, INANI, PSYFTR, ABS) <E,0,0>; SCORE=0.924104
– ;[engineering] {} "engineering(icl>structure)" (N, INANI, OBJCT, ARTFCT, PHSCL) <E,0,0>; SCORE=0.90438
– [technology] {} "technology(icl>subject)" (N, INANI, PSYFTR, ABS) <E,0,0>; SCORE=0.924104
– ; [technology] {} "technology(icl>exercise)" (N, INANI, EVENT, ABS) <E,0,0>; SCORE=0.894572
– [biological] {} "biological(mod<thing)" (ADJ, REL) <E,0,0>; SCORE=0.924506
– [physical] {} "physical(mod<thing)" (ADJ, DES, QUAL) <E,0,0>; SCORE=0.924204
– [scienc] {} "science(icl>power)" (N, INANI, PSYFTR, ABS) <E,0,0>; SCORE=0.926118
– ; [scienc] {} "science(icl>subject)" (N, INANI, PSYFTR, ABS) <E,0,0>; SCORE=0.898305
– [Irrigation] {} "irrigation(icl>act)" (N, INANI, PROP, EVENT, ABS) <E,0,0>; SCORE=0.926247
– [conservation] {} "conservation(icl>protection)" (N, INANI, EVENT, ABS) <E,0,0>; SCORE=0.919366

5 Semi-Automatic Generation of Hindi-UW Dictionary

The prime resources we use for generating document specific Hindi-UW dictionaries are *Hindi-WordNet* [5] [6] and a Hindi UW dictionary which contains about 80,000 entries. The difficulty in automatic generation of document specific Hindi-UW dictionary is the absence of a *part-of-speech tagger* and a *Word Sense Disambiguator*. In our method we generate dictionary entries for all possible parts of speech and for all possible senses of the input word (present in the Hindi WordNet). Once the dictionary is generated, the irrelevant entries are disabled by the lexicographer by placing a semi-colon at the beginning of the entry. The document-specific dictionary generation in this case is not fully automatic (because of absence of POS tagger and WSD system for Hindi) like English-UW dictionary generation, but it has reduced the manual efforts required for Hindi lexicon generation substantially. The methodology of generating document specific Hindi-UW dictionary is described in the sub-section below, and the process is also shown in the Figure 7.

5.1 Methodology Used for Dictionary Generation

1. For every input word we use *Hindi-WordNet API* to obtain all possible *parts of speech* and all possible senses (present in the Hindi WordNet) for that word. In this step, an intermediate *tagged document* is generated in which the entries are in the format- *Word:POS:SenseNo* (shown in figure 7).
2. In Hindi WordNet every *synset* is linked to an *ontology node* (figure 3, which makes it easy for us to derive *semantic attributes* for a word (present in the Hindi WordNet) in its given POS and sense number. Hindi WordNet design has special support for linking *synsets* with the *ontology nodes* [5] [6]. For every *Word:POS:SenseNo* pair in the *tagged document*, Hindi WordNet is queried (by using Hindi WordNet API) to obtain the *semantic attributes*.
3. For generating an appropriate UW, we use a Hindi UW dictionary which contains about 80,000 entries. For the efficient retrieval of the UWs, we have stored the Hindi UW dictionary in a MySQL database table having the structure (*Hindi Word, UW, POS, attributes*). After obtaining the *semantic attributes* from the Hindi WordNet database, the Hindi UW dictionary database is queried to obtain an appropriate UW.
4. After collecting appropriate Semantic Attributes in step 2 and obtaining UWs from step 3, the *document-specific* Hindi-UW dictionary is generated. In this step the irrelevant entries (entries with irrelevant POS and Sense) are disabled and the incorrect ones are corrected manually by the lexicographer. This process reduces the burden of lexicography considerably. A snapshot of Document specific Hindi UW dictionary generated for a document containing the following paragraph on Indian Agriculture (Written in phonetic font format) is shown below. The incorrect entries are marked by a *.

“bhaarat eka krishi pradhaan desh hai. yahaan kii aadhe se adhika janasankhyaa gaavon mai nivaas karatii hai. jinakaa mukhya vyavasaaya krishi hai. swatantrata ke baad bhaarat ne krishi ke kshetra mai bahut vikaas kiyaa hai”.

– [bhaarat] {} "India(icl>country)" (N, INANI, PHSCL, PLC) <H,0,0>;
– [eka] {} "a(icl>number)" (ADJ, DES, NUM) <H,0,0>;
– * [eka] {} "unity(scn>mathematics)" (N, INANI, ABS, MATHS) <H,0,0>;
– [krishi] {} "agriculture(icl>farming)" (N, INANI, ABS, ACT, PHSCLACT) <H,0,0>;
– [pradhaan] {} "cardinal" (ADJ, DES, QUAL) <H,0,0>;
– [hai] {} "have(icl>be)" (V, VE) <H,0,0>;
– * [se] {} "since" (ADV) <H,0,0>;
– [adhika] {} "full(equ>most)" (ADJ, DES, QUAN) <H,0,0>;
– * [janasankhyaa] {} "population(fld>biology)" (N, INANI, GRP) <H,0,0>;
– [nivaas] {} "lodging(icl>accommodation)" (N, INANI, PHSCL, PLC) <H,0,0>;
– [mukhya] {} "arch(icl>chief)" (ADJ, DES, QUAL) <H,0,0>;
– [vyavasaay] {} "firm(icl>shop)" (N, INANI, ABS, ACT, OCP) <H,0,0>;
– [krishi] {} "agriculture(icl>farming)" (N, INANI, ABS, ACT, PHSCLACT) <H,0,0>;
– [hai] {} "have(icl>be)" (V, VE) <H,0,0>;
– [baad] {} "later(icl>afterwards)" (ADV) <H,0,0>;
– [bhaarat] {} "India(icl>country)" (N, INANI, PHSCL, PLC) <H,0,0>;
– [krishi] {} "agriculture(icl>farming)" (N, INANI, ABS, ACT, PHSCLACT) <H,0,0>;
– [bahut] {} "abundant(icl>lot)" (ADJ, DES, QUAN) <H,0,0>;
– [vikaas] {} "advance(icl>development)" (N, INANI, ABS, ACT) <H,0,0>;

6 Conclusion and Future Work

In the machine translation process using UNL as an *interlingua*, the burden of lexicography has been reduced considerably by using the *multilingual lexicon generation system*. The system is being routinely used in our work on machine translation in a tri-language setting (English, Hindi, and Marathi). The incorrect entries- which are not many are corrected manually.

Efforts are also on to implement the automatic lexicon generation system for *Marathi* language. The architecture of *Marathi WordNet* is same as that of *Hindi WordNet*. Like Hindi WordNet- every *Marathi synset* is linked to an *ontology node* (shown in Figure 3). The method of generating *semantic attributes* for Marathi words is same as that of Hindi (described in Section 5.1). At present we are making efforts to prepare a UNL KB dedicated to Marathi language which will enable us to automatically generate Universal Words for Marathi-UW dictionary.

The presence of *part of speech tagger* and *word sense disambiguator* for Hindi and Marathi will improve the performance of multi-lingual lexicon generation by many folds. Our future work will also be directed towards the implementation of *part of speech* tagger and *word sense disambiguator* for Hindi and Marathi languages.

References

1. W. John Hutchins and Harold L. Somers. “An Introduction to Machine Translation”. *Academic Press*, 1992.
2. Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K. “Five papers on WordNet”. Available at URL: http://clarity.princeton.edu:80/~wn/ , 1993.
3. N. Verma and P. Bhattacharyya. “Automatic lexicon generation through WordNet.” *Global WordNet Conference*, Jan 2004.
4. G. Ramakrishnan, Prithviraj, A. Deepa, P. Bhattacharyya and S. Chakrabarti. “Soft Word Sense Disambiguation”. *Global WordNet Conference*, Jan 2004. Available at URL: www.cse.iitb.ac.in/~nitinv/softWSD.ps.
5. S.Jha, D. Narayan, P. Pande and P. Bhattacharyya. “A WordNet for Hindi”. *Workshop on Lexical Resources in Natural Language Processing*, Hyderabad, India, January, 2001.
6. Dipak Narayan, Debasri Chakrabarty, Prabhakar Pande and P. Bhattacharyya. “An Experience in Building the Indo WordNet- a WordNet for Hindi”. *International Conference on Global WordNet (Global Wordnet 2002),* Mysore, India, January, 2002.
7. “The Universal Networking Language (UNL) Specifications”, *United Nations University*. Available at URL: http://www.unl.ias.unu.edu/unlsys/ , July 2003.
8. Christiane Fellbaum. WordNet: “An Electronic Lexical Database”. *The MIT Press*, 1998.
9. P. Bhattacharyya. “Multilingual information processing using UNL”. in *Indo UK workshop* on Language Engineering for South Asian Languages LESAI, 2001.
10. Shachi Dave, Jignashu Parikh and Pushpak Bhattacharyya, “Interlingua Based English Hindi Machine Translation and Language Divergence”, *Journal of Machine Translation*, Volume 17, September, 2002. (to appear).
11. Hiroshi Uchida and Meiying Zhu. “The Universal Networking Language beyond Machine Translation”. *UNDL Foundation*, September 2001.
12. Adrian A. Hopgood. “Knowledge-Based Systems for Engineers and Scientists”. *CRC Press LLC*, 1992.

METHODOLOGIES

Gradable Quality Translations through Mutualization of Human Translation and Revision, and UNL-Based MT and Coedition

Christian Boitet

GETA, laboratoire CLIPS,
385 rue de la bibliothèque - BP 53, 38041 Grenoble Cedex 9, France
Christian.Boitet@imag.fr

Abstract. Translation of specialized information for end users into many languages is necessary, whether it concerns agriculture, health, etc. The quality of translations must be gradable, from poor for non-essential parts to very good for crucial parts, and translated segments should be accompanied with a measured and certified "quality level". We sketch an organization where this can be obtained through a combination of "mutualized" human work and automatic NLP techniques, using the UNL language of "anglosemantic" graphs as a "pivot". Building the necessary multilingual lexical data base can be done in a mutualized way, and all these functions should be integrated in a "Montaigne" environment allowing users to access information through a browser and to switch to translating or postediting and back.

1 Introduction

Translation of specialized information into many languages is necessary, notably in agriculture, but also for health and other domains, because it is often crucial for final users, who don't master the source language. Quality should be very high, at least for the crucial parts. In many cases, also, it is urgent to use the information, and only automated translation could offer a solution. At the same time, resources are scarce, especially to produce high quality translations. Does that mean that nothing can be done? No, of course.

The first idea which comes to mind is to "mutualize" the translation effort. That becomes possible thanks to the wide availability of Internet. There is always a minority of targeted readers who understand the source language, and could produce good translations. Also, they would translate only a fraction of their time, so that, even with machine helps which may be developed by and by, it is reasonable to assume that not every part of every document could be translated in this way. Why not, then, use "rough" machine translation (MT), or even "active reading helps" (annotations of the source text by possible translations of words, terms and even phrases), and have human readers decide on which crucial parts are difficult to understand when presented in this way, and improve them?

We claim that, in this and similar domains, the quality of translations theoretically can and practically must be gradable, from poor to very good. Translations of each

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 395–412.

fragment (down to the level of a sentence) should be accompanied with a measured and certified "quality level". We propose an organization where this can be obtained by combining "mutualized" human work and automatic NLP techniques, using the UNL language of "anglosemantic" graphs as a "pivot".

We begin by assessing in more detail this type of "translational situation" and show why gradable multitarget translation of agricultural information is necessary. We then present the first part of our design, which relies only on mutualized human work, made possible by having the documents and the lexicons on a central server, while readers/translators share mutualized versions of translation aid tools such as a translation editor, a lexical data base, and a translation memory. Then we describe more advanced functionalities, to be integrated in the same framework as they become available. At the end, we should have a multilingual TA/MT system, where the MT part is also inherently designed to be helped by humans. Using the UNL language of "anglosemantic" graphs as a "pivot" is the key, because UNL graphs are understandable and can be directly improved by college level persons using graphical editors and presentations localized for each language.

2 Necessity of Gradable Multitarget Translation

The "translational situation" envisaged is characterized by the type of information, the intended readers, the available resources, and various constraints on the result.

Original information

The information to be translated is:

- *mainly monolingual,*
- *specialized & important,*
- *updated frequently,*
- *large.*

This is true for agriculture, health, weather, traffic, cultural heritage, crisis situations, human rights, etc. If the source information is not monolingual, it is usually in 2 or 3 languages at most (e.g., Hindi and English in India for agriculture, or English and French in Canada for weather bulletins).

The documents may each be quite small. A typical weather bulletin in English has 100-200 words, a 2-page leaflet in Word (Times 12, single-spaced) contains typically 1000 words or less. Note however that a standard "translator's page" is 250 words long (1400 characters, double-spaced) and that, in a professional context, without machine aids but a text editor and a dictionary, it takes 1 hour to produce a draft output and 20 minutes to polish it to obtain what is judged as "professional quality". Hence, a 1000 word leaflet would cost an average of 160 hours to translate and polish into 30 languages ($5^{h}20$ per language).

Frequent updates lead to huge quantities. In Canada, for example, each weather station updates its bulletin every 4 hours. That adds up to 20 million words a year in

English, 10 million in French. The METEO system handling these translations since 1978 (Chandioux 1988) replaces about 100 translators (in 1600 hours per year, a translator can translate and polish about 300000 words).

Readers

Most intended readers:

- *are not at ease in the source language,* even if it is English, especially whan it comes to technical terms and descriptions of procedures;
- *use various languages* (hundreds in India, may be than 30 in the territory of Thailand, Burma, Cambodia, Laos, and Vietnam);
- *can increasingly access the web.*

Indeed, although it is believed that all Indians know English, official figures say that only about 5% of the population really masters it to the point of reading and understanding administrative or technical information. In other parts of South-East Asia (such as Thailand), a large majority of farmers don't master the source languages of the information at a sufficient level, but speak a variety of dialects or other languages. Translation must hence be into N target languages, with N anywhere from 20 (Europe) to maybe 300 (India).

The only good news on the readers side is that they are increasingly connected to Internet. The harware is there, and quite cheap, and browsers can display information in all Unicode-supported languages.

Resources

On the resource side, the main points are:

- *the scarcity of competent translators*
- *the scarcity of financial resources*
- *often, the absence of commercial MT systems*

A main characteristics of agriculture-related information, at least in South-East Asia (but also in many parts of Europe), is that target languages are "π-languages" (Berment 2004), that is, languages which are poor (π) in NLP-related resources and applications such as dictionaries and MT systems.

Here again, there is one positive point: with modern technology putting emphasis on abstract, interlingual representations of texts, and using corpus-based and mutualization techniques, multilingual MT prototypes can be relatively quickly built at the laboratory level. If such "kernel systems" can be put to use without having to first go through a long and very costly development process needing important funding (which will never come), then they will grow as time goes, much in the way a full Linux has grown from a small kernel by the contributions of many.

That point is crucial, because the reason why there are few "language pairs" on sale today (perhaps less than 40, almost all having English as source or target) is simply that, whatever the MT approach used, the market for language pairs containing π-languages can not justify the development costs.

Constraints

There are three main constraints: speed, quality, and "honesty" about quality.

– *Information must be quicky available or it becomes useless.*
– *Quality is quite important, for some crucial parts.*

What is "quality" of translation in this context? From the reader point of view, it has three dimensions: understandability, fidelity, and fluency. The last one is slightly less important than the others in this context. Hence, a translation of an agricultural document, intended to be read and acted upon by farmers, will be deemed "very good" for the purpose if it is judged "quite good but not really fluent" by expert translators qualified to judge "professional quality".

Unfortunately, the dream of FAHQMT (Fully Automatic High Quality Machine Translation of texts) for general users has not come true and will not come true, for fundamental reasons, even if FAHQMT can be achieved on restricted kinds of texts (METEO, ALT/Flash[1]) or between very similar languages (e.g. Castillan, Galician, and Catalan).

If the final purpose of a MT system is high quality, a good measure is the time it takes a trained human to produce a final output of professional quality from the raw MT output, compared to what it takes starting from a human draft: **¡Error!**. With METEO, it is 1 minute per weather bulletin, 7 to 10 times less than what it takes to postedit a raw translation produced by a junior translator (before METEO existed). By that measure, the machine quality is 7 to 10 times better (Q_{rel} = 7, 10).

With systems tuned (at a high cost) to a specific kind of technical documents, quite broader than weather bulletins, such as agricultural information, MT can still beat humans ($Q_{rel} > 1$), as J. Slocum demonstrated with METAL in 1984 (Slocum 1984) on Siemens computer manuals.

But, as one tries to extend the coverage to all kinds of (sub)languages and situations, the finely tuned "expert systems" break down. That is why the bulk of useful automation for text translation has gone to translation aids (bilingual editors, online dictionaries, terminology extractors, and translation memories).

□ *Quality labels should be put on translated segments of information.*

What seems to be important, as anybody using web translators to access web pages in foreign languages knows, is to show to the reader which parts of a translated documents are deemed "good" and which are "bad". Humans translating or postediting part of a document are quite able to put marks saying how confident they are in their production.

Ultimately, the other parts should remain untouched MT outputs. Here, it is also often possible to program the MT system so that it outputs various marks of doubt or "self-evaluating" grades. In any case, the document management system could easily put <MT_output> tags around those parts. Of course, style sheets can then produce informative presentations (with different colors or layouts for the different qualities).

[1] A system derived from NTT ALT/JE and translating the Nikkei stock market flash reports from Japanese to English. It was introduced around June 2001 but the author could not check whether it was still running in 2004.

Given these characteristics of the translational situation, a pragmatic approach should be envisaged. First, mutualize manual translation and build lexical resources (Montaigne appproach). Second, build & integrate a UNL-based MT framework allowing incremental, interactive, mutualized quality improvement.

3 Mutualize Manual Translation and Build Lexical Resources

The basic idea of the Montaigne[2] approach, which we introduced in 1995 as a follow-up of the Eurolang Eureka project, but for which no funding could be raised at the time, is to let users share a common translation memory and other support tools such as a bilingual editor and online dictionaries, freely, through the network, in exchange for their agreement to share their data « products » with others. These data products are aligned sentences and dictionary entries produced by their translation activity. The pricing model is that of IE or Netscape : free clients and paying servers. Servers should be funded by institutions wanting their members to publish both in their native tongue and in English. That approach seems well suited to the dissemination of agricultural information in many languages at low cost, with high quality for crucial parts.

A concrete scenario would be to transform a source document into an XML "multilingual document", export the source sentences into a web-oriented translation tool (Montaigne), let bilingual targeted readers translate or postedit crucial parts, and produce an up to date HTML monolingual document each time a change is made on the text of its language. During the process, the shared multilingual lexical data base and translation memory will be enriched.

Transform source documents into "multilingual documents"

There are three steps; only the second requires limited human intervention.

1. *Transform a source document in XML, encoded in UTF-8.*
2. *Segment the text into sentences (or titles, captions…), and create one XML element per sentence.*

Although there are good algorithms for doing that, they are not perfect, so that some interaction is necessary at that point. If some errors remain, segmentation should also be modifiable in the translation editor.

We propose to use a special XML "namespace" for sentence elements, with top element <mld:p> (Annex, Fig. 9). This DTD takes over at paragraph level <mld:p> so that a paragraph is a possibly empty list of sentences (that covers other units of translation such as titles or captions).

Each sentence <S> is a "*polyphrase*", that is, a complex element containing:

[2] Mutualization Of Nomadic Translation Aids for Groups on the NEt
Mutualisation d'Outils Nomades de Traduction avec Aides Informatiques pour des Groupes sur le NEt.

- one or more *versions* of the *original sentence* (here, versions are used to keep track of corrections of errors of any kind),
- *translations* into other languages[3], each made of one or more *proposals* (e.g. by MT systems, or by humans).

Each proposal has one or more versions and corresponds to translations by different humans of MT systems. For humans, versions are as before. For MT systems, they refer to various parameter settings or dictionary combinations.

3. *Make each sentence element a multilingual structure.*

In each sentence <S>, the <org> element is filled, all others are empty.

Put the sentences into a web-oriented human translation tool

Many professional TA tools (such as Trados, TM2, Transit, Eurolang Optimizer) are integrated in a document processor (Word, Interleaf, Framemaker, or other), but that design is not applicable if we want several users to edit the document at the same time from their PC. Some have also argued that this design is too sophisticated (hence costly) and also somewhat counter-productive. They prefer a more "bare-bone" tool (like Xerox XMS bilingual editor), with a screen layout from which most formatting, images, etc., have been removed, so that they can concentrate on translation alone.

- Typical screen layout of a TA screen (Fig. 1)

It consists of a 2-column table with one line for each sentence and a frame for suggestions coming from the translation memory (TM) and MT system(s).

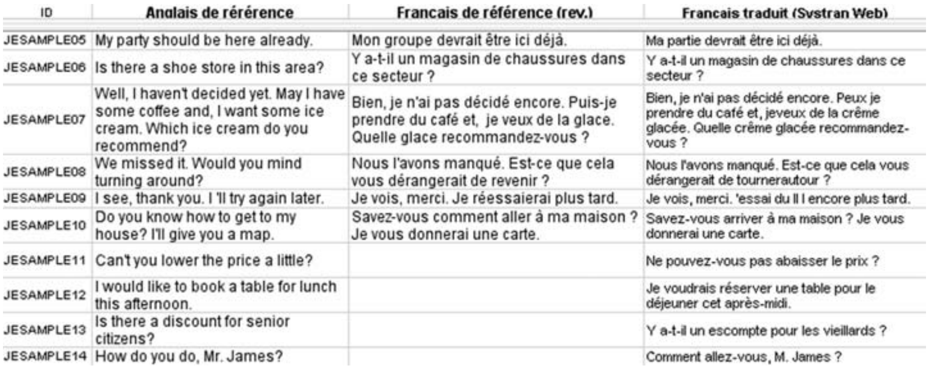
…	…	
source segment N-2	translated segment (done)	
source segment N-1	translated segment (done)	suggestion(s) from the TM
source segment N	translated segment (currently being created)	and/or from MT systems
source segment N+1	— empty —	dictionary suggestions
source segment N+2	— empty —	

Fig. 1: typical layout of a bilingual editor in a TWB.

At the beginning, there may be no TM, but the very process of translation creates at least one, that of the document, which can then be integrated in a larger TM, resulting from the translation of many document (parts).

Suggestions for translations of sentences and words or terms appear to the right, when one clicks a translation segment. Using usual editing functions and specific shortcuts, the user translates or postedits. When s/he clicks in the next segment or quit, the server updates the document with the proposal. Before that TA tool is available, one can use a database or a speadsheet. Some translation aids can be implemented as macros, but it is far less efficient, and not sharable.

[3] If one is the source language, it is rather a paraphrase, but we use one term only.



ID	Anglais de rérérence	Francais de référence (rev.)	Francais traduit (Systran Web)
JESAMPLE05	My party should be here already.	Mon groupe devrait être ici déjà.	Ma partie devrait être ici déjà.
JESAMPLE06	Is there a shoe store in this area?	Y a-t-il un magasin de chaussures dans ce secteur ?	Y a-t-il un magasin de chaussures dans ce secteur ?
JESAMPLE07	Well, I haven't decided yet. May I have some coffee and, I want some ice cream. Which ice cream do you recommend?	Bien, je n'ai pas décidé encore. Puis-je prendre du café et, je veux de la glace. Quelle glace recommandez-vous ?	Bien, je n'ai pas décidé encore. Peux je prendre du café et, jeveux de la crême glacée. Quelle crême glacée recommandez-vous ?
JESAMPLE08	We missed it. Would you mind turning around?	Nous l'avons manqué. Est-ce que cela vous dérangerait de revenir ?	Nous l'avons manqué. Est-ce que cela vous dérangerait de tournerautour ?
JESAMPLE09	I see, thank you. I 'll try again later.	Je vois, merci. Je réessaierai plus tard.	Je vois, merci. 'essai du ll l encore plus tard.
JESAMPLE10	Do you know how to get to my house? I'll give you a map.	Savez-vous comment aller à ma maison ? Je vous donnerai une carte.	Savez-vous arriver à ma maison ? Je vous donnerai une carte.
JESAMPLE11	Can't you lower the price a little?		Ne pouvez-vous pas abaisser le prix ?
JESAMPLE12	I would like to book a table for lunch this afternoon.		Je voudrais réserver une table pour le déjeuner cet après-midi.
JESAMPLE13	Is there a discount for senior citizens?		Y a-t-il un escompte pour les vieillards ?
JESAMPLE14	How do you do, Mr. James?		Comment allez-vous, M. James ?

Fig. 2: Example of work in progress
(under Excel, without specific translation aids)

– *Link with the multilingual document*

As mentioned before, the translator should be able to change the segmentation from the TA tools, and to correct errors (spelling, grammar, vocabulary) in the source document. Hence, objects have to be uniquely identified (id attributes in Fig. 9). It is even possible to present the sentences in an order different from that of the text, e.g. to group similar ones to speed up translation.

Using such a linking scheme is useful to solve a well-known problem: translated documents are not always aligned sentence by sentence. Sometime, 2 sentences are translated by 1 sentence, or 1 by 2, or 2 by 3… Then, we may slightly extend the notion of polyphrase and create a "compound" polyphrase with a new id for a segment of 2 sentences, without destroying the individual sentences. It is also common that 2 sentences in Japanese are equivalent to 2 sentences in French or English, but not in the same order[4]. Linking solves this problem. However, we don't yet know how to link sentences with their contexts.

Let bilingual human readers translate the most important parts

In practice the scenario is that:

– a user uses a brower to read an html page produced from the document, then sees a passage in need of translation or revision,
– s/he selects that passage and chooses a "Translate/Revise" menu item,
– thanks to code (<span> tags) included in the html page, the translation editor is called on the sentences intersecting with the selection,
– the contributor does some translation/revision, then exits and returns to the normal reading mode.
– Some points are important here:

[4] For example, in Japanese, *"X. That is why Y."*, and in English *"Y. That is because X.",* or *"Y because X."*.

- the current formatted (html) document can be shown, in one browser window per language, and updated as translation or revision progresses;
- the translation editor runs on the server as a web service, so that several persons can work concurrently on the same sentence of the document;
- translations of the same sentence by different users are simply added as different proposals, in a "monotonic" way, so that there is no conflict.

Build up bilingual lexical knowledge

- All TA tools include a dynamic dictionary: when the translator finds a new equivalent, s/he puts it there, and it is immediately active. Of course, dictionary items should be marked with their authors, in particular, for crediting contributors as a way to motivate them.
- The set of polyphrases corresponding to the sentences of a document constitutes a *"multilingual polyphrase memory",* or *MPM,* relative to that document. The "good graded" parts of all MPMs should be consolidated in a main, shared MPM.

4 Build & integrate a UNL-based MT framework allowing incremental, interactive, mutualized quality improvement

The second part of our design relies on building UNL-based resources for the languages at hand, and integrating them in the same Montaigne web site.

The UNL language of "anglo-semantic hypergraphs"

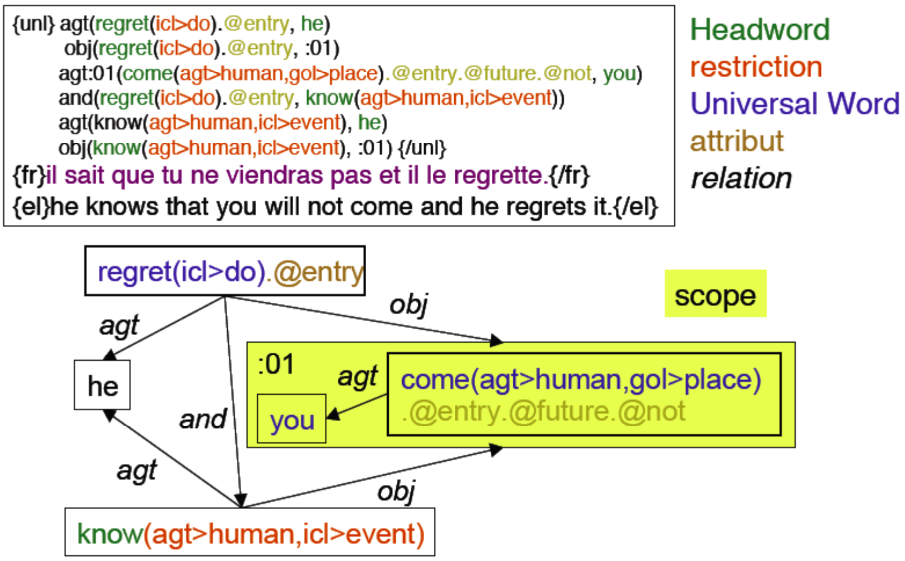
Definition and example

UNL is a project, an html-based format for multilingual documents, and, essentially, a computer language to represent the meaning of natural language sentences (in the same sense as above) through semantic hypergraphs. Its labels are built from English lexemes, and, in order to have a clear reference, a UNL graph is to be understood as an abstract structure of an English sentence, the original one or an English equivalent if the original is in another language.

As an example, Fig. 3[5] shows a graph corresponding to the sentence *"he knows you won't come and regrets it"* (or any semantically equivalent rendering, in English, French, etc.), and its linear description in the UNL syntax.

Nodes contain lexical units and attributes, arcs bear semantic relations. Connex subgraphs may be defined as "scopes" (here there is one, corresponding to *"you will not come"*, so that a UNL graph is in general a hypergraph.

[5] It has colors: green for headwords (come), red for restrictions (agt>human,gol>place), brown for attributes (.@entry.@future.@not).



```
{unl} agt(regret(icl>do).@entry, he)
       obj(regret(icl>do).@entry, :01)
      agt:01(come(agt>human,gol>place).@entry.@future.@not, you)
      and(regret(icl>do).@entry, know(agt>human,icl>event))
      agt(know(agt>human,icl>event), he)
      obj(know(agt>human,icl>event), :01) {/unl}
{fr}il sait que tu ne viendras pas et il le regrette.{/fr}
{el}he knows that you will not come and he regrets it.{/el}
```

Fig. 3: Example of a UNL graph

A lexical unit, called *Universal Word (UW),* or "Unit of Virtual Vocabulary", represents a word meaning, something less ambitious than a concept[6]. Their denotations are built to be intuitively understood by developers knowing English, that is, by all developers in NLP. A UW is an English term or pseudo-term[7] possibly completed by semantic restrictions. A UW such as *"process"* represents all word meanings of that lemma, seen as citation form (verb or noun here). The UW *"process(icl>do, agt>person)"* covers the verbal meanings of processing, working on, etc.

The attributes are the (semantic) number, genre, time, aspect, modality, etc., and the 40 or so semantic relations are traditional "deep cases" such as agent, (deep) object, location, goal, time, etc.

One way of looking at a UNL graph corresponding to an utterance U-L in language L is to say that *it represents the abstract structure of an equivalent English utterance U-E as "seen from L",* meaning that semantic attributes not necessarily expressed in L may be absent (e.g., aspect coming from French, determination or number coming from Japanese, etc.).

UNL graphs are understandable and manipulatable by non-specialists.

See (Blanc 2001, Boitet 2002) or the UNL web site (www.undl.org) for more information on UNL graphs. What is important for our design is that this representation strikes a very good balance between abstractness and practicality. Although abstract, the formalism of UNL graphs is not equivalent to first-order logic, and may contain

[6] Indeed, two different UWs may correspond to the same concept.

[7] Number in various notations, part number, punctuation, formatting tag, file name or path, hyperlink…

indeterminacies,[8] which is very useful in practice. Its nature leads also to direct manipulation through graphical interfaces.

Experience gathered by the UNL project

To date, the UNL project has initiated 16 language groups[9], each working on its native language.[10] Practical work with UNL has involved building UNL-L dictionaries (typically, more than 50000 lemmas "connected" with UWs), manual encoding in UNL to learn and test the specifications, deconverters (from UNL to a language, some quite large), and enconverters (mostly prototypes), and performing experiments (deconverting from UNL graphs prepared by other groups, building UNL annotated corpora).

Some critics have claimed that the UNL approach to MT cannot work because the "abstract pivot" technique cannot work, and in any case cannot support large coverage applications. That view is completely false, because:

- the "pivot" technique has been not only experimented but deployed successfully (ATLAS-II by Fujitsu, PIVOT by NEC, KANT / CATALYST by CMU at Caterpillar, IBM speech translation MASTOR).
- in particular, ATLAS-II uses a pivot from which UNL has evolved. H. Uchida, main designer of UNL, was the main designer of ATLAS-II.
- ATLAS-II has been recognized as the best EJ/JE MT system in Japan for over 15 years and has a very large coverage (586,000 words in English and Japanese in 2001, about 1,000,000 in 2003 as reported during ACL).
- while it is true that interlingual representations can not in principle be used (alone) to achieve the highest quality achievable by transfer systems, they can give quite high quality as demonstrated by ATLAS-II.

Enconversion

To stress that the passage from a written sentence to a UNL graph is not a traditional analysis, the UNL project refers to it as *enconversion*. The converse process is called *deconversion*. An analysis process produces a representation with lexical symbols attached to the source language, while enconversion is more a translation, because UNL has an autonomous set of lexical symbols.

At the beginning of the enterprise, one should enconvert a "trickle" of documents manually, to prepare data for building an automatic or semi-automatic enconverter, and for starting immediately work on the deconverters. Note that, even if it takes 5 hours per page (about 15 minutes per sentences) to enconvert manually, the total

[8] If a precise relation cannot be determined, one simply uses "mod", if a word sense cannot be totally disambiguated, one uses a less precise UW, etc.

[9] Active in 2004: Arabic, Armenian, French, Hindi, Indonesian, Italian, Japanese, Portuguese, Russian, Spanish. Inactive or stopped: Chinese, German, Korean, Latvian, Mongolian, Thai.

[10] English is a special case, as it is handled by the UNL center. But other groups, such as IPPI in Moscow, use their preexisting L-En systems to build L-UNL systems, and can handle English as a "byproduct".

human time to produce a page in N target languages is less than the time needed for usual human translation if $N \geq 6$.[11]

Nevertheless, enconversion should be mostly automatic if information has to be delivered very quickly. Hence, the idea is to produce the best possible analysis within a given time, for example, 5 minutes per page. This can be done with 2 different techniques: heuristic analysis, and multiple analysis followed by some interactive disambiguation (ID).

Heuristic approach: one analysis is produced.

Techniques based on direct programming (Systran and many others), on ATNs[12], on Prolog (LMT of IBM & Linguatec), or on tree transducers[13], usually fall in that category. Direct programming and tree transducers permit the production of a structure containing the representation of *some* ambiguities. In that case, it is possible to produce translations showing alternatives, which is quite useful.

ID approach: multiple analysis, then interactive disambiguation.

Many other parsers are based on extended context-free formalisms, including ATNs again, attributed CFGs, Prolog DCGs, and all "xyzG" formalisms such as LFG, GPSG, TAG, HPSG, and their variants. The parser produces a *set* of either "*concrete*" trees, or of "*abstract*" trees. These trees may be scored or not. Anyway, even after keeping only those with the best scores, the size of the candidate set may be quite large and still exponential in the length of the input (3000 or more for a 20-word sentence, using a compact formalisms, millions if no disjunction is allowed in a given solution).

Interactive disambiguation can be done at that point to reduce the size of the candidate set. When a human answers a question, it is typically divided by 2, 3 or 4 according to the number of possible answers. Hence, the maximal number of questions to reduce the set to 1 candidate is linear in the size of the sentence. In our LIDIA-1 experiments, we arrived at 1 question for 2 words, hence, about 120 questions for 1 page, answerable in 10 minutes or less.

If the allowed time is too short, or there is nobody to perform the ID, automatic disambiguation is used on the remaining candidates. As decisions impossible to make reliably by a program have been made by ID, the result is far better than without ID. In other words, even a very partial ID, answering 10% of the questions, can dramatically improve the quality of the output[14].

[11] Time permitting, a table with detailed numbers will be shown during the oral presentation.

[12] Spanam/Engspan of PAHO, AS/Transac of Toshiba, Reverso of Prompt-Softissimo.

[13] ROBRA in Ariane-G5, GRADE in MU-Majestic, HICATS of Hitachi, GWS at ISS/CRDL in Singapore.

[14] We should also take into account the fact that, during the ID process, the human may tell the system to remember some decisions and reapply them if a similar case arises.

In both cases, direct edition of the UNL graph is possible.

In the most frequent case, analysis does not produce a UNL graph, but a tree containing lexemes of the source language, not UWs. Enconversion continues by a classical "transfer" into UNL. Lexical transfer replaces lexemes of the source language by UWs, and structural transfer produces a special kind of deep dependency tree, called "UNL tree", and folds it into a UNL graph.

The UNL-Spain group has long proposed and produced a UNL editor which presents a UNL graph in a "localized" way (e.g., using Spanish words). We feel that even children would like to play with a full-fledged editor of that kind, provided it is linked with a deconverter showing almost in real time renderings of the graph in one or more languages. Direct edition of the UNL graphs can be seen as complementing interactive disambiguation to improve enconversion.

Deconversion

There is a lot less to say about deconversion.

- *In the usual approach, it is fully automatic.*
- *As shown by (Blanc 2001), one can interactively improve lexical selection during deconversion.*

However, in our translational situation, we can not expect readers to help the deconversion process. Interactive processes are acceptable only if humans decide when they will help the machine, not if they are "slaves of the machine".

Coedition

The concept

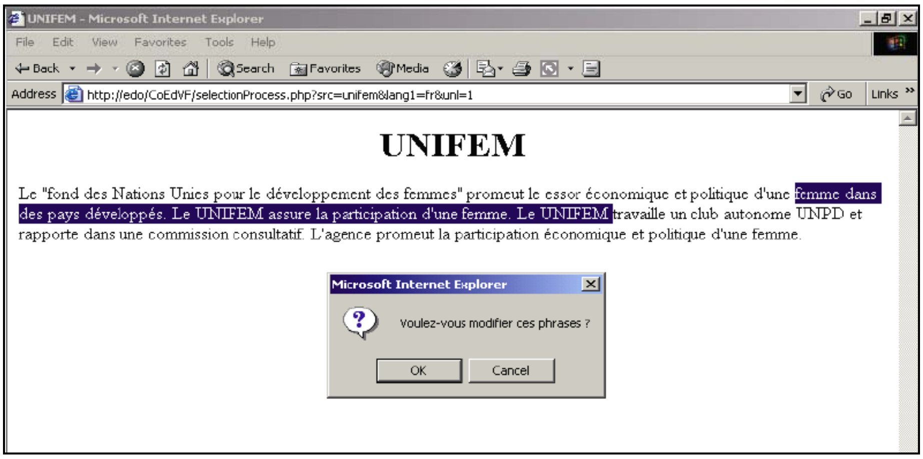
The main idea is to *share revision across languages*. If a reader sees a mistake in a sentence and corrects it directly, sharing is impossible, even if there is an associated UNL graph, as a program cannot infer modifications on the graph from modifications on the text without calling a UNL enconverter. A technique proposed by (Boitet & Tsai 2002) and prototyped by (Tsai 2004) is that:

- revision is not done by modifying directly the text, but by using menus,
- the menu items have a "language side" and a hidden "UNL side",
- when a menu item is chosen, only the graph is transformed, and the action to be done on the text is stored and shown next to its focus.
- at any time, the graph may be sent to the deconverter, to check the result.

If it is satisfactory, errors were due to the graph and not to the deconverter, so that the graph may be sent to deconverters in other languages. Deconversions in languages known by the user may be displayed, to make improvements visible and encourage his/her contribution.

Example

First, the reader accesses a web page, as below (example from a text on Forum Barcelona 2004), and sees a passage with mistakes in 3 consecutive sentences.


UNIFEM - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
Microsoft Internet Explorer
OK
Cancel

http://edo/CoEdVF/selectionProcess.php?src=unifem&lang1=fr&unl=1

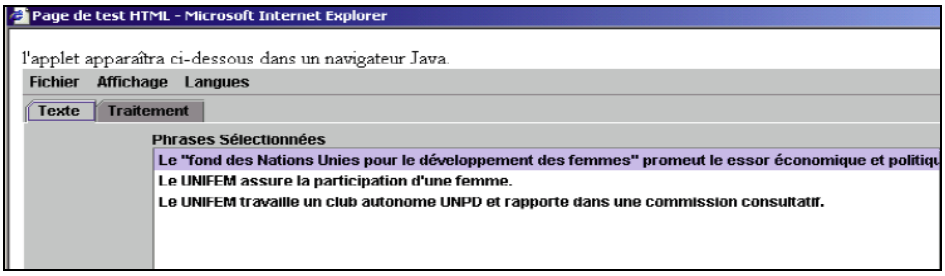
UNIFEM

Le "fond des Nations Unies pour le développement des femmes" promeut le essor économique et politique d'une femme dans des pays développés. Le UNIFEM assure la participation d'une femme. Le UNIFEM travaille un club autonome UNPD et rapporte dans une commission consultatif. L'agence promeut la participation économique et politique d'une femme.

Voulez-vous modifier ces phrases ?

Fig. 4: Reading information "roughly" translated" in a web browser

S/he selects a portion intersecting with these sentences, and chooses the "coedition" menu item. Thanks to <span> tags in the html page, the 3 complete sentences are identified, and a java application running on the server opens, showing them. The user selects each in turn to "coedit" it.


Page de test HTML - Microsoft Internet Explorer
Fichier Affichage Langues
Texte Traitement

l'applet apparaîtra ci-dessous dans un navigateur Java.

Phrases Sélectionnées

Le "fond des Nations Unies pour le développement des femmes" promeut le essor économique et politiqu

Le UNIFEM assure la participation d'une femme.

Le UNIFEM travaille un club autonome UNPD et rapporte dans une commission consultatif.

Fig. 5: Sentences determined by the selection appear in a java window

Now, the system must establish a correspondence between the sentence and its UNL graph. That is possible even if no analyzer (and no deconverter) exists for this language, and even if the translation has been produced manually!

Our method relies on low-level resources, the first which are built for any π-language: *a word-segmenter* (and lemmatizer in case of an inflected language), *and a bilingual dictionary* between that language (L) and English. If and when a UNL-L dictionary if available, it can be used also. The good news are that such resources become more and more available, in free mode, because of the contributions of volunteer developers. For example, V. Berment has developed a web site for Lao[15], and proposed ways to computerize groups of languages[16].

It is interesting in a paper like this to explain how a text-UNL correspondence can be established without any analyzer or generator, but it should be clear that, when it comes to using a coedition system, this remains absolutely hidden from the user. *As a matter of fact, the "normal" user should never even see the graph!* Here is a brief account of how it works. For more details, see (Tsai 2004). The UNL graph is first transformed (by program) into a *UNL tree.* Lexemes of L are attached to each node having a UW *u* by using the headword of *u* as a key and its restrictions as a filter (e.g., *icl>do* indicates a verb or an action noun).

Then, a *lexicomorphosyntactic lattice (LMSL)* is produced using a segmenter-lemmatizer. English lexemes are attached to it using again the En-L dictionary. A "best" correspondence between the LMSL and the tree is computed in two steps. Lexical links are created between two nodes (LMSL, tree) if their "lexical intersections" (in English and/or L) are not empty. Then, only the lexemes in these intersections are kept. Note that a link may in fact link more than one node on each side (e.g., two nodes in the LMSL for a verb and its particle in German or English, or one node in the LMSL for a simple word in L rendered by a compound word in English and hence by a scope in the UNL graph[17]).

The second step is to compute non lexical links[18]. Such links are established if they are "near" to lexical links: we keep links such that, if the linear precedence[19] in the tree is adjusted (the tree "rotates"), there as few crossings of links as possible. When this is done, a "trajectory" (a segmentation of the sentence in words, and a LMS interpretation for each word) has been determined in the LMSL. There is a (possibly partial) correspondence between it and the UNL tree, and a total correspondence between the UNL tree and the UNL graph.

If the user clicks on the text, the word (in the chosen trajectory) surrounding the cursor is selected. If there are links between the corresponding node(s) in the LMSL and the tree, they can be used to go from the text to the graph, and a menu is prepared. If not, no coedition action is possible from that word.

A menu item contains two parts: an annotation, for the interface, and a hidden part, for the system, expressing what actions to do on the graph and on which nodes. Here is an example on a French sentence deconverted from a graph propared from Chinese for *"UNIFEM ensures the participation of women".* In French, we got *"d'une femme"* (singular, not definite) because the input graph did not contain the appropriate attributes on the node corresponding to *"women".* After the user has chosen *"plural",* the

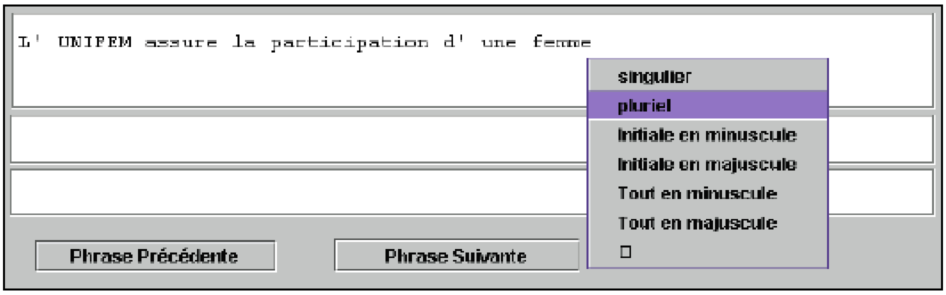
[15] www.laosoftware.com.

[16] (Berment 2004) actually shows how to build generic NLP components and how this leads to dramatic cost reductions, e.g. by 100 for deriving BanglaWord (for Bengali) from LaoWord (a tool for handling Lao in Word).

[17] E;g., *lombarda* in Italian and *red cabbage* in English, *profiter de* in French and *enjoy the benefit of* in English.

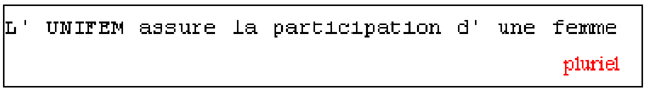
[18] between a preposition and a node containing a semantic relation, an article and a node containing the corresponding determination feature, an article and a node containing the corresponding number, etc.

[19] Linear precedence is the "horizontal ordering" determined by totally ordering all daughter nodes of each node.



L' UNIFEM assure la participation d' une femme

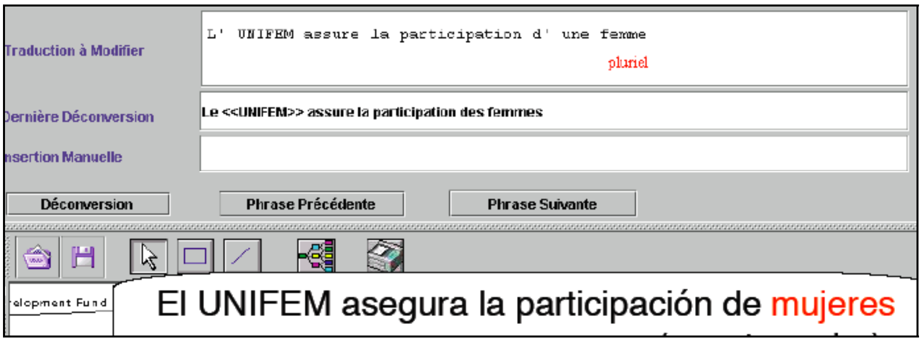
Fig. 6: Possible corrections are proposed



L' UNIFEM assure la participation d' une femme

Fig. 7: What the user has asked is shown as an annotion

Finally, the user calls the deconverter(s).



L' UNIFEM assure la participation d' une femme

Le <<UNIFEM>> assure la participation des femmes

El UNIFEM asegura la participación de mujeres

Fig. 8: Deconversion in two languages after coedition in one language

.*@plur* attribute is put on the corresponding node, and the annotation is left next to the word selected.

In the example, we obtain *"des femmes",* without having modified the article, simply because the whole sentence is deconverted again from the new graph, which generates agreement in gender. Here, the user has also asked to see the Spanish output, and the same change (of singular to plural) can be observed.

Of course, there are things which are impossible to do by coedition, and things which are not well handled by the deconverter at hand. That is why the user should always be free to modify the result of deconversion. Here, the French deconverter did not (yet) correctly generate "UNIFEM"[20], so that the user will copy the result into the "free translation" text area and modify it directly.

[20] *Le <<UNIFEM>>* instead of *L'UNIFEM.*

Conclusion

Translation of specialized information into many languages is necessary, notably in agriculture, health, and other domains, because it is often crucial for final users, who don't master the source language. Here, quality should be very high, at least for the most important parts. At the same time, resources are scarce, especially to produce high quality translations. In many cases, also, it is urgent to use the information, and only automated translation can offer a solution in the long run. However, in this and similar translational situations, it is acceptable that the quality of translations varies from poor for inessential parts to very good for crucial parts. Translated sentences or paragraphs should be accompanied with a measured and certified "quality level". We have proposed an organization where this can be obtained through a combination of "mutualized" human work and automatic NLP techniques, using the UNL language of "anglosemantic" graphs as a "pivot". UNL graphs (produced automatically, manually, or semi-automatically) can be directly improved by college level persons using graphical editors and presentations localized for each language. Many very important improvements can also be performed on UNL graphs by monolingual readers, using a "coedition" environment to annotate sentences and indirectly modify their UNL graphs.

Building the necessary multilingual lexical data base should and can be done in a mutualized way, for example by contributing to the MLDB Papillon project[21], and getting from it lexical files in appropriate formats (MT lexicons, usage dictionaries for human readers, terminological lists for specialized translators). All these functions could be integrated in a "Montaigne" environment allowing users to access information through a browser and to switch easily to translating or postediting and back.

Acknowledgments I would like to thank Prof. Asanee Krawtrakul for having invited me to present a first version of this paper at AFITA-04, and waited with Thai patience and kindness while I was struggling with it and finally developing its scope and detail far beyond what I anticipated. Many thanks also to Tsai Wang-Ju, who developed the coedition approach in his PhD, and from whom I borrowed some figures and screen shots.

References

Al Assimi A.-B. & Boitet C. (2001) *Management of Non-Centralized Evolution of Parallel Multilingual Documents.* Proc. of Internationalization Track, 10th International World Wide Web Conference, Hong Kong, May 1-5, 2001, 7 p.

Berment V. (2004) *Méthodes pour informatiser des langues et des groupes de langues « peu dotées »*. Thèse, UJF (thèse préparée au GETA, CLIPS), 18/5/04, 277 p.

Blanc E. (2001) *From graph to tree : Processing UNL graph using an existing MT system.* Proc. of First UNL Open Conference - Building Global Knowledge with UNL, Suzhou, China, 18-20 Nov. 2001, UNDL (Geneva), 6 p.

[21] www.papillon-dictionary.org.

Boguslavsky I., Frid N., Iomdin L., Kreidlin L., Sagalova I. & Sizov V. (2000) *Creating a Universal Networking Language Module within an Advanced NLP System*. Proc. of COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL & Morgan Kaufmann, 1/2, pp. 83-89.

Boitet C. & Zaharin Y. (1988) *Representation trees and string-tree correspondences*. Proc. of COLING-88, Budapest, 22–27 Aug. 1988, ACL, pp. 59—64.

Boitet C. (1999) *A research perspective on how to democratize machine translation and translation aids aiming at high quality final output*. Proc. of MT Summit VII, Singapore, 13-17 September 1999, Asia Pacific Ass. for MT, pp. 125—133.

Boitet C. (2001) *Four technical and organizational keys for handling more languages and improving quality (on demand) in MT*. Proc. of MTS2001 Workshop on "MT2010 — Towards a Road Map for MT", Santiago de Compostela, 18/9/01, IAMT, 8 p.

Boitet C. (2002) *Advantages of the UNL language and format for web-oriented crosslingual applications*. Proc. of Seminar on linguistic meaning representation and their applications over the World Wide Web, Penang, 20-22/8/2002, USM, 4 p.

Boitet C. (2002) *A roadmap for MT : four « keys » to handle more languages, for all kinds of tasks, while making it possible to improve quality (on demand)*. Proc. of International Conference on Universal Knowledge and Language (ICUKL2002), Goa, 25-29/11/02, 12 p.

Boitet C. (2002) *A rationale for using UNL as an interlingua and more in various domains*. Proc. of LREC-02 First International Workshop on UNL, other Interlinguas, and their Applications, Las Palmas, 26-31/5/2002, ELRA/ELDA, pp. 23—26.

Boitet C. & Tsai W.-J. (2002) *Coedition to share text revision across languages*. Proc. of COLING-02 WS on MT, Taipeh, 1/9/2002, 8 p.

Boitet C. (2003) *Automated Translation*. Revue française de linguistique appliquée, Vol., N° VIII-2, pp. 99-121.

Chandioux J. (1988) *10 ans de METEO (MD)*. In *Traduction Assistée par Ordinateur. Actes du séminaire international sur la TAO et dossiers complémentaires*, edited by Abbou A., Paris, mars 1988, Observatoire Francophone des Industries de la Langue (OFIL), pp. 169—173.

Coch J. & Chevreau K. (2001) *Interactive Multilingual Generation*. Proc. of CICLing-2001 (Computational Linguistics and Intelligent Text Processing), Mexico, February 2001, Springer, pp. 239-250.

Sérasset G. & Boitet C. (1999) *UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction*. Proc. of MT Summit VII, Singapore, 13-17 September 1999, Asia Pacific Ass. for MT, pp. 220—228.

Sérasset G. & Boitet C. (2000) *On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter*. Proc. of COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL & Morgan Kaufmann, 2/2, pp. 768—774.

Tsai W.-J. (2004) *La coédition langue – UNL pour partager la révision entre langues d'un document multilingue*. Thèse, UJF (thèse préparée au GETA, CLIPS), 9/7/04, 311 p.

Uchida H. (1989) *ATLAS*. Proc. of MTS-II (MT Summit), Munich, 16-18 août 1989, pp. 152-157.

Vasconcellos M. & León M. (1988) *SPANAM and ENGSPAN : Machine Translation at the Pan American Health Organization*. In *Machine Translation systems*, edited by Slocum J., Cambridge Univ. Press, pp. 187—236.

Vauquois B. & Chappuy S. (1985) *Static grammars: a formalism for the description of linguistic models*. Proc. of TMI-85 (Conf. on theoretical and metholodogical issues in the Machine Translation of natural languages), Aug. 1985, pp. 298-322.

Zaharin Y. (1986) Strategies and heuristics in the analysis of a natural language in Machine Translation. Proc. of COLING-86, Bonn, Aug. 1986, pp. 136—139.

Annex

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- PMLD.dtd (paragraph of multilingual document).
This DTD takes over at paragraph level <pmld:p> so that a para-
graph is a possibly empty list of sentences (that terms covers
other units of translation such as titles or captions).
Each sentence <S> is what we call a "polyphrase", that is, a
complex element containing
- one or more versions of the original sentence (versions are
there to keep track of modifications)
- translations into other languages (if one is the source lan-
guage, it is rather a paraphrase, but we use one term only),
. each having one or more proposals (e.g. by MT systems, or by
humans),
. and each proposal having in turn one or more versions.
 $Author: Christian Boitet Christian.Boitet@imag.fr
 $Date: 2004/07/22 9:30 TU $
 -->
<!ELEMENT p                (S*)>
 <!-- sentence: translation unit, also title, caption -->
<!ELEMENT S                (org,transl*)>
<!ATTLIST S                id CDATA #REQUIRED>
 <!-- original sentence, with possible versions -->
<!ELEMENT org              (version+)>
<!ATTLIST org              xml:lang CDATA #REQUIRED>
<!ATTLIST org              auth CDATA>
<!ATTLIST org              id CDATA #REQUIRED>
 <!-- version: v is a string of form n.m.p, such as 0.1.1 -->
<!ELEMENT version          (#PCDATA)>
<!ATTLIST version          v CDATA #REQUIRED>
<!ATTLIST version          auth CDATA>
<!ATTLIST version          date-creat #IMPLIED>
<!ATTLIST version          date-modif #IMPLIED>
<!ATTLIST version          id CDATA #REQUIRED>
 <!-- translation: never 2 <transl> for same <lang> -->
<!ELEMENT transl           (proposal+)>
<!ATTLIST transl           xml:lang CDATA #REQUIRED>Algo
<!ATTLIST transl           auth CDATA>
<!ATTLIST transl           date-creat #IMPLIED>
<!ATTLIST transl           date-modif #IMPLIED>
<!ATTLIST transl           id CDATA #REQUIRED>
 <!-- proposal: all in same <transl>ation are in same <lang>uage
-->
<!ELEMENT proposal         (version+)>
<!ATTLIST proposal         id CDATA #REQUIRED>
```

Fig. 9: PMLD.dtd, for a paragraph-to-paragraph-aligned multilingual document.

Towards a Systematic Process in the use of UNL to Support Multilingual Services

Jesús Cardeñosa, Carolina Gallardo, Edmundo Tovar

Departamento de Inteligencia Artificial- Facultad de Informática
Universidad Politécnica de Madrid
Campo de Montegancedo, s/n
28660 Madrid- Spain
{carde, carolina, etovar}@opera.dia.fi.upm.es

Abstract. The UNL Programme of the United Nations University (UNU) was launched in 1996 aiming at the elimination of linguistic barriers in Internet. Now, eight years later, UNL is not ready to support real applications due to several circumstances. This eight-year period can be divided in two: a first four-year period devoted to the formal definition of UNL as a formal language (under the sponsorship of the Institute of Advanced Studies (IAS) of the UNU) and the remaining four years devoted to the technical experimentation of UNL. A new period is starting right now, which could be the period of maturity at all levels, especially at technical and business levels. In this paper, the authors summarize the more significant experiences until now, their conclusions and the set of procedures to produce marketable multilingual services. This kind of work will be the work of the UNL consortium during the next two years before launching UNL to the market.

1 Introduction

The natural evolution of UNL as a project and as a Programme is the support of useful applications for a multilingual society. Apart from other uses of UNL, like cross-lingual information retrieval or support for ontologies, the more understable use and possibly the easiest application, is the support of multilingual services, that is, to represent contents written in any language and to generate any other language [1].

UNL is not conceived to become a (fully automatic) machine translation system (MT hereafter). Up to date, MT systems based on the transfer architecture have achieved reasonable results, always involving pairs of languages. These systems are somehow handicapped by their *language coverage*. In other words, a transfer based system involving N languages requires the development of N × (N–1) systems, which ends up with the consequent combinatorial explosion of the number of systems to be developed as the number of languages grows.

On the other hand, interlingua-based MT systems show, in principle, a highly attractive advantage over transfer systems: interlingua-based systems do not grow exponentially as the number of language increases since for a system to support N languages, only 2 × N systems have to be developed. The ATLAS system [2] and the PIVOT system [3] in open domains, and Mikrokosmos [4] and Kant [5] in restricted

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 413–429.

domains are the most representative systems within the interlingua-based MT paradigm.

However, not everything is so easy and straight ahead in interlingua-based systems. In fact, currently there are not interlingua-based systems in open domains, nor Interlingua systems that have been able to penetrate in the market. One of the possible reasons to explain this fact is the practical design of the Interlingua itself and the pivotal role it plays in a MT system. The reason for this minor development of interlingua-based MT systems (specially in open domains) could be the difficulty in designing a formal language that simultaneously is far enough from the surface forms of natural languages (so that almost all languages can fit in the interlingua representation) and that is expressive and rich enough to convey the subtleties in meaning expressed in natural languages [6]. Thus, the proper design of the Interlingua will affect the overall behavior of the system in the analysis and generation processes.

UNL, in terms of Interlingua design, had to find the balance between a representation where linguistic meaning could be naturally expressed and a representation not devoted or inspired by a given natural language and, of course, not restricted to particular domains. After years of debates and discussions, it seems that this difficult balance was found. However, massive encoding experiences in the UNL context have given away a worrying aspect of UNL: the lack of common understanding of the specifications in almost all the components of the language (universal words, attributes and relations), possibly due to the incomplete definition of the language and codification procedures in the current version of the UNL specifications [7].

This incompleteness and imprecision in the definition of the specifications of the language provokes a wide variety of UNL code according to the encoder's understanding of the UNL language and even according to the source language of the contents to be encoded. Such a variety negatively affects the results of language generators (independently of the target languages and used systems). Not only should be pursued the interdependence among participants in the process of defining a uniform way to encode contents into UNL but also uniformity in the processes and methodology when working with UNL. That is, independently from low-level linguistic and codification considerations, the clear definition of both the working processes and the complete definition of the UNL as a language is indispensable if the development of services based on UNL is targeted at.

Some members of the UNL consortium have thoroughly considered these two aspects since time ago. The first experience with this purpose was in 2001 when, as a result of some conversations with the organizing committee of the international event Forum Barcelona 2004, it could be seen that the UNL lacked the necessary infrastructure to be able to provide multilingual services. During the encoding tasks of the Barcelona experience, it could be proved how the UNL specifications did not provide a clear answer of how to codify real texts (not just toy examples). The same applied to the definition of the Universal Words. Besides, there was neither a formal definition of the Knowledge Base nor how it has to be used, with the final result that even having the capacity to build a knowledge base for UNL, there was no way to do it. There were also no tools for UNL massive codification (the manual process is tedious, and with high risk of error), and moreover there was not a definition of the processes to be carried out in the production of UNL code and multilingual generation. From the point of view of the standards of technological development, in particular Software

Engineering, it could be confirmed that UNL was far from being considered mature when facing massive production [8].

From that moment on, several partners of the UNL Consortium agreed on beginning to define such processes and at least some common guidelines for codification that will unify the procedures in order to assure a reliable production later on. The outcomes of such experience were encouraging. Initial guidelines for the codification were produced [9] and the first set of processes could be exposed [10].

Subsequently and up to now, there have been two more experiences trying to emulate the problems that may arise in massive codification scenarios. These are the so-called "HEREIN experience" [11] and UNESCO [12]. Both experiences proved that commercial production of UNL goes through the creation of huge amounts of contents in UNL and the concise definition of the involved processes, roles, techniques, tools and standards. Without all that, UNL would never surpass the theoretical limits of its possibilities.

This article presents a general methodology for multilingual generation in the UNL context. The article is organized as follows: section 2 summarizes a comparative analysis of the experiences carried out so far and their most representative drawn conclusions. In section 3, a working methodology will be presented. This methodology has been defined after the experiences of Barcelona, Herein and UNESCO and it is the first step in the staging of UNL as a support for multilingual services. Section 4 presents some advances in the definition of metrics, necessary to estimate costs and productivity. Without methodologies and processes it is impossible to evaluate costs in the development of applications based on UNL and, consequently, to evaluate possibilities of UNL in the market.

2 Experiences. A Comparative Analysis

The need to define and determine the involved procedures in the process of multilingual generation has lead to the UNL Consortium to undertake several experiences that will explore the processes of the complete cycle of production –that is, from contents written in a given language, to their enconversion and final deconversion into other languages. For the time being, the most general tasks in this process were:

- Lexicographic tasks: where UWs had to be defined and dictionaries updated with the new UWS.
- Codification task: once the UWs have been defined, the UNL code for the text is produced.
- Generation task: each source language must tune its generator to the new phenomena appearing in the text.
- Post-edition task: generated texts have to be revised by human post-editors, since no automatic translator or generator have (in this moment) enough quality to assure grammatical correctness and a natural and legible style.

These tasks were at the core of all the experiences so far. However, each of them has helped in one way or another to more concisely define the processes that are involved in multilingual generation and to bring into light some deficiencies of UNL.

2.1 Barcelona Experience (2001)

In the Barcelona experience, the original text was written in English and its approximate size was 3000 words. Lexicographic and codification tasks were shared among the four participant teams (Russia, France, Italy and Spain). There were continuous debates about definition of UW and codification issues among the teams. This process was fruitful for the most theoretical aspects of UNL (UWs and codification). The outcomes of such work were the definition of some common guidelines that will facilitate the unification of encoding styles.

However, the division and organization of work in this experience cannot be taken as a paradigm for competitive projects involving massive amount of contents, since the time and resources employed were out of any criterion of profitability. Certainly, experiences like Barcelona are extremely helpful to improve the bases for productivity and profit criteria. In the case of Barcelona, quality had priority over productivity.

2.2 HEREIN (2002)

Here the approach is different from Barcelona's. This experience tried to prove the UNL capacity for representing a big amount of contents coherently. The experiment was unilateral in the sense that the original text and the generated one involved the same language in order to update the rules of the language generator. The definition of UWs and UNL codification was undertook by one single team. In the codification work the guidelines produced during Barcelona experience was followed. The size of the text to be encoded was considerable around 12000 words dealing with many aspects of the cultural heritage of Spain.

An effective work requires a well trained team, and useful tools that could go from (semi-)automatic UNL editors to language generators. Work in Herein represents a borderline among what can be done and cannot with almost manual tools, dictionaries with reduced coverage and a generator with an acceptable quality, so that minor changes are required.

This time the novelty of the experiment lies in the fact that the contents were expressed in a complex type of language, resembling a legal style, which could occasionally yield more complex UNL representations that consequently would originate problems for deconversion. The produced UNL code in Herein, which was undertaken by just one team without intervention or consensus among other teams, could be posed difficulties to the generators of other languages, and even to any other expert codifiers. That is, the lack of uniformity in the process of codifying can yield UNL code not appropriate for real multilingual generation.

The main conclusion of this experiment is that the lack of agreement in the way to codify and the non existence of clear criteria for codification (like those following the spirit of the guidelines but more comprehensive) is the direct cause for an important loss of quality.

As a result of this experiment, it was established the need for the UNL teams to work together and cooperatively to define a definite Manual for Codification in UNL.

In the Herein experience, productivity increased but the overall quality decreased.

2.3 UNESCO (2003–2004)

This experiment was the first one that was developed in the laboratory context but under a contract that will demand results. It was the first contract for multilingual production using UNL. Apart from multilingual generation, the contract also included the measurement of productivity and associated costs. The objective was to establish a benchmarking that would allow for the establishment of some general definition of the processes of production and of the maximum costs associated to each process in any language. Taking into account the multilateral nature of Barcelona and the unilateral nature of Herein, this project was defined in between, as the closest model to achieve productivity in the medium term.

More concretely, the tasks for UWs production and UNL codification were assigned to a single team (with the associated risks of lack of consensus). The tasks for local dictionaries and generation along with post-edition were carried out by the other teams. The volume of contents was also considerable (15000 words) in the domain of World Heritage. For the first time, the codifying team used a UNL Editor that substantially accelerated this process and increased productivity up to the point of starting to define business models based on the use of UNL. In this case, there was neither debate nor consensus in principle but the produced UNL code could be improved with the feedback of other teams. The use of the tool for UNL edition was essential also for revision of errors (reaching 1 minute per sentence as average in the revision process, quite a distant measure from manual revision and codification).

The objective of UNESCO was the establishment of metrics for productivity in every process and task on the one hand; and on the other, specifying the processes that needed improvement and what sort of improvement. The results of this experience have been positive, although still they somewhat incomplete. The main issues that need to be improved in the nearby future are:

- A consensus should be reached when codifying into UNL as an essential condition for massive production.
- A higher degree of automatization in the lexicographic and codification process is indispensable. They require for clear standards in production that will help to alleviate the error rate in these two processes.
- A standardization of the processes that will allow for measuring costs and will make compatible the processes in different languages.

During both the Herein experience and the UNESCO experience the Spanish Language Centre attempted to measure the employed time in all the processes involved in multilingual generation. The processes are depicted in detail in the next section, whereas the obtained metrics and the results will be the topic of section 4.

3 Methodology

3.1 Overview: Context, Roles and Goals of the Methodology

This section contains a description of a general methodology for multilingual generation within the UNL system. This methodology is mainly derived from the multiple experiences involving UNL codification and targeting at multilingual generation carried out by the UNL consortium.

The purpose of the methodology is to show the main processes involved under the broad concept of “multilingual generation”. For the sake of generality, these processes have been described avoiding concrete procedures that depend on particular applications and technologies.

The common context where this methodology applies is that of a given customer (be it an institution or any particular customer) providing a document or set of documents in a specific natural language. For each document, it is required:

1. The UNL codification of the document (that is, a UNL document)
2. The generation of the UNL document into the number of natural languages that the customer establishes (multilingual generation *per se*).
3. The resulting bilingual Natural Language – UNL dictionaries of the involved languages (multilingual lexical resources).

In order to carry out these three main tasks, the methodology distinguishes two types of participants, according to the roles they play.

- **Coordinator**: The co-ordinator supports direct communication with the providers. The coordinating team will receive the original documents that will be codified into UNL and lately generated into several natural languages. Normally, the “working” language of the co-ordinator will coincide with the language of the provided documents. The reason for this equality in the language is simple: the co-ordinator is in charge of creating the relevant UWs and the UNL codification of the document.
  The general tasks that the co-ordinator carries out are:
  - **Vocabulary extraction** from the original documents (in the original language)
  - **Construction of the list of UWs** belonging to the complete vocabulary of the document (they are pairs of words)
  - **Codification** of the original document into UNL.
  - Distribution of aforementioned materials (UWs and UNL code) to the rest of participants.
  - Finally, elaboration of the project **documentation**, if needed.
- **Local Teams:** They communicate with the coordinator. Local teams are defined according to the language they work on. So, if generation assignments are required in three languages (say English, French and Spanish) there will be three local teams: English team, French team and Spanish team.
  The tasks of local teams are three-fold:
- **Creation of the pairs** (Headword-UW) according to the UWs provided by the co-ordinator. (local dictionaries).
- **Generation** of the provided UNL document into the local language.
- **Post-edition** of the generated language.

Please note that if one of the involved languages is the own language of the co-ordinator, these tasks also apply to the co-ordinator team. For example, if one of the involved languages is Spanish, being Spanish the “working” language of the co-ordinator team, the co-ordinator team will have to follow all the processes described for local teams.

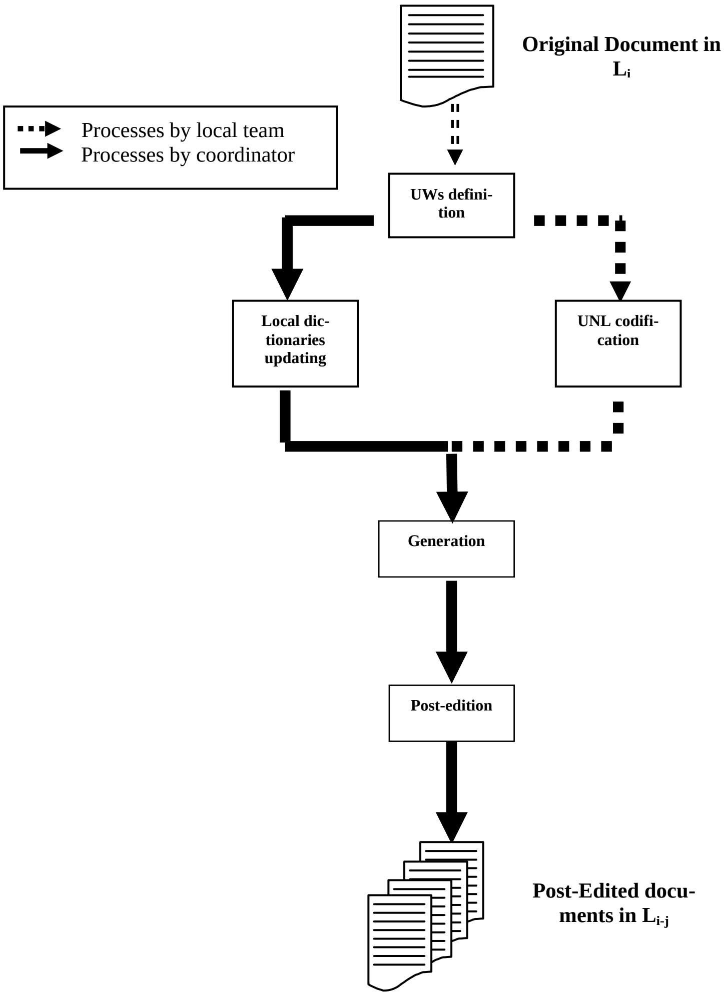


Fig. 1. Overview of the methodology.

The remaining subsections specify each process and subprocess that conform the methodology. For each process (or subprocess is the process is decomposable), the objectives, input and expected outputs are specified. As have been mentioned, no explanations or hints about how to perform these processes are included in the methodology, since such procedural information depends much on the state of the technology available for every language and for every local team.

The fact that this “know-how” information is not included does not mean, of course, that processes are to be performed without the help of specialized tools and software. In fact, some processes can be done automatically with the use of adequate tools. For example, some tools may be designed *ad hoc* to perform some processes like lexical extraction and lemmatisation (in Process 1) or instead the process can be done manually. Other processes (especially Process 3, language analysis) tackle very well known problems in the area of Natural Language Understanding, and thus the availability of tools and specialised software may vary from language to language and from team to team. For this reason, the methodology is not defined according to a given language processor or analyser, to the extent that the process could be performed with no machine aid at all. The same applies for Process 2 (updating of dictionaries), that heavily depends on the specific dictionary physical support and design of each team.

However, two issues should be pointed out for processes 4 and 5. Process 4 is fully automatic (that is, generation should be fulfilled automatically and with the lest amount of human interaction). On the other hand, Process 5 (as it will be explained) is a complete manual activity.

Finally, for clarity reasons some conventions has been used when referring to documents and different languages. These are the following:

- The document (or set of documents) provided by the customer will be referred to as <u>Original Document.</u>
- Such document is written in a specific language, referred to as <u>Language A</u>, or L_A as an abbreviation.
- The different natural languages involved in the methodology (those of the local teams) will be referred to as <u>Local Languages</u> or L_N as an abbreviation.

A general overview of the first level processes of the methodology is shown in Figure 1**.** A concise description of each process will be included in the remaining of the section, from section 3.2 to section 3.6. The presentation of both the general methodology and specific processes will be done according to the following schema:

- A description of process or subprocess.
- A table detailing the input and output of process or subprocess.
- A graphical representation of the process, showing the workflow, input and output.

3.2 Process 1: Definition of Universal Words

This process is decomposed in the following 3 subprocesses.

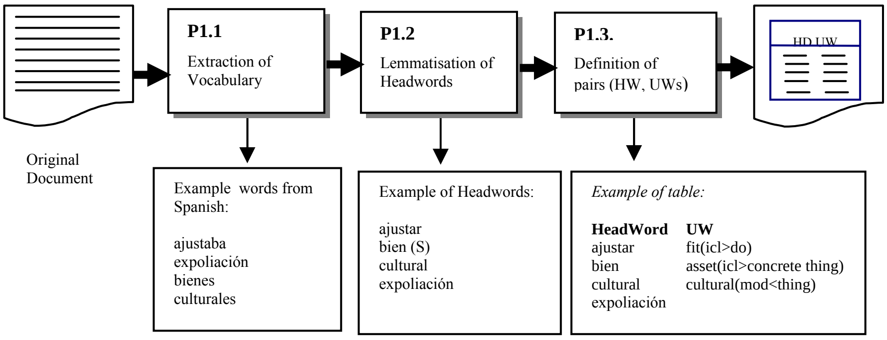


Fig. 2. Workflow for Process 1: Definition of Universal Words.

Sub-process P.1.1: Extraction of Vocabulary

Given a document, the relevant vocabulary (id est, lexical items or words) must be identified and extracted. For relevant vocabulary, it is understood lexical items that denotes concepts and thus have an equivalent Universal Words. Such lexical items are usually refers as “lexical categories” as opposed to closed-class categories (articles, auxiliary verbs, some prepositions, etc).

Input and expected output are detailed in Table 1.

Table 1. Input & Output of Subprocess P 1.1

INPUT	Original document in Language A.
OUTPUT	List of words belonging to the document that require a UW.

Sub-process P.1.2.: Lemmatisation

In the document, words appear inflected. That is, a verb may appear in the 3rd person singular of tense present in the subjunctive mood, or and adjective may appear in the feminine plural form. In this subtask, the inflected forms found in the document should be converted into headwords or lemmas. Lemmatisation is done in the following way:

1. For an inflected verb, convert it into the infinitive form.
2. For an inflected noun, convert it into the singular and nominative form (if case applies)
3. For an inflected adjective, convert it into the masculine singular noun.

Input and expected output are detailed in Table 2.

Table 2. Input & Output of Subprocess P 1.2

INPUT	List of words belonging to the document that require a UW.
OUTPUT	List of headwords that require a UW

Subprocess P.1.3: Definition of pairs

In this subtask, the pair (Headword L_A, UW) must be constructed. That is, for each headword of the list of headwords resulting from P1.2, the equivalent Universal Word must be identified.

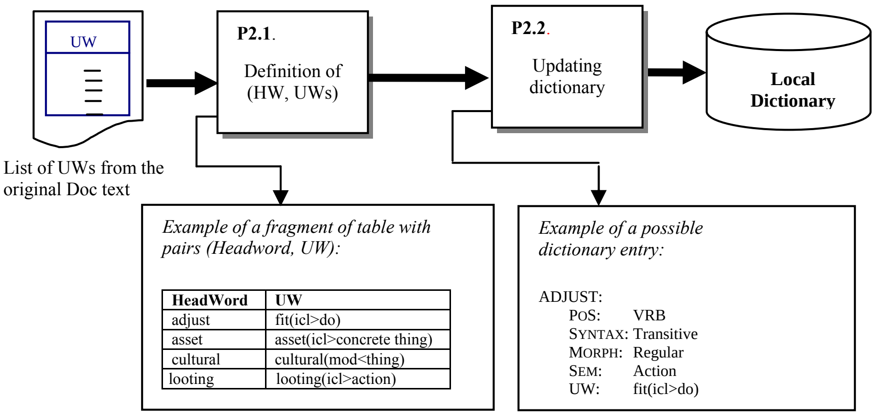
Input and expected output are detailed in Table 3.

Table 3. Input & Output of Subprocess P 1.3

INPUT	List of headwords, output of P1.2
OUTPUT	Table with the pairs (Headword L_A, UW) for the whole list

3.3 Process 2: Updating or Building Local Dictionaries

This process is decomposed in 2 subprocesses.


UW
List of UWs from the original Doc text
P2.1.
Definition of (HW, UWs)
P2.2.
Updating dictionary
Local Dictionary
Example of a fragment of table with pairs (Headword, UW):
Example of a possible dictionary entry:

HeadWord	UW
adjust	fit(icl>do)
asset	asset(icl>concrete thing)
cultural	cultural(mod<thing)
looting	looting(icl>action)

ADJUST:
POS: VRB
SYNTAX: Transitive
MORPH: Regular
SEM: Action
UW: fit(icl>do)

Fig. 3. Workflow for Process 2

Subprocess P.2.1: Definition of local pairs

In this subtask, each local team is provided just with the list of UW that has been resulted from the complete table, outputted in Process 1. The objective is to “find” the headwords belonging to the local team language that best fits into the UW. As a help, local teams can be also be provided with the original document and with the complete table with the pairs L_A – UNL. Note that this will be only helpful if the Language A is familiar to the local teams; otherwise, providing the original document and the complete table will have no apparent utility.

Input and expected output are detailed in Table 4.

Table 4. Input & Output of Subprocess P 2.1

INPUT	List of UWs belonging to the original document.
OUTPUT	Table with the pairs (Headword L_N, UNL)

Subprocess P.2.2: Updating or building the local dictionary

In this subtask, local teams must update their dictionaries and insert (or update) the adequate entries (the headwords identified in the previous table) together with the corresponding Universal Word.

Input and expected output are detailed in Table 5.

Table 5. Input & Output of Subprocess P 2.2

INPUT	Previous dictionary of the local Language - UNL UNL and table with the pairs (Headword L_N, UNL).
OUTPUT	Updated dictionary of the local Language - UNL

3.4 Process 3: Conversion into UNL

This process is decomposed in two subprocesses.

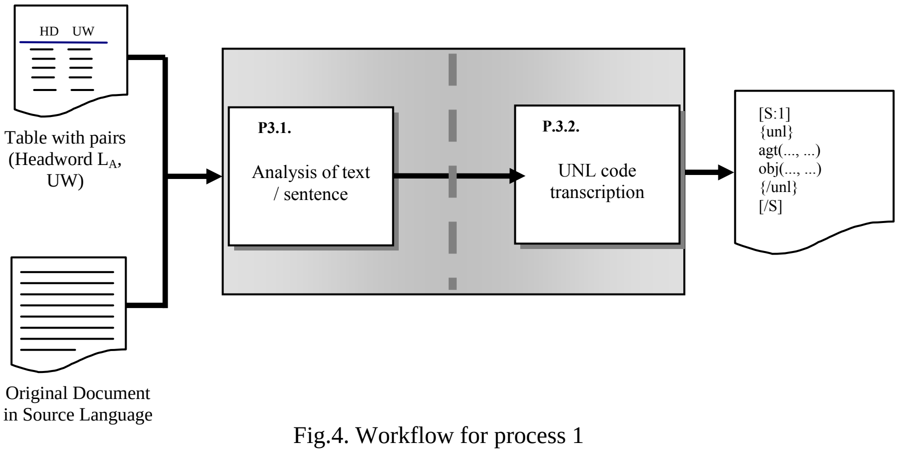

HD UW
Table with pairs (Headword L_A, UW)
Original Document in Source Language
P3.1.
Analysis of text / sentence
P.3.2.
UNL code transcription
[S:1]
{unl}
agt(..., ...)
obj(..., ...)
{/unl}
[/S]

Fig.4. Workflow for process 1

Subprocess P.3.1: General Understanding of the text

This is quite an analytical task, the objective is to comprehend the meaning of the text and how this meaning is expressed in the sentence, that is, to "understand" the grammatical and semantic relations of the text. Since UNL expressions correspond to sentences, this subtask is performed iteratively sentence by sentence.

The borderline between subtask P3.1 and P3.2 is rather fuzzy. Analysis of the text may be guided by the UNL final representation or it can be done more independently from the final UNL representation, simulating NLP components that carry out the analysis tasks in the following traditional processes:

- Morphological and Lexical analysis
- Syntactic Analysis
- Semantic Analysis

Be it that as it may, there are two clear conceptual processes: and analytic one, and a "transforming" one: transform the meaning of the sentence into a UNL representation. Table 6 specifies input and output for this subprocess.

Table 6. Input & Output of Subprocess P 3.1

INPUT	Original document and list of pairs (Headword L_A, UNL)
OUTPUT	Abstract representation of the meaning of the sentence*

Please note that this subtask may not have a physical output, this "abstract representation" can be allocated in the head of the codifier.

Subprocess P.3.2: UNL ENCODING

This subprocess is the "transformation" of the abstract representation of the sentence obtained in P.3.1 into the UNL representation according to the UNL specifications and codification manuals if available. In this subtask, also document markers should be included in the final UNL document. Input and expected output are specified in table 7.

Table 7. Input & Output of Subprocess P 3.2

INPUT	– Abstract representation – UNL specifications
OUTPUT	UNL document (corresponding to the original document).

Figure 4 shows the workflow of process 3. The grey box in the graphic representation of the process simply gives account of such fuzziness in the separation of both processes.

3.5 Process 4: Generation into Local Languages

This process consists on the generation of the UNL document (output of P.3) into the local languages. This process is not decomposable, since generation is performed automatically. Each local team should be provided with language generators that will actually perform this task. Inputs and outputs to the process are presented in Table 8. The workflow of the process is illustrated in figure 5.

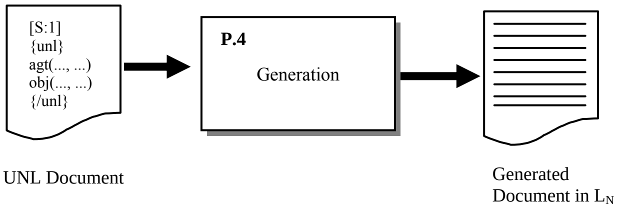


Fig. 5. Workflow for process 1

Table 8. Input & Output of Subprocess P 4

INPUT	– UNL document – Updated local dictionary
OUTPUT	Document with the raw generation of the original document in the local language

3.6 Process 5: Post-Edition

Since language generators may occasionally produce incorrect language, or at least, a low quality language (incorrect style, non fluent language, etc.), texts are post-edited. Post-edition consists merely on giving “style” to texts, that is, making them natural. At this moment of the technology, this task is performed entirely manually. As usual, input and outputs of this process are gathered in table 9, whereas the graphical representation of the process is shown in figure 6.

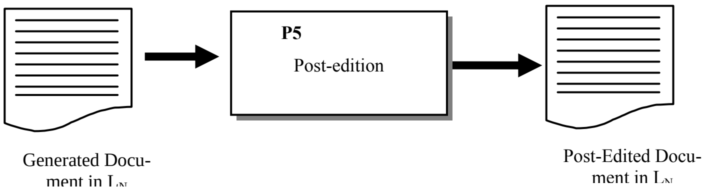


Fig. 6. Workflow for process 1

Table 9. Input & Output of Subprocess P 5

INPUT	Generated Document in the local language.
OUTPUT	Post-edited document in the local language.

4 Results and Conclusions

These processes are necessary for establishing a benchmark in order to evaluate the productivity of global processes for UNL to become a firm candidate to support multilingual services in the market. However not only the definition and description of each involved process is required, productivity cannot be accounted for without defining explicitly its associated costs, measured (usually) in time. Thus the definition of metrics associated to each process in the global methodology is of paramount importance, so that there will be no business future in UNL without a way to evaluate costs, which inevitably involves measuring tasks.

Metrics, evaluation, validation, etc. are quite obscure fields in NLP; however there is not any engineering product or project that is thrown into the market that obviates metrics. UNL cannot be an exception.

There are several aspects that may hinder a straightforward establishment of metrics in the UNL contexts. These are:

– The non uniform nature of the UNL Consortium. We have different systems, different dictionaries, different generators, and different tools. At this point we could think that it is not comparable the time employed in creating a lexical entry in a x-uw.txt dictionary or in a dictionary in another system (say ETAP or Ariane). Likewise, analyzers and editors are different from team to team.

- The degree of expertise of the actors in charge of the processes. Obviously, a higher degree of expertise will reduce the extra load time for review in all processes.
- Until clear and definite instructions for building UWs and for codification into UNL is made, metrics for the overall UNL enconversion process will be flawed.

In spite of all this apparent drawbacks, the Spanish Language Centre noticed the urgency and need to begin establishing metrics for all the processes exposed in the methodology (section 3). Almost all processes were measured in time, especially in the following tasks:

- Construction of UWs
- Construction of dictionaries entries
- UNL codification
- UNL post-edition

Measures were taken in two different domains and experiences: Herein and UNESCO, with different actors showing different degree of expertise, and different available tools in the enconversion task. Let's have a look at the results.

4.1 Metrics in the Enconversion Process

The context of Herein is the following:

- No proper tools available for UNL enconversion, the available tools were either too rudimentary or not robust enough to undertake a massive codification task. Therefore the codification process was made mainly manually. This implies almost the same amount of time in reviewing the code (in particular reviewing syntactic aspects of UNL expressions).
- The degree of expertise in UNL encoding was acceptable (no need for prior training).

When measuring the employed time for codification, several decisions have to be taken: are we interested in measuring time to encode a text, a sentence, a paragraph or simply the number of words? Since these matters were not very clear, it was decided to take into account the time of enconversion per sentence, thus obtaining a correlation between sentence/time for codification.

The sentences extracted in Herein showed an average length of 20 words and an average time of 4'8 minutes per sentence. If counted on total values, the 16 sentences amounts to 322 words, and the total time to codify all the sentences was 77 minutes, which means 14'4 seconds per word.

At this point, it has to be remarked that the UNL code in Herein was produced manually, needing ulterior revision and requiring additional tools to catch up syntactic errors.

On the other hand, the UNESCO metrics differs in two main aspects: the degree of expertise and the available tools. In UNESCO, there exists data for a total of 116 sentences. The total amount of words in the 116 sentences is 3178. In this case, the average length of the sentences is superior to Herein, the sample of the sentences shows

27’4 words average length. The arithmetic average of time of codification per sentence is 9’95 minutes. When taking into account total facts (total of words and total of time), there results in 21 seconds per word. Data for UNESCO is summarized in table 10.

Table 10. Results of the metrics taken in the Unesco experience

Number of sentences	116 sentences
Total number of words	3178 words
Average length of sentences	27’4 words
Total time for enconversion	1155 minutes
Average time for enconversion of a sentence	9’95 minute /sentence
Time for codification of a word	21 seconds

As can be observed, there is a significant increase in time of codification per sentence. Common sense will make us predict that, due to the use of edition tools, there would a significant improvement in the time of codification; however, there is not. A possible reason for this is that the length of the sentence may interfere in the time of codification (being shorter sentences easier to codify than longer sentences) and the degree of expertise. That is, the difficulty in codifying may be related to the domain and type of language used in the domain.

Further, one does not have to forget that the UNL code obtained in UNESCO was syntactically, at least, correct. Whereas the UNL code obtained in Herein required subsequent syntactic revision.

4.2 Metrics in the Post-edition Process

Post-edition, as conceived in the UNL context, has to be carried out manually completely. In the metrics for the post-edition process there were involved two different actors and different types of domains as well. The actors varied in the degree of expertise, from a native speaker of a language to a professional translator.

Regarding the native speaker of the language to be post-edited, the average time to post-edit a sentence showed a striking uniformity: disregarding the domain, the average time for post-edition of a sentence is 1 minute.

The data collected by a professional translator is summarized in table 11, being the most significant conclusion a considerable descent in time.

Table 11. Specific data in the post-edition process by a professional translator

Number of sentences	164 sentences
Total number of words	4188 words
Average length of sentences	25’7 words
Total time for post-edition	120 minutes
Average time for post-edition of a sentence	45 seconds
Time for post-edition of a word	0’6 seconds

4.3 Metrics in Lexicographic Processes

For the construction of UWs, bilingual and monolingual dictionaries were used and the metrics obtained pertains to just one actor. The average time was 3 minutes for the construction of an UW and 1 minute for the construction of a lexical entry in a dictionary x-uw.txt type. This data applies both to Herein and UNESCO experiences.

5 Conclusions

For the time being we cannot say that we dispose of reliable, systematic and trustworthy metrics. As can be seen, there are a lot of parameters that influence in the final metrics. Some of them are expectable (like the degree of expertise or the use of tools) but other (like the linguistic particulars of a given domain) may be not so obvious, and even debatable. In such a heterogeneous context like the UNL consortium, all these hidden variables have to be made explicit and taken into account when establishing common metrics and common reference times for us all.

The metrics and times presented here are, of course, not definite. However, they hint at the possible maximal boundaries of the time to be employed in each process that should not be surpassed by any team in the UNL consortium in order to achieve a minimum degree of productivity. The objective of the metrics and of the definition of a common benchmarking is to determine the minimum time required for the several process so that a cost evaluation can be done. Such evaluation would be as a reference for the others. It is a very critic point for the exploitation of the UNL to acknowledge the most competitive costs we can have.

References

1. Jesús Cardeñosa; Luis Iraola; Edmundo Tovar; (2001) "*UNL: a Language to Support Multilinguality in Internet*" International Conference on Electronic Commerce. ICEC2001. Vienna. Austria, 2001, 9 pp.
2. Uchida, H. ATLAS-II: A Machine Translation System using Conceptual Structure as an Interlingua. Proceedings of the Second Machine Translation Summit, Tokyo, 1989.
3. Muraki, K. PIVOT: Two-phase machine translation system. Proceedings of the Second Machine Translation Summit, Tokyo, 1989.
4. Beale, S., S. Nirenburg and K. Mahesh. Semantic Analysis in the Mikrokosmos Machine Translation Project. Proceedings of the 2nd Symposium on Natural Language Processing. Bangkok, Thailand, 1995.
5. Nyberg E y Mitamura T. The KANT System: Fast, Accurate, High-Quality Translation in Practical Domains. En Proceedings de COLING-92: 15th International Conference on Computational Linguistics, 1992.
6. Arnold, D.J., Balkan, L., Meijer, S., Humphreys R.L., and Sadler, L. Machine Translation: an Introductory Guide, Blackwells-NCC, London, 1994, clwww.essex.ac.uk/MTbook/HTML/.
7. Uchida, H. The Universal Networking Language Specifications v3.2, 2003, www.undl.org.

8. Cardeñosa, J., Tovar, E. A Descriptive Structure to Assess the Maturity of a Standard: Application to the UNL System. Proceedings of the 2nd IEEE Conference on Standardization and Innovation in Information Technology. Boulder, Colorado USA, 2001.
9. Boguslasvsky, I. Some remarks on the UNL encoding conventions. Proceedings of the First International UNL Open Conference "Building global knowledge". SuZhou, China, 2001.
10. Cardeñosa, J., Iraola, L. and Tovar, E. Managing a real implantation of the UNL System. First International UNL Open Conference "Building global knowledge". SuZhou. China, 2001.
11. European Commission. HEREIN Project (IST-2000-29355). Final Report (2003)
12. Cardeñosa, J., Gallardo, C. UNESCO Project: General Methodology for UNL conversion and multilingual generation. Technical document". Spanish Language Centre. Facultad de Informática. UPM. Spain, 2003.

Knowledge Representation Issues and Implementation of Lexical Data Bases

F. Sáenz and A. Vaquero

Departamento de Sistemas Informáticos y Programación, Universidad Complutense de Madrid,
E-28040 Madrid, Spain
Tel: +34913947622
Fax: +34913947529
{fernan,vaquero}@sip.ucm.es

Abstract. We propose to apply classical development methodologies to the design and implementation of Lexical Databases(LDB), which embody conceptual and linguistic knowledge. We represent the conceptual knowledge as an ontology, and the linguistic knowledge, which depends on each language, in lexicons. Our approach is based on a single language-independent ontology. Besides, we study some conceptual and linguistic requirements; in particular, meaning classifications in the ontology, focusing on taxonomies. We have followed a classical software development methodology for implementing lexical information systems in order to reach robust, maintainable, and integrateable relational databases (RDB) for storing the conceptual and linguistic knowledge.

1 Introduction

Due to the immaturity of the knowledge representation topic, lack of standardization is broadly felt as a very undesirable state into the community around language resources [LREC 02]. For instance, standard terminology for a common reference ontology is yet a goal to be reached. There is no doubt about what lexicon means, but ontology is differently understood in the computational linguistic literature. For instance, WordNet is mentioned as an ontology [USC 96], CYC is provided with a formal ontology [PRI 01], etc. Here, ontology, in a LDB, is the set of concepts in the domain of the base and the relationships that hold among them, without including linguistic knowledge, and common to all of the languages supported in the base.

Weak attention has been paid on topics about development methodologies for building the software systems which manage LDB, and dictionaries in particular. We claim that the software engineering methodology subject is necessary in order to develop, reuse and integrate the diverse available linguistic information resources. Really, a more or less automated incorporation of different lexical databases into a common information system, perhaps distributed, requires compatible software architectures and sound data management from the different databases to be integrated. The database subject have already done a long way reaching a strong standardization, and supplying models and methods suitable to develop robust information systems. We apply RDB design methodologies to develop LDB consisting of ontologies and

© J. Cardeñosa, A. Gelbukh, E. Tovar (Eds.)
Universal Network Language: Advances in Theory and Applications.
Research on Computing Science 12, 2005, pp. 430–442.

lexicons. The conceptual knowledge is represented as an ontology, and the linguistic knowledge, depending on each language, is stored in its lexicon.

Subjects about electronic dictionaries for diverse natural language processing applications have been extensively studied [ZOC 03], [WIL 90], [WIL96], as well as LDB [MIL 95], world knowledge bases [LEN 90], ontologies in general [ONT], ontologies for computational linguistics [NIR], and the like. But there are no references on how these information systems have been developed and upgraded along their life. Moreover, tools for managing ontology-based linguistic information systems have been described [MOR 02], but there is no a declared software engineering approach for the development of these tools.

We follow the classical RDB design based on the conceptual, logical, and physical models for building LDB, and software engineering techniques based on UML for building LDB interfaces (these are not described in this paper). The result is a methodology to develop information systems for building and querying LDB [SV 02]. Based on this methodology, we have developed software tools for authoring and consulting different kinds of linguistic resources: monolingual, bilingual and multilingual dictionaries. In this paper, we detail the conceptual development of a bilingual dictionary with relational technology.

Conventionally, dictionaries are conceived for human use and lexical databases are conceived for natural language processing (NLP) applications. Our methodology leads to friendly usable dictionaries, but structurally prepared to be easily embedded in computer applications, as we show along the paper.

The rest of the paper is organized as follows. Conceptual and linguistic requirements embodied in the lexical and ontological resources are first exposed in section 2, because of their relevance in building different lexical databases, such as electronic dictionaries, and distinguishing certain relevant aspects of our approach from others. The next section introduces how to apply the relational design methodology to develop LDB, and section 4 details its application to a bilingual dictionary. Finally, in section 5 certain conclusions are summarized and future work is foreseen.

2 Conceptual and Linguistic Requirements

In this section, conceptual and linguistic knowledge incorporated in computing systems devoted to NLP are pointed out because of their relevance in the definition of the conceptual model showed below.

Regardless of the language, the knowledge in the discourse universe is conventionally divided in two classes: conceptual and linguistic. Terms and sentences refer to concepts, but they have particular structural and morphological features in each language. All of this information is not available in any dictionary, electronic or not, although it is the objective in the most exigent ontology-based linguistic Knowledge Bases, such as MikroKosmos [MIK].

In the next paragraphs, we limit the conceptual and linguistic knowledge to the level we are interested in. Then, we show the structure of these two kinds of knowledge, and how both are linked.

2.1 Lexicographic Order. From Paper to Electronic Dictionaries

No kind of term order is suitable for electronic dictionaries, because a random direct access is better than alphabetical sequential access for human use. The first generation of electronic dictionaries [COW 99] is characterized by the direct access to terms, but the provided information and the ways for accessing to it differ from one dictionary to other, having unclear (not formally specified) structure and lack of declared methodology. The new generation dictionaries intend to cover these holes.

2.2 Terms and Meaning. Polysemy and Synonymy

In every language there exists the well known naming problem [KAT 93], which consists of two elements: one is polysemy (under the synchronic point of view, that is, embodying polysemy itself and homonymy), by which a term can have several meanings; and the other is synonymy, by which one meaning can be assigned to different terms. We are going to study in the next section how to relate terms and meanings. The naming problem will be automatically solved by completely separating Lexicon from Ontology, as we shall see.

2.3 Semantic Relationships and Lexicon

Each meaning of a given term is precisely identified by its semantic category (category from now on, for the sake of brevity). Therefore, categories provide classification for meanings, and such classification can be arranged in a taxonomy [RK 02]. Here we do some remarks about the relationships among categories, meanings and terms. On the one hand, a given term can belong to several categories under different meanings. On the other hand, a given term can belong to several categories under the same meaning. We must also note that a category has a meaning described by a definition. This meaning is the extensional definition of the category. See [SV 02] for more details.

2.3.1 Lexical Databases

For a given language, we have a set of terms, meanings and categories holding certain relationships among them. Conventional LDB, such as WordNet [MIL 95], have term classification through synonymy (grouped in the so-called synsets). LDB based on ontological semantics go beyond by playing the role of meaning taxonomy and supporting more complex semantic relationships [NIR 95]. All of the relationships (meronymy, holonymy, hypernymy, hyponymy, and so on) represented in the more complete lexical databases, such as WordNet or EuroWordNet [EWN], are also represented in ontology-based databases, such as MikroKosmos; but in this case, all of the concepts and their relationships are present in the ontology, while each lexicon has the terms for each language and their linguistic arguments, as well as the links with the concepts into the ontology. The mapping between ontology and lexicon is the key for successfully coordinate all of the lexical and semantic relationships. This approach does full separation between ontology and lexicon. If we now think of several languages, the same ontology applies for each one of the lexicons.

Other approaches has been adopted. Each one leads to a more or less complex LDB structure. We claim that the ontology-lexicons approach is the most appropriated to reach a simple, robust and controlled LDB structure, prepared to be reused in different applications and integrated with another ones with the same structure.

The architecture ontology-lexicons is criticized in [POL 03], given that each language has its own lexical semantics. Then, strictly speaking, there is no one single ontology independent of the considered languages. In favor of our position, we argument that the fact of the nonexistence of one single ontology common to diverse languages is independent of assuming one imposed undesirable a priori hierarchy, which is considered in [POL 03] as unavoidable considering the common ontology approach. But in our methodology, the hierarchy (taxonomy) is incrementally created when building the LDB. For a monolingual database (French in the case of the DiCo LDB), there is only one ontology; thus, there is no problem. However, certain problems could arise in multilingual LDB, because the boundary between ontology and lexicon does not appear clearly always. There are many ways to face up these problems considering other approaches different from ours, when the ontological semantics is distributed among the different languages at multiple levels. For instance, in the Papillon project [MAN 03], the different languages are linked to a common dictionary of meanings (axies in French). In the EuroWordnet project, the different WordNets (one for each considered language) are linked by two levels of common concepts, and the resulting structure is not appropriated for the multilingual applications. In MILE [ABB 02], SIMPLE templates play the role of ontologies; so the resulting LDB structure is more complex than that resulting from the ontology-lexicons approach.

We adhere to the criterium from [MAH 95] conceiving ontology as a language-neutral body of concepts. In this case, the problems can be solved putting in each specific lexicon the own lexical-semantic information required, which is not present in the common ontology [VIE 98]; so the ontology is the conceptual model of the domain and each lexicon is linked to the same ontology. From this approach, the system design to develop LDB is enhanced in robustness, because an architecture with two abstraction levels is reached.

From this approach we apply very carefully the RDB techniques to reach a methodology assuring a sound and simple structure of the LDB, and a controlled way for building any particular LDB through an administration interface. This work is indeed previous to the formal definition of an interlingua [FAR 04]. We are far from reaching this goal in general. But there are a lot of NLP applications, not only monolingual, that do not need formally and completely represent the text meaning. We claim for reaching an interlingua in the future from LDB conceived from the ontology-lexicons approach and developed with our methodology.

Our presented ontology gives a quite limited interlingua since we focus only into a single relationship, as exemplified in the LDB for the bilingual dictionary later on. As more semantic relationships are added to this ontology, more expressive interlinguas can be reached. Then, a complete interlingua could be developed when all of the semantic relationships in the natural language were embodied into the ontology.

Another central idea in this work is to develop for each group of applications one LDB, the most appropriated one. Certain applications are more exigent of linguistic resources than others. Why to use the same LDB with all of the linguistic resources for no matter what application? It is more efficient to use a LDB with the required linguistic resources depending on the application, as we propose. This vision contemplates, besides our methodology to build different LDB, building subsets of LDB already build as 'views' of the DB. In this case, the LDB should be developed from the ontology-lexicons approach. Then, particular LDB can be extracted from the more general one. We claim for this way in order to integrate different LDB.

2.3.2 Our LDB for Dictionaries

In this approach, relationships among terms from different languages come from considering jointly the involved ontology-lexicon schemes, as we will see later when considering the bilingual dictionary. In the dictionary here considered, the ontology only consists of one relationship which gives tree-structure to the conceptual taxonomy. A taxonomy is a natural structure for meaning classification. Each node in the taxonomy corresponds to a category. In principle, every category in the taxonomy can have meanings, regardless of its taxonomy level. It must be noted that every category in the taxonomy contains at least the term which names the category, so that all categories are non-empty. On the other hand, the creation of new categories as belonging to several predefined ones should be avoided, in order to reach a compact relationship as the taxonomy structuring backbone. Next sections show the development of a dictionary without overlapped classifications [RK 02], and only permitting tree-structured taxonomies. Since a meaning can belong to different categories, the extensional definition of categories is hold [SV 02].

When consulting or building dictionaries, there are a number of advantages in classifying meanings as taxonomies. First of all, meaning taxonomy is a useful facility for an electronic dictionary, because meaning classification embodies additional semantics, which provides more information to the user than usually provided. As long as we know, this kind of facilities (meaning classification), normally used in conceptual modeling through ontologies [MCG 00], has not been implemented before into dictionaries.

One demanded facility in electronic dictionaries is the semantic relationship 'See' among terms. When a definition for a term A in a dictionary has the entry 'See B' (B is another term) it only refers to B, not the particular definition for B the author thought of, so that the user has to read all the definitions assigned to B until he reaches the intended one. Section 4 shows how we solve this problem in our approach.

Along the next sections, we propose how to accomplish the conceptual and linguistic requirements into a LDB for electronic dictionaries by using a sound design methodology.

3 Designing Lexical Databases with Relational Technology

We understand lexical databases as information systems which are composed of a database core and an application layer which allows the user and applications to interact with the lexical data. On the one hand, the justification for having a database core instead of other file related approaches comes from well-known issues in the database community (e.g., see classical texts as [SKS 02]). In particular, we do need integrity constraints for maintaining consistency when modifying data. On the other hand, the application layer should be understood as possibly containing user interfaces for both consulting and modifying lexical data, as well as NLP applications. When considering these two components of the information system, we do isolate data from applications, so that all consistency checking is encapsulated into the database core.

Both components should be developed following known software engineering methods. It is more likely to find these methods applied to the application layer, but, in general, we do not find them applied to the modeling of lexical databases.

In our work, we focus on relational databases because of a number of reasons: they are widely used, efficient RDBMS (Relational Database Management Systems) are available, and a database design methodology has matured for them. The latter is the most important point we highlight, since it provides several design stages which help in designing consistent (from an integrity point of view) relational databases. This methodology comprises the design of the conceptual scheme (using the Entity/Relationship (E/R) model) and the logical scheme (using the relational model). A final stage is the physical scheme, which is generally omitted in the literature since it depends tightly on the target RDBMS. This work only describes the first design stage.

We emphasize here the dependence between the design stages and the DB structure. Besides, the way to build a LDB comes through this dependence, as is expressed in section 4 after considering the constraints in section 3.1.

In other projects of LDB, when RDB techniques have been applied, there is no awareness of how this dependence is crucial to establish a development methodology and a formal common DB structure. We take two examples as representative samples.

In [MOR 02], an E/R model is defined, but there is no expressed relation between the development stages and the DB creation.

MILE [ABB 02] uses an E/R model in the lexical entry for automatically generating a RDB with different purposes. Our approach leads to very different E/R models, with less complexity. Besides, the development of their DB is not described neither the integrity constraints.

3.1. Constraints in Relational Design

The relational database design methodology is not only focused on representing data and their relations, but more important for us in this work, constraints about them. These constraints allow us to impose restrictions for both data and relations that any database instance must obey. Although these constraints can be implemented in the application layer, we advice against this. We claim that they must be implemented in the database core because consistency would be maintained by the RDBMS, instead the applications. By this, the constraints encapsulated into the database are independent from the applications. Next, we introduce the constraints at each design stage which are useful for our purposes.

The E/R model is the most common tool for the first design stage, the conceptual modeling, allowing several kind of constraints which we relate with the constraints needed in a lexical database, since there are several (philosophical) notions that such an ontology-based database has to represent (e.g., identity and membership [GW 00]). Primary and candidate keys are used for the identity concept, i.e., given a class (entity set), every instance of the class (entity) can be unambiguously identified. Domain constraints play the role of defining valid characteristic properties that entities can have. Cardinality constraints restrict the number of entities a given entity can be related to, which is useful, for instance, for restricting graphs to trees in taxonomy classifications (membership property). A total participation for an entity set in a relationship set impose that every entity in the entity set must be in the relationship set. Unique constraints are related to primary key constraints in the sense that they represent unique values for properties that an entity in an entity set can have. Besides these constraints supported directly in the E/R model, other constraints for this stage can be completed by using natural language descriptions or a more formal specification language. These constraints are passed to the next design stage.

The relational model used in the second design stage, in turn, offers several kinds of constraints, inheriting some of the E/R model, such as primary and candidate keys, domain, cardinality, and unique constraints. In addition, we have referential integrity constraints, and functional dependencies. Referential integrity constraints are used for several purposes: to restrict the values a property (attribute) can take from a given set defined in an entity set (which can be understood as a dynamic domain definition in the sense that the domain can change by modifying the instance relation), and to restrict the possible entities a given entity can be related to. Functional dependencies are useful for imposing cardinality constraints among attributes of an entity, although, usually, they are only used in the normalization process for finding decomposition anomalies.

Constraints at the final database design stage, whose result is the physical model, depends on the RDBMS considered, but usually we find primary keys, candidate keys (by means of indexes with unique keys), domain constraints, referential integrity constraints (used, for instance as the basic cardinality constraints one to one, one to many, and many to many), which can be deferred to implement total participation. Moreover, constraint predicates can be stated in the state-of-the-art RDBMS by means of the CHECK clause and triggers. In this way, the designer can implement, among others, functional dependencies.

Because of the authoring nature of lexical databases, we cannot impose all of the identified constraints (since there is absent information which can be known afterwards). Therefore, we are ought to provide consistency checking features to the lexical database authors. These features must inform the author about authoring constraints which are violated by the instance database. Such constraints which may be violated during the authoring are known as soft constraints, by contrast with the hard constraints that every database instance must hold at any time.

Forthcoming sections show how to apply this design methodology to the development of a consistent lexical database. The next section shows a lexical database for a bilingual dictionary, which can be instantiated for a monolingual dictionary, and can be generalized for a multilingual one.

4 Designing a Lexical Database for a Bilingual Dictionary

As stated in former sections, we are interested in the representation of language information from an ontology point of view in order to build a lexical database, and, in this section, for a bilingual dictionary. First of all, we need to represent the meaning (concept) as a language independent entity, so that a set of terms (the so-called synonym set – synset in WordNet) in a given language is used to identify such a meaning. In this way, the synonymy property holds for the set of terms in a particular language related to a meaning. Further, a synset for each language can be found. Polysemy comes from the fact that a given term may be in different synsets for the same language (obviously related to different meanings). Finally, we are interested in classification of meanings, which can be represented with categories related to meanings, so that each meaning belongs to a category. If we restrict classifications to taxonomies, we have to impose a constraint stating that a category can only have a parent category, and only one category (root category) can have no parent.

Since we are interested in an ontology-based lexical database, we must highlight some points. Meanings are directly related to categories, instead of terms. Synonymy is a set-oriented property of terms, and the set itself is related to a meaning, instead of each term in the set. A term in a synset belongs to a category via a transitive relation among the synset, the meaning the synset belongs to, and the category the meaning is classified under. In order to fulfill the intensional definition of categories explained in section 2.3, a meaning is needed for defining each category, and a non-empty synset is needed for such a meaning.

4.1. Conceptual Design for the Bilingual LDB

Following these premises, we propose the E/R scheme shown in Figure 5 (an upgrade from [SV 02]) as a result of the first stage design (conceptual modeling). In this figure (following some recommendations in [PRE 97, SKS 02]), entity sets are represented with rectangles, attributes with ellipses (those which form a primary key are underlined), and relationship sets with diamonds, which connect entity sets with lines. Undirected lines (edges) represent a many to many mapping cardinality. A one to many mapping cardinality from entity set A to entity set B is represented by an arc from B to A, meaning that an entity belonging to B is related at most with an entity in A. A total participation of an entity set in a relationship set is represented by double lines. Undirected lines also connect attributes to entity sets. Relationship set and entity set names label each diamond and box, respectively. Each side of a relationship set relating an entity set with itself is labeled with its role.

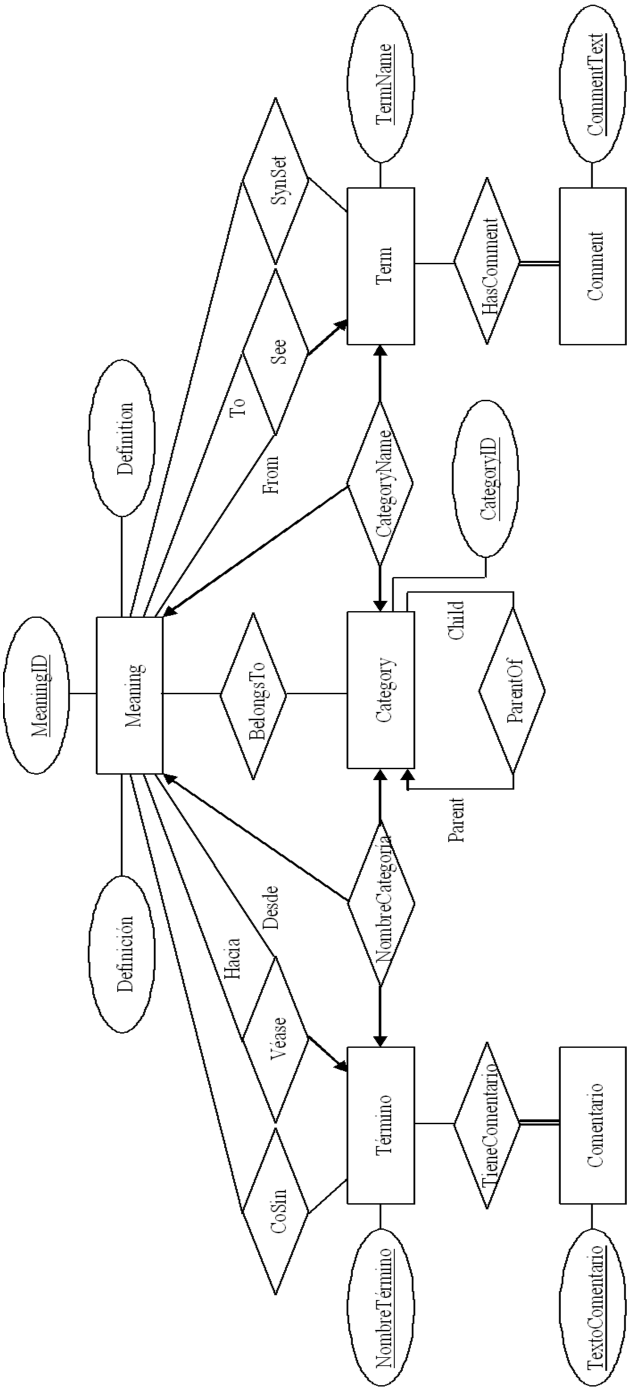


Fig. 1. Entity-Relationship Scheme for an English-Spanish LDB

In this figure, we show an instance of a simple bilingual lexical database for Spanish and English. In the following, we describe entity sets and its attributes, relationship sets, and constraints.

Entity Sets. The entity set Meaning is the central entity set other entity sets rest on and has three attributes: MeaningID (artificial attribute intended only for entity identification as shall be explained later), Definition and Definición, intended for the textual definitions of the meaning in both languages, English and Spanish, respectively. The entity set Term represents all of the English terms that compose the lexical database, and it has one attribute: TermName, which denotes the textual name of each term in this set. The entity set Category denotes the category each meaning belongs to, and it has one attribute: CategoryID (similar to MeaningID). The entity set Comment represents the comments about each term, and it has the attribute CommentText, which holds the textual comment for each term in this set. This entity arises from our need to develop a dictionary which can hold comments about terms in particular, not related to the concept itself (for instance, comments about the origins of the term). The entity and relationship sets from the Spanish language (CoSin, Véase, Término, TieneComentario, Comentario, and NombreCategoría) are homologous to the ones in English (SynSet, See, Term, HasComment, Comment, and CategoryName, respectively).

Relationship Sets. The relationship set SynSet between Meaning and Term denotes the English synonym set. The relationship set See denotes the semantic relationship ‘See’ among two meanings and a term (given a meaning, the user is referred to a representative term of another meaning, which is linked with the former via the relationship 'See'). The relationship set BelongsTo between Category and Meaning is used to categorize meanings, and it embodies the fact that our classification is not lexical (there is not a direct relationship between Category and Term) but semantic (we relate meanings to categories, i.e., we categorize meanings). The relationship set ParentOf is used to represent taxonomies. The relationship set CategoryName is intended to relate a category with the term which names it, under the meaning that defines the category. The relationship set HasComment links comments with terms.

Constraints. Mapping cardinalities are as follows: SynSet is many to many since a synonym set may contain several terms, and a term may be contained in several synonym sets (obviously, with different meanings). The ternary relationship set See which connects Meaning (two times for the “from” and “to” parts) and Term is many to many because a meaning may refer to several English terms, and one term may be referenced by several meanings. BelongsTo is many to many since many meanings are in a category, and a meaning could be in several categories (this situation is expected to be reduced to the minimum since our goal in developing dictionaries is to keep the classification as disjoint as possible). ParentOf is one to many since a given category has only one parent, and a given category can have multiple children. CategoryName has cardinality one for the three entity sets related because terms, meanings, and categories are unique in this set. HasComment is many to many since a term may have several comments attached and a comment may refer to several terms

Note that there are less total participation constraints that one could expect, all of them derived from the incremental creation of a database instance, because of the following reasons. A meaning does not have to be categorized. A meaning does not have to have a term for its representation in *one* language (if we create a meaning, it is likely to have at least a term in a language for its representation, but not necessarily in both languages). A category may have no name (a term) in a given language provided that its name is defined in the other language. A category does not have to have related meanings. Finally, ParentOf has no total participation since a category may have no parent (the root category), and a category may have no children (leaf categories).

A consistent LDB should hold total participation for the former constraints but they should be considered as soft constraints since they can be violated during the authoring process. We can identify other soft constraints which cannot be expressed with E/R-related constraints. For instance, a given meaning must have synsets in *both* languages in order to find translations, categories must be arranged in a tree, and all of the categories must have names in *both* languages. These constraints which cannot be expressed with E/R constructors are known as predicate constraints.

All of the attributes, but Definition and Definición, are primary keys. This means that they have an existence constraint automatically attached. But, if we consider that, for instance, a meaning is added to the database, it can be from any of the two languages, i.e., the LDB designer may have an English or Spanish definition for it. Although we can think of the attributes Definition and Definición as candidate keys, they cannot be since the null value will be, in general, in any of them. Therefore, an extra attribute is needed for identifying this entity set, which we call MeaningID. In the physical model, these attributes must have a type for identifiers (such as the sequences or autonumbers). From the discussion above, we should also impose soft existence constraints (for instance, there should be a definition for each meaning) and hard uniqueness constraints (each definition must be different) for Definition and Definición.

We have also developed (but not shown in this paper) the logical and physical schemes for the design of our lexical database, which also follow the classical database design that ensures us a formal way of defining the database that the tools will adhere to.

5 Conclusions and Future Work

Continuing with the refinement of our development methodology of information systems for lexical databases, an elaborated and well sound design method has been presented here. The design is based on the ontological semantics approach, and we have signaled the advantages of this approach in face of the non-ontological one. The design has been tested and used to complete the development of certain information systems to build and consult monolingual, bilingual and multilingual dictionaries.

Of course, the advantages of applying software engineering principles and methods to information systems for lexical databases are evident. Moreover, by using the resulting tools, the LDB authoring is a friendly simple task, and the inserted information has to accomplish certain constraints (consistency, non recurrence, ...) controlled by the system, helping the authoring process (avoiding violation of hard constraints and reporting the violation of soft constraints). Besides, the integration of diverse LDB built with these tools is assured by the migration tools developed for this purpose. In addition, the resulting dictionaries are friendly usable and supply very useful semantic information to the reader.

As a continuation of this work, we foresee a very promising R&D line, which consists of, among others:

– Refining the design and development methodology from the current state, in order to take into account other possible structures of the taxonomy (for instance, graph-shaped classifications), providing to the ontology with support for explicit generalized relationships, and admitting more linguistic information in the terms of the lexicons.
– Developing new information systems according to the required characteristics of the LDB to come in the future.
– Studying the application of the methodology to the integration of heterogeneous LDB, interoperability among them, and so on.
– Building LDB structurally prepared to be easily embedded in NLP applications.
– Applying the tools to formal and informal Education with the aim of building individual or community dictionaries.

Acknowledgments This work has been partially supported by the Spanish CICYT project number DPI2002-02924.

References

[ABB 02] Atkins S., Bel N., Bertagna F., Bouillon P., Calzolari N., Fellbaum C., Grishman R., Lenci A., MacLeod C., Palmer M., ThurmairR., Villegas M., Zampolli A. (2002) "From Resources to Applications. Designing TheMultilingual ISLE Lexical Entry". In Proceedings of LREC 2002, Las Palmas, Canary Islands, Spain.

[COW 99] A. P. Cowie “English Dictionaries for Foreign Learners. A History”. Oxford. Clarendon Press, 1999.

[EWN] http://www.uva.nl/EuroWordNet.html

[FAR 04] D. Farwell “Intermediate Representation”. Seventh Interlingua Workshop AMTA’04: Determining Interlingua Utility for Machine translation. Washington, DC, October, 2004.

[GW 00] N. Guarino and C. Welty, “Ontological Analysis of Taxonomic Relationships”, Proc. of ER-2000: The International Conference on Conceptual Modeling. LNCS, October 2002.

[KAT 93] B. Katzenberg and P. Piela, "Work Language Analysis and the Naming Problem", Communications of the ACM, Vol. 36, No. 4, June 1993.

[LEN 90] D.B. Lenat, and R.V. Guha, “Building Large Knowledge-Based Systems”, Reading, Massachussets, Addison-Wesley, 1990.

[LREC 02] Workshop on "International Standards of Terminology and Language Resources Management", Las Palmas de Gran Canaria, June, 2002.

[MAH 95] K. Mahesh, and S. Nirenburg, « A situated ontology for practical NLP ». IJCAI’95. Montreal, August 19-21.

[MAN 03] M. Mangeot-Lerebours, G. Sérasset, M. Lafourcade. « Construction collaborative d’une base lexicale multilingue. Le projet Papillon ». TAL, Vol. 44 – 2. 2003

[MCG 00] Deborah L. McGuinness. "Conceptual Modeling for Distributed Ontology Environments," In the Pro-ceedings of The Eighth International Conference on Conceptual Structures Logical, Linguistic, and Computational Issues (ICCS 2000), Darmstadt, Germany, August 14-18, 2000.

[MIK] MikroKosmos, http://crl.nmsu.edu/Research/Projects/mikro/index.html

[MIL 95] G. Miller, “WordNet: A Lexical Data Base for English”, Communications of the ACM, Vol. 38, 11, 1995.

[MOR 02] A. Moreno, and C. Pérez, “Reusing the Mikrokosmos Ontology for Concept-based Multilingual Terminology Databases”, Proceedings of LREC2000, 2002.

[NIR 95] S. Nirenburg, V. Raskin, and B. Onyshkevich, “Apologiae Ontologiae”, Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation, Center for Computational Linguistics, Catholic University, Leuven, Belgium, pp. 106-114, 1995.

[NIR] S.Nirenburg and V.Raskin, “Ontological Semantics.” In crl.nmsu.edu/Staff.pages/Technical/sergei/book.html.

[ONT] http://www.ontology.org/main/papers/iccs-dlm.html

[POL 03] A. Polguère, « Etiquetage sémantique des lexies dans la base de données DiCo ». TAL, Vol. 4 – 2. 2003.

[PRI 01] U. Priss, « Ontologies and Context ». Midwest Artificial Intelligence And Cognitive Science Conference. Oxford, OH, USA, 2001

[RK 02] C. Raguenaud and J. Kennedy, “ Multiple Overlapping Classifications: Issues and Solutions”. 14th International Conference on Scientific and Statistical Database Management (SSDBM'02). Edingburgh, Scotland, 2002.

[SV 02] Sáenz, F. & Vaquero, A. “Towards a Development Methodology for managing Linguistic Knowledge Bases”. Proceedings ES’2002. Springer-Verlag, 2002. pp 453 – 466.

[SKS 02] A. Silberschatz, H.F. Korth, S. Sudarshan, "Database System Concepts", WCB/McGraw-Hill, 2002.

[USC 96] M. Uschold and M. Gruninger, “Ontologies: principles, methods, and applications”. Knowledge Engineering Review, Vol. 11, 2. 1996, pp 93-155.

[VIE 98] E. Viegas, “Multilingual Computational Semantic Lexicons in Action: The WYSINNWYG Approach to NLP”. Int. Conference on Computational Linguistics, ACL. Montreal, 1998.

[WIL 90] Y.A. Wilks, D.C. Fass, C.M. Guo, J.E. McDonald, T. Plate, and B.M.Slator, “Providing machine tractable dictionary tools”. Machine Translation, 5, 1990, pp. 99-151.

[WIL 96] Y. Wilks, B. M. Slator, and L.M. Guthrie, "Electric words: Dictionaries, Computers and Meanings". MIT Press. Cambridge, 1996.

[ZOC 03] M. Zock and J. Carroll “Les dictionaires électroniques”. TAL, Vol. 44, 2. 2003.

Author Index

Índice de Autores

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Febrero de 2005.
Printing 500 / Edición 500 ejemplares.