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

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^h20 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 (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).

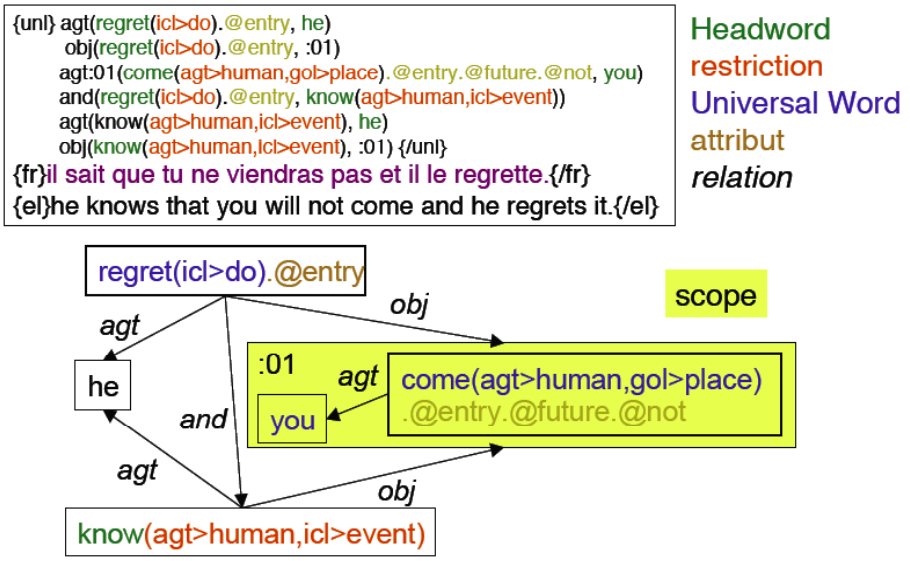

Fig. 3: Example of a UNL graph

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.

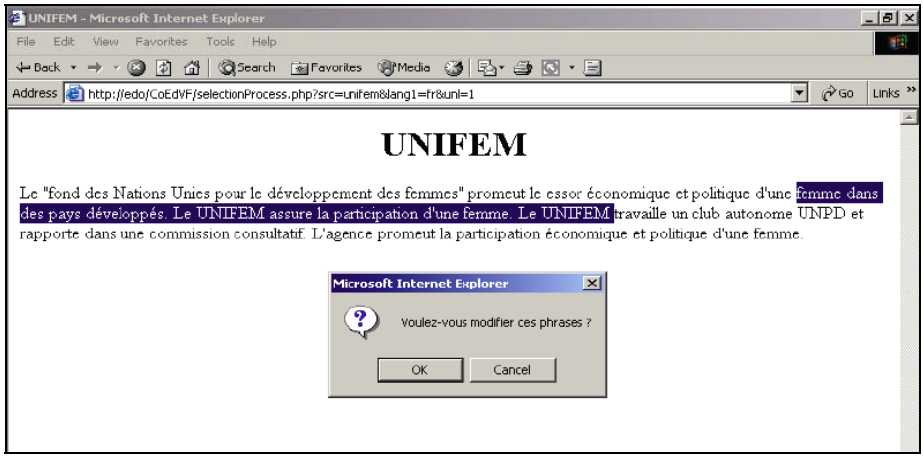

Fig. 4: Reading information "roughly" translated" in a web browser

S/he selects a portion intersecting with these sentences, and chooses the "coedition" menu item. Thanks to 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.

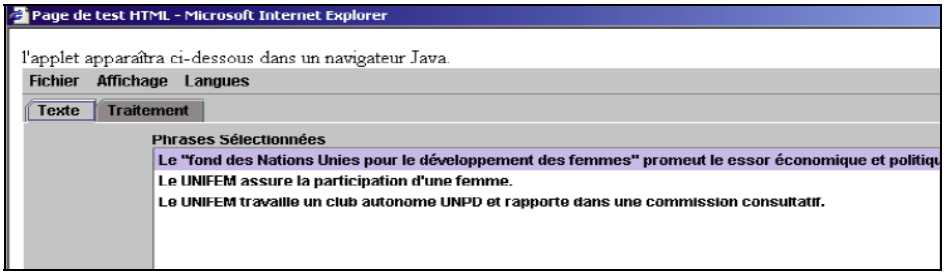

Fig. 5: Sentences determined by the selection appear in a java window

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 vol-

unteer 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.

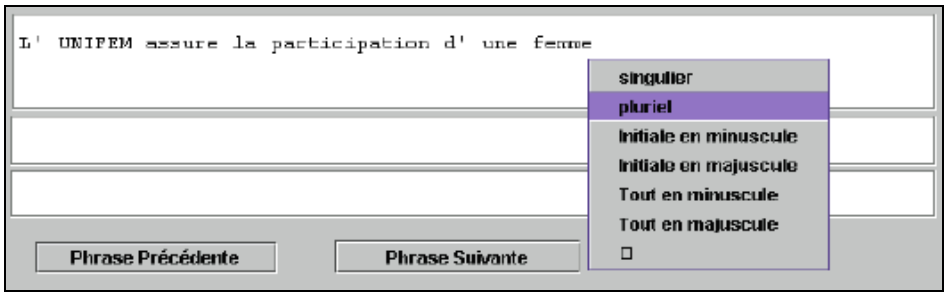

Fig. 6: Possible corrections are proposed

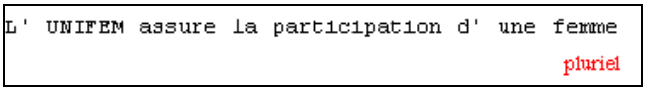

Fig. 7: What the user has asked is shown as an annotion

Finally, the user calls the deconverter(s).

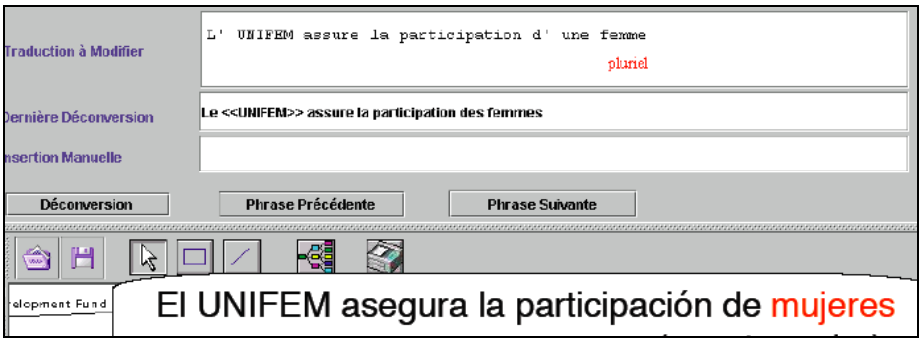

Fig. 8: Deconversion in two languages after coedition in one language

.*@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.