Evaluation of Named Entity Extraction Systems

Mónica Marrero, Sonia Sánchez-Cuadrado,
Jorge Morato Lara and George Andreadakis

Computer Engineering Department, University Carlos III of Madrid
Av. de la Universidad 30, 28911 Leganés (Madrid), Spain
mmarrero@inf.uc3m.es, ssanchec@ie.inf.uc3m.es, jmorato@inf.uc3m.es, gand@ie.inf.uc3m.es

Abstract. The suitability of the algorithms for recognition and classification of entities (NERC) is evaluated through competitions such as MUC, CONLL or ACE. In general, these competitions are limited to the recognition of predefined entity types in certain languages. In addition, the evaluation of free applications and commercial systems that do not attend the competitions has been lightly studied. Shallowly studied have also been the causes of erroneous results. In this study a set of NERC tools are assessed. The assessment of the tools has consisted of: 1) the elaboration of a test corpus with typical and marginal types of entities; 2) the elaboration of a brief technical specification for the tools evaluated; 3) the assessment of the quality of the tools for the developed corpus by means of precision-recall ratios; 4) the analysis of the most frequent errors. The sufficiency of the technical characteristics of the tools and their evaluation ratios, presents an objective perspective of the quality and the effectiveness of the recognition and classification techniques of each tool. Thus, the study complements the information provided by other competitions and aids the choice or the design of more suitable NER tools for a specific project.

Keywords: Named entity extraction, named entity recognition and classification, information extraction, named entity extraction tools.

1 Introduction

There is currently a wide variety of named entities (NE) recognition systems. Competitive events are organized for the evaluation of NERC systems, in which the ability of identification and classification of the entities existing in a corpus is analyzed. Nevertheless, the competitions normally establish certain limitations such as:

– They focus on a limited group of NE types. This feature is quite variable due to the ambiguity in the use of the term *Named Entity* depending on the different forums or events. In the case of the MUC conferences, NEs were considered *personal names*, *organizations*, *locations* and at a later stage, *temporal entities* and *measurements* [1]. On the other hand, the CONLL-2002/2003 conferences defined the categories *person*, *organism, localization* and *miscellaneous* [2, 3]. The latter (*miscellaneous*), includes proper names of different nature with different categories*:* *gentilics, project names, team names*, etc. Finally, the ACE

Advances in Computational Linguistics.
Research in Computing Science 41, 2009, pp. 47-58
Received 10/11/08
Accepted 09/12/08
Final version 04/02/09

conferences, used categories such as *arms, vehicles* or *facilities* (4). ACE also incorporates temporal expressions, but as an independent task. In addition, as far as the NE typology is concerned, there are at least two hierarchies of entity types: BBN categories [5] and Sekine's extended hierarchy [6]. The proposed hierarchical structures fluctuate respectively between 64 and 200 types and subtypes.

- The underlying concept behind every entity varies amongst the different competitions. For example, the entity type *person* includes different subtypes depending on the competition (e.g. titles of person, personal pronouns etc.). These differences along with the mismatch of entity types analyzed in every competition impede their comparison.
- Conferences normally focus on specific languages. Frequently, the languages present in a conference vary from year to year. Currently, ACE is the most competitive and prestigious event that evaluates the NERC tasks. At the same time, it is the most ample as far as the idiomatic coverage is concerned (Arabic, Chinese, English and Spanish).
- The usability and technical characteristics of the software tools is not a factor considered in the competitions.
- The evaluation algorithms are different in each competition. Generally, the precision-recall ratios for the identification and classification of all the criteria considered in the competition are presented in a single measurement. Evaluation varies from the simplicity of CONLL to the complexity of ACE. In CONLL, partial identifications are not considered and false positive errors are not penalized. ACE evaluation [4] is based on a complex algorithm where different named entities are weighted with different weights, making difficult the interpretation and the comparison of results to those of other competitions [7].
- Results obtained in these competitions are conditioned to the manually tagged training and test corpora, which are provided to the participants.
- Large tagged corpora may favor tools that possess larger gazetteers but nevertheless, this does not imply a superior tool quality.
- Tools presented are not necessarily available commercially or for research.
- Various research groups and commercial systems are presented in these competitions establishing a ranking of tools. Thus, the evaluation of NERC systems is limited to those that participate in these competitions. With the current resources, the comparison with tools that fail to attend in these events seems to be impossible.

This article proposes a framework that permits the assessment of NER systems. It intends to provide an evaluation system to those applications which for one reason or another do not attend official competitions. Even in the cases of tools that do participate, this analysis will permit obtaining a complementary vision of their results.

2 Analysis and Methodology

2.1 Characteristics of the Evaluated NERC Tools

There are many operating tools that have been located through references in scientific work or commercial documentation. However, for this study we have defined the following criteria for selecting a NERC system.

1. The system has to permit the processing of texts which are not domain dependent
2. It has to work independently. In other words, it shouldn't require the user to provide resources necessary for its operation.
3. It should process texts in a common language, since language dependency limits the applied techniques. English has been the selected language for this assessment, because it is widely used and supported by the tools.

Tools, such as Trifeed [8], have been discarded for not fulfilling the necessary requirements, as it only accepts predetermined newspaper articles. Other popular tools of the biomedical domain such as AbGene [9], Abner [10] and BioNer [11] have also been discarded. In other cases, some tools have been eliminated due to their reduced efficiency or lack of maintenance. Finally, a couple of tools with good results in the competitions were not considered since they did not dispose a free version: EROCS by IBM and the NERC system by BBM technologies.

Consequently, NERC evaluation will be performed on the following tools: Supersense – Model CONLL, Supersence – Model WNSS, Supersense – Model WSI, Afner, Annie, Freeling, TextPro, YooName, ClearForest and Lingpipe. Freeling is considered as a NERC system, but in the case of the English language it does not perform classification. It has been included because, according to its characteristics and previous evaluations in recognition, it has been giving out moderate results.

- *LingPipe* [12] is a set of Java libraries developed by *Alias-I* for natural language processing. It is by default prepared for the detection and the classification of NEs such as persons, organizations and locations in the English language, but it is also possible to train it through a corpus for other languages. Additionally to the detection and the identification of entities, it is also offering additional functionalities such as orthographic correction and text classification in English. It offers a user interface and various demos through which it is possible to test texts. It is open-source and free of charge for research causes, but it is possible to purchase it for commercial use.
- *ClearForest SWS* [13] is a commercial tool made by *ClearForest Ltd*., currently acquired by *Reuters*. It allows the analysis of English texts and the identification of *ENAMEX* types, in addition to some other types such as products, currencies, etc. A web service, partially based on this tool, has been made for the capture of entities: *Gnosis*, a free plug-in based on this tool for the *Mozilla Firefox* browser, captures numerous types of different entities in web pages. They also offer a Web API that may be used freely under certain conditions. Currently, it has evolved to a tool called *Calais*, which amongst other additional services it

permits the establishment of relationships amongst entities and the detection of events and roles.

- *Annie* [14] is an entity extraction module incorporated in the *GATE* framework. It is open-source and under a GNU license, developed at the University of Sheffield. It is implemented in Java and incorporates in the form of plug-ins and libraries its own or external resources for a variety of aspects related to natural language processing (i.e. Lucene, MinorThird, Google, Weka etc.). It can be used as an API but it also provides its own interface for an independent use. Annie also offers as a module a set of default resources (i.e. tokenizer, sentence splitter, POS tagging, co-reference resolution, gazetteers, etc.) that can be used in combination for the capture of entities. This set can be substituted by other plug-ins or even be disabled. The evaluation of the tool has been realized using its default resources, which are adapted for the English language.
- *Freeling* [15] is a tool developed in C++ at the *TALP Research Center* of the Polytechnic University of Catalonia. It is an open source tool with GNU license that may be used as an API or independently. There is also a Web demo where you can type text. It offers various services related to natural language processing, amongst which the detection of entities. It supports English, Spanish, Catalan, Galician, and Italian. The tool recognizes the usual entities of person, organisms and locations as well as quantities of various types and dates. It separates the identification activities to those of classification, and utilizes automatic learning as well as linguistic (dictionaries, Word-net, lists) and heuristic resources.
- *Afner* [16] is an open-source NERC tool, under GNU license, developed in C++ at the University of Macquaire. Currently it is used as part of a Question Answering tool called *AnswerFinder*, which is focusing to maximizing recall. Afner can also be used as an API for other applications or can be used independently. It uses lists, regular expressions and a supervised learning model which amongst other features, can report the entity's membership to a list or the entity's match with a regular expression. It also allows the addition of lists and regular expressions, as well as the training of new models. It is by default capable of recognizing persons' names, organizations, locations, miscellanea, monetary quantities, and dates in English texts.
- *Supersense Tagger* [17] is an open-source tagger developed in C++ with a version 2.0 Apache license. It is designed for the semantic tagging of nouns and verbs based on WordNet categories which include persons, organizations, locations, temporal expressions and quantities. It is based on automatic learning, offering three different models for application: CONLL, WSJ and WNSS. Given the differences in the tagging and the behavior amongst these three models, they have been considered independently in this study.
- *TextPro* tools suite [18] is developed in C++ at the *Center for Scientific Research and Technology (ITC-irst)*, in Trento, and offers various NLP functionalities interconnected in a pipeline order. It is under a GNU license and uses automatic learning and gazetteers. It is available for English and Italian and offers a web demo for both these languages.
- *YooName* [19] is a tool developed at the University of Ottawa by David Nadeau. It incorporates semi-supervised learning techniques applied to the web, that

permit the identification of entities using a predefined classification of nine types of NEs (person, organization, location, miscellanea, facility, product, event, natural element and unit) and 100 subtypes. There is a web version for doing demos where you can also type English texts in order to be analyzed. The tool also offers a blog with news and information related to its operation and other NER subjects (http://yooname.wordpress.com/).

The main characteristics of each tool are presented on Table 1. As can be observed, the majority of these are developed in C++, offering a console user interface and an API. With respect to the degree of computer usage dexterity that is needed in order to operate each tool, the majority of them have been classified as Advanced and just one of them as Simple (Simple, indicates that it is enough downloading and executing the respective file, and Advanced refers to a more complex process -i.e. additional libraries, compilations, expert configurations etc-). The dash (-) indicates that there was not any information available.

Table 1. Tool features

Tool	Develop. Language	Interface	License	Simple (S) /Advanced (A) Installation	Demo	Entity types
Supersense-CONLL	C++	Console/API	Apache 2.0	A	No	4
Supersense-WNSS	C++	Console/API	Apache 2.0	A	No	27
Supersense-WSJ	C++	Console/API	Apache 2.0	A	No	> 100
Afner	C++	Console/API	GNU	A	No	6
Annie	Java	Graphical /API	GNU	A	Yes	~12
Freeling	C++	Graphical /API	GNU	A	Yes	0
TextPro	C++	Console/API	GNU	A	Yes	4
YooName	-	-	-	-	Yes	>100
ClearForest	-	Web/API	Commerc.	-	Yes	6
Lingpipe	Java	API	Free/Devel op./Startup	S	Yes	3

2.2 Methodology

The data analysis has been realized having a triple focus:

– Comparison of the tools' characteristics: task realized through a brief technical specification based on usability aspects.

- Comparison of results obtained by the tools, for entities found in the test corpus. This evaluation has been realized through distinct measures of precision – recall based on :
    - Identification of the entities and false positives in the identification
    - Classification of entities
    - Classification by NE types that each tool recognizes.
- Comparison of the tools according to the typographic, lexical, semantic or heuristic factors that has been considered in the entities recognition. This analysis has been realized with data mining classification algorithms. For doing this, information referring to all the nominal elements (entities or not) of the corpus has been introduced into the Weka [20] tool and analyzed with the PART algorithm to extract rules reflecting the behavior of each tool.

In particular, the typographic, lexical, semantic and heuristic features analyzed in the entity recognition processes are:

- Words at the first position of the phrase.
- Words written with the first letter in uppercase.
- Words in quotes.
- Words written totally in uppercase.
- Words written totally in lowercase.
- Polysemic words.
- Noun phrases
- Entities previously identified/classified in the text
- Possible use of:
    - Verb argument (based on semantic roles)
    - Trigger word based recognition
    - Gazetteer based recognition
    - Regular expressions

An English test corpus has been made containing all the above features in order to evaluate the behavior of the tools. It has a total of 579 words, distributed in 13 paragraphs in which more than 100 occurrences of various types of entities have been accumulated. Some of these NEs may be recognized and classified using gazetteers (e.g. Spain), and some others may be recognizable through trigger words (e.g. Inc., Co., Mr.). These entity types were distributed in various phrases in the corpus with different typography (dash, quotes, etc.), the relative position in different sentences, and the orthographic form (e.g. upper or lower case letters).

Invented NEs (*dontknowhere, dontknowho*) and polysemic entities (e.g. *Rose*) allows the verification of the use of NLP techniques. On the other hand, the recognition of fictional entities in lower case and with no special features or contextual information that could assist in their identification, shows the influence of pre-processing stages on the tools.

Finally, it must be taken in account that entities in a tool could neither totally coincide in number nor in semantic with their equivalent entities in other tools so the analysis has to be specialized for every tool (It's the corpus that should be adapted to the tools and not the tools to the corpus).

3 Results

3.1 General Results

The results obtained for each of the parameters considered in the evaluation are presented next. The charts of precision-recall for both identification (Fig. 1) and classification (Fig.2), present a performance which is generally over 50%. Exceptions are the precision and recall values of the Afner tool, and the recall values of the YooName tool. ClearForest stands out with its behavior for obtaining precision rates that exceed 90%. Other tools such as Supersense Tagger and Annie achieve inferior values, although they exceed 70% and seem to be more equilibrated in respect to their recall.

A detailed analysis should additionally take in account the false positive errors, i.e. the elements erroneously identified as entities, as this could result more damaging in a project than partial identification or erroneous classification. Therefore, the tools that obtain a greater number of false positive errors are Freeling and Annie, whilst WNSS model of SupersenseTagger does not identify erroneously any element as an entity.

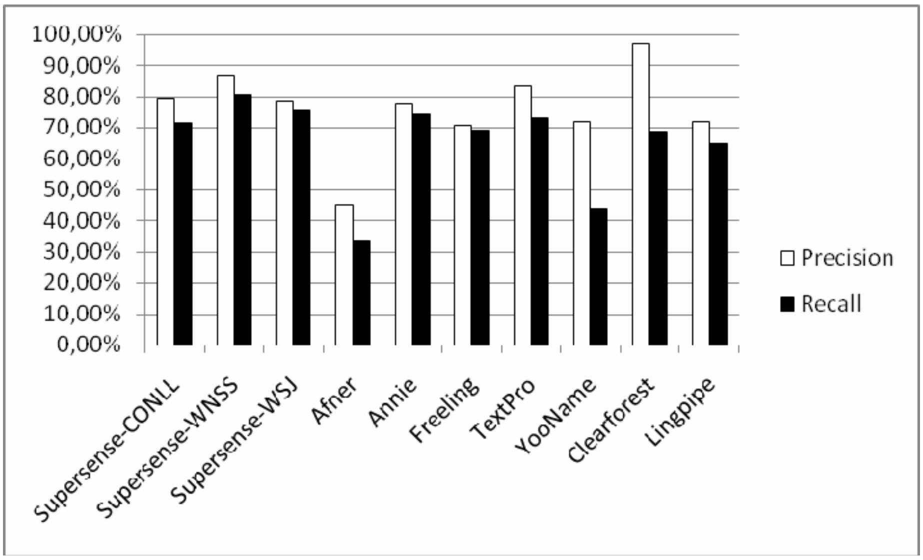

Fig. 1. Precision-Recall in entity identification

Given that classification is a process that depends on the identification of entities, the f-measure in identification is always superior to that of the classification's (Fig. 3). However it is generally observed that the values are similar. The most notable differences appear with the TextPro tool and to a lesser degree, with the WSJ Model of SupersenseTagger, which stand out in their identification processes but not in the classification of entities that have previously managed to identify.

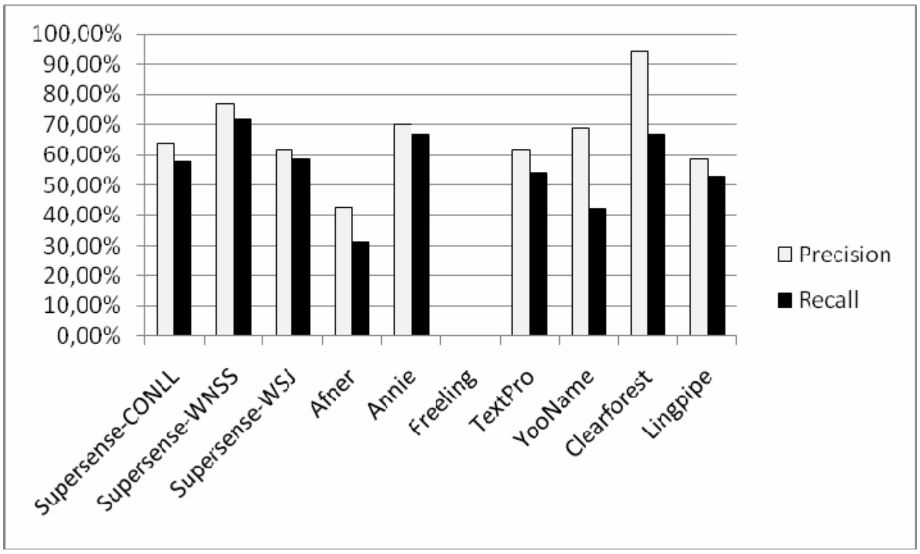

Fig. 2. Precision-Recall in entity classification (Freeling has not been evaluated in this process)

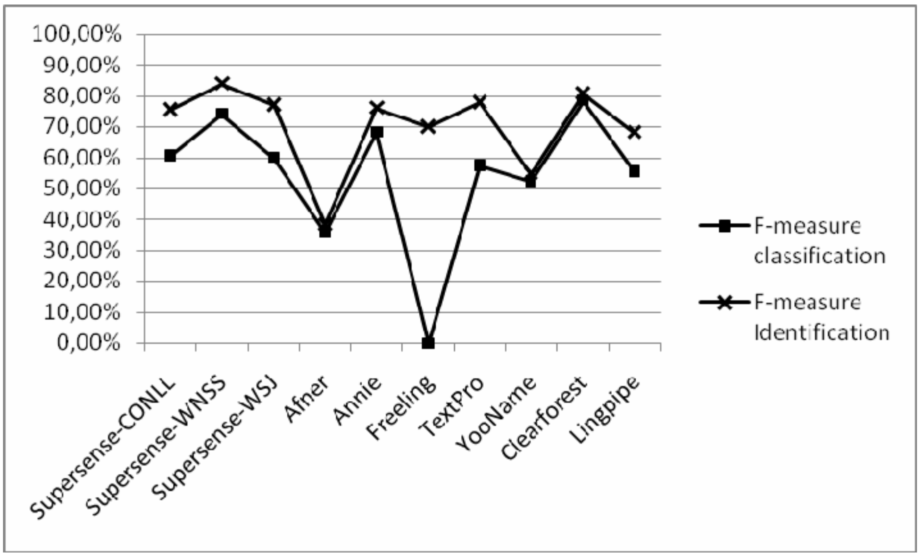

Fig. 3: F-measure in entity identification and classification (Freeling has not been evaluated in classification).

3.2 Results by Entity Type

The number of categories that each tool can recognize (Table 2) is an important factor for the evaluation of a tool. It is quite different having a tool able to recognize over one hundred different types of entities, to having a tool that can only recognize three.

However, the utility and difficulty of recognition of some types against some others is different, which demonstrates the need for a study based on the entity's types. In this case the study was carried out for each one of the entity types that the tool was able to recognize in the corpus. Thus, given the ambiguity relative to the term *entity* [21] and the lack of uniform use of tags, we have to previously analyze the precise significance of each tag in each tool and make them uniform.

The analysis illustrated in Table 2 allows us to observe some differences to the global analysis. Afner, which initially had worst results that the other tools, is performing better on person recognition than Supersense-WSJ or YooName. Additionally it is remarkable how YooName has an f-measure on the entity type *Company* of 0.08, whilst ClearForest achieves 0.95.

Table 2. Results by entity type

Tool	Entity type	N	F	Tool	Entity type	N	F
Supers. CONLL	Person	32	0'63	YooName	Person	33	0'30
	Location	43	0'64		Location	44	0'51
	Org.	13	0'72		Org.	13	0'88
	Miscelanea	4	0		Vocation	4	1
Supersense-WNSS	Person	38	0'65		Country	21	0'66
	Location	48	0'78		State/Prov.	4	0'75
	Group	25	0'88		City	7	0'70
	Time	9	0'66		Loc (other)	11	0
	Quantity	9	0		Company	12	0'08
	Food	6	1		Month	2	1
	Communicat.	1	1		Week Day	2	1
	Cognition	2	0'66		Food	6	1
	Substance	1	0		Mineral	1	0
	Relation	1	0		Vegetal	1	1
	Plant	6	1	Clear Forest	Person	53	0'72
	Object	1	1		Country	19	0'97
	Other	1	1		State/Prov.	6	1
Supersense-WSJ	Person	32	0'30		City	10	0'18
	Person-Desc.	5	1		Company	12	0'95
	Geo-Pol.(other)	20	0'10	Text Pro	Person	32	0'59
	Country	18	0'70		Location	44	0'51
	State/Province	6	0'66		Org.	13	0'88
	Geo-Pol-Desc.	9	1	Lin gpipe	Person	32	0'67
	Corporation	12	0'69		Location	47	0'47
	Org.-Descrip.	12	0'86		Org.	12	0'78
	Date	10	1	Afner	Person	32	0'50
	Money	4	0'28		Location	44	0'36
	Food	7	1		Org.	12	0
	Ordinal	1	1		Date	2	0'50
	Cardinal	7	0'92				

3.3 Inference

With the aid of Weka, inferences have been made about the behavior of the tools. The typographic, lexical, semantic and even contextual characteristics of every corpus element susceptible into been captured as an entity, have been annotated. Using an automatic learning algorithm (PART) applied to the results of each tool we have obtained rules that characterize the behavior of the tools in the combined task of identification and classification. Finally, we have applied this algorithm to the aggregate of results of all tools in order to detect common behavioral patterns. Those rules demonstrate the features most involved in the errors obtained by each tool during the processes of identification and classification.

One of the most important features seems to be the orthographic form of the entities: Supersense-CONLL, Supersense-WNSS, Afner, Annie and Freeling have remarkable problems in the recognition of entities written in lowercase, and Supersense-WSJ, Afner, Annie and Freeling have a significant number of false positives with words written totally in uppercase.

On the other hand, the existence of noun-phrase entities influences the errors committed by many tools: Supersense-WNSS, Afner, Annie, YooName and LingPipe have problems in the recognition of noun-phrase entities mainly when the typography or orthographic form of the terms in the noun-phrase, are different. The triggers work fine in all the tools except for Supersense-CONLL, which which does not seem to handle them well. Finally, the existence of polysemic entities is a handicap for all tools, but the rules make this handicap to stand out in the case of ClearForest. This does not necessarily mean that ClearForest performs worst with polysemic entities, but yet it is the only noticeable problem that this tool has.

4 Conclusion

An analysis of various NERC tools has been presented in this study. The evaluation proposes a model that eliminates some of the competitions' limitations into assessing these tools. This model is based on the creation of a small corpus, and the adaptation of the evaluation methodology to the NERC typology of the tools, not the contrary as it is common in the major competitions. The analysis of all the identified entities and the errors committed during this process permits a study using data mining in order to discover the most frequent errors in the identification and classification of NEs.

All the evaluated tools are oriented to experts who may integrate them in other systems. The major programming languages utilized are C++ and Java. The election of these languages could be related to their efficiency, portability, or their abundance in libraries.

At first sight as far as the performance of the recognition of entities is concerned, the NERC tools that performed best were Supersense-WNSS and Clearforest. It can be observed that the variety of entity types that the tools can recognize does not determine the results: tools that recognize the largest number of entities, such as Supersense-WSJ or YooName, do not achieve very good results; on the other hand, the lowest ratios are achieved by Afner, which recognizes a few different entity types.

In other words, an important factor in the evaluation of the different systems is not only the number of different entity types recognized but also their "quality". Metrics presenting the average performance in the identification of entity types is not always representative of its success. The performance of every tool in the identification of individual entity types should be examined in order to extract better conclusions.

The errors committed by all tools have been analyzed using data mining in order to determine which could be their cause and identify common patterns. This information was rarely analyzed in the competitions. The most common difficulties and the deficiencies detected in NERCs denote a handicap in the management of noun phrases and reveal a strong dependency on gazetteers. Tools that focus on gazetteers (as in the case of Afner and YooName) seem to produce poor results. This deficiency seems to be due to the scarce importance given to context analysis. Another deficiency is the lack of a preprocessing stage during which the tools could acquire knowledge useful in the tagging of ambiguous entities. This may lead to the failure of identifying an entity that previously has been successfully recognized (TextPro, LingPipe). An exception to this was YooName, although in this case, if the typography of the same entity through the corpus is different, this tool can conclude that it is not an entity.

NERC systems present an elevated dependency to uppercase characters, not being able to recognize the same entity if written in lowercase, even though it does with other typographic elements such as quotes. The management of dashes (-) and full stops (.) can significantly influence the recognition process, separating parts of a multi-word entity or uniting terms of different entities even if those are located in different paragraphs and are separated by full stops.

Moreover, the results point to semantic problems such as the inability of the NERCs to recognize polysemic entities and the inconsistency in the detection of cardinal or ordinal types, which are only recognized when they are written numerically but not when written alphabetically.

The techniques that have given the best results in the experiment have been the consideration of linguistic information (in the case of Supersense-WNSS), and the triggers (in the case of Supersense-CONLL). On the other hand, being limited to typography and gazetteers does not seem to improve results (Afner). The identification seems to be based mostly on the capitalization of words (Freeling, Annie).

Acknowledgements. This work was carried out within the framework of a research project (Ref. CCG07-UC3M/TIC-3401) financed by the Comunidad Autonoma de Madrid (CAM) and University Carlos III of Madrid (UC3M). We acknowledge in particular Raúl Matallanos to realize previous studies and test about evaluated systems.

References

1. Grishman, R., Sundheim, B.: Message Understanding Conference-6: A Brief History. In: Proc. 16th Conference on Computational Linguistics, vol. 1, pp. 466—471. ACL: NJ (USA) (1996)

2. Tjong Kim Sang, Erik. F. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In: Proc. Conference on Natural Language Learning, pp. 155--158. Taipei, Taiwan (2002)
3. Tjong Kim Sang, Erik. F.; De Meulder, F. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proc. Conference on Natural Language Learning. pp. 142-147. Edmonton, Canada (2003).
4. NIST: ACE08 Evaluation Plan. http://www.nist.gov/speech/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf (2008)
5. Brunstein, A. Annotation Guidelines for Answer Types" BBN technologies (2002). http://www.ldc.upenn.edu/Catalog/docs/LDC2005T33/BBN-Types-Subtypes.html
6. Sekine, S., Sudo, K., Nobata, C.: Extended named entity hierarchy. In: Proc. of the Third International Conference on Language Resources and Evaluation (LREC-2002), pp. 1818--1824. Las Palmas (SP) (2002)
7. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Linguisticae Investigationes, vol 30, n.1, pp. 3--26 (2007)
8. Trifeed http://www.trifeed.com/
9. AbGene ftp://ftp.ncbi.nlm.nih.gov/pub/tanabe/AbGene/
10. Abner http://pages.cs.wisc.edu/~bsettles/abner/
11. BioNer. POSBioTm Postech Biological Text-Mining system http://isoft.postech.ac.kr/Research/BioNER/POSBIOTM/NER/main.html
12. Baldwin, B., Carpenter, B.: LingPipe. http://www.alias-i.com/lingpipe/
13. ClearForest Semantic Web Services (SWS), http://sws.clearforest.com/
14. Cunningham, H., Maynard D., Tab-lan V., Ursu C., Bontcheva K.: Developing Language Processing Components with GATE. GATE v5 User Guide, http://www.gate.ac.uk/sale/tao/tao.pdf
15. Atserias J., Casas B., Comelles E., González M., Padró Ll., Padró M.: FreeLing 1.3: Syntactic and semantic services in an open-source NLP library. In: Proceedings of the fifth international conference on Language Resources and Evaluation. ELRA , Paris (2006).
16. Zaanen M., Molla D.: A named entity recogniser for question answering. In: Proceedings PACLING (2007)
17. Ciaramita, M., Altun, Y.: Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In: Proceedings of EMNLP-06, pp. 594--602, Sydney, Australia, 2006
18. Pianta, E., Girardi, C. and Zanoli, R.: The TextPro tool suite, Proceedings of Sixth International Language Resources and Evaluation. ELRA, Paris (2008)
19. Nadeau, D., Turney, P.; Matwin, S.: Unsupervised Named Entity Recognition: Generating Gazetteers and Resolving Ambiguity. In Proc. Canadian Conference on Artificial Intelligence (2006)
20. Witten, I.H., Franck, E., Trigg, L., Hall, M., Holmes G., Cunningham S.J.: Weka: Practical machine learning tools and techniques with Java implementations, Proc. ANNES'99 International Workshop on emerging Engineering and Connectionnist-based Information Systems, pp. 192–196 (1999).
21. Borrega, O, Taulé, Mariona and Martí, Mª. Antonia. What do we mean when we speak about Named Entities? Proceedings of the Corpus Linguistics Conference, CL2007. University of Birmingham (2007)