Ontology-based interoperation of linguistic tools for an improved lemma annotation in Spanish

Antonio Pareja-Lora[*], Guadalupe Aguado de Cea[**]
[*] Dept. Sistemas Informáticos y Computación (DSIC). Universidad Complutense de Madrid (UCM)
Prof. José García Santesmases, s/n. 28040 – Madrid (Spain)
[**] Ontology Engineering Group. Universidad Politécnica de Madrid (UPM)
Campus de Montegancedo, s/n. 28660 – Boadilla del Monte (Spain)
E-mail: apareja@sip.ucm.es, lupe@fi.upm.es

Abstract

In this paper, we present an ontology-based methodology and architecture for the comparison, assessment, combination (and, to some extent, also contrastive evaluation) of the results of different linguistic tools. More specifically, we describe an experiment aiming at the improvement of the correctness of lemma tagging for Spanish. This improvement was achieved by means of the standardisation and combination of the results of three different linguistic annotation tools (Bitext's DataLexica, Connexor's FDG Parser and LACELL's POS tagger), using (1) ontologies, (2) a set of lemma tagging correction rules, determined empirically during the experiment, and (3) W3C standard languages, such as XML, RDF(S) and OWL. As we show in the results of the experiment, the interoperation of these tools by means of ontologies and the correction rules applied in the experiment improved significantly the quality of the resulting lemma tagging (when compared to the separate lemma tagging performed by each of the tools that we made interoperate).

1. Introduction

The appearance of the World Wide Web (or, simply, the Web) and, more recently, of the Semantic Web (Berners-Lee *et al.*, 1999), has made available a huge amount of information, which requires being indexed, retrieved, extracted, translated, summarised and/or analysed. All this processing cannot be done solely by humans. Due to the increasing number of web pages, this processing requires being highly automated.

Corpora annotation and, more generally speaking, the field of linguistic annotation, can help the (Semantic) Web achieve this high degree of automation. Indeed, one of its most important achievements is the creation of some computational tools (POS taggers, linguistic parsers, sense and semantic taggers or, in general, linguistic annotation tools) for the automatic analysis and annotation of texts.

Even though the most pressing need of the (Semantic) Web lies in the semantic level of annotation, POS taggers and linguistic parsers are also very helpful at segmenting and pre-processing texts before they are semantically annotated (Vargas-Vera *et al.*, 2000; Aguado de Cea *et al.*, 2002). Accordingly, POS taggers and linguistic parsers pave the way for semantic and sense taggers to identify the main entities and relationships in a text and annotate them correctly. Hence, it seems more urgent than ever to incorporate all these linguistic annotation tools into the automatic processing of (Semantic) Web texts.

However, although these tools seem to be quite developed for languages such as English, some other widely used languages, such as Spanish, lack the vast number of (freely available) tools already developed for the former. In addition, laboratory tests of these tools yield magnificent results (around 5-10% nominal error rate in POS taggers, for example), but they usually behave much worse in practice (around 20% actual error rate, as it is shown below). These high error rates in POS taggers and linguistic parsers make it more difficult for semantic and sense taggers to achieve a satisfactory level of correctness themselves. Therefore, it is required to reduce the error rate of POS tagged and linguistically parsed texts as much as possible.

The present paper shows the results of an experiment carried out in order to reduce automatically the error rate in POS tagged Spanish texts and, more specifically, to reduce the error rate of lemma annotation for Spanish. This error rate reduction was based on the interoperation, comparison and combination of the results of three different linguistic annotation tools. This was supported by a set of ontologies (Gruber, 1993; Borst, 1997) and a set of rules for lemma combination, empirically determined.

This paper has been structured as follows. First, we present the three linguistic tools used in this experiment and the methodology followed for the combination of their results, respectively, in Section 2 and in Section 3. Second, we give further details about the rules for lemma combination (Section 4) and the ontologies (Section 5) aforementioned. Third, we show the results obtained in the experiment in Section 6. Fourth, we state the conclusions derived from this experiment in Section 7. Finally, the last two sections (*i.e.*, Section 8 and Section 9) include the acknowledgements and the references relating this work.

2. Description of the tools

The first of the three different linguistic annotation tools used in this experiment was **DataLexica**[1]. DataLexica is a non-disambiguating lemma and POS tagger. Its input is a

[1] http://www.bitext.com/ES/datalexica.asp

single word and its output is the set of the possible morphosyntactic analyses of the input word. Its lack of any kind of disambiguation or, more simply, its ambiguity at morphosyntactic annotation is a most prominent drawback of this tool. This ambiguity is shown by the appearance of multiple labels related to the same phenomenon being tagged for a large number of the tokens annotated. However, it is a rather reliable tool for the morphological annotation of Spanish texts, provided that it is used in conjunction with a morphosyntactic disambiguating tool. Besides, (i) its annotations are very accurate, once the right one has been discerned from the spurious ones; and (ii) it is very robust. In fact, it annotates almost all the tokens in any document (however, it does not annotate punctuation signs, for example).

The second tool was **a POS tagger developed by the LACELL research group**[2]. Its input is a plain text ('*.txt') ASCII file and its output is an MS Excel file that contains the lemma and the morphosyntactic annotations of the morphosyntactic units in the input file. LACELL's POS tagger can recognise and correctly tag a large number of input tokens. However, it fails to annotate the type of de-contextualised, foreign or newly-coined words that are pervasive in the Web domain.

Finally, the third tool involved in this experiment was **Connexor's FDG Parser** (Tapanainen & Järvinen, 1997). Its input is also a plain text ('*.txt') ASCII file and its output is either a plain text file or an XML file that contains not only the lemma and the morphosyntactic annotations of the words in the input file, but also an account of the functional dependencies holding between them (that is, a type of dependency parsing of the input). The main contribution of this tool to the experiment was its reliability at the annotation of infrequent tokens, and, more precisely, the types of words that LACELL's POS tagger failed to annotate. However, an important drawback of this tool was the level of ambiguity of its annotations, shown by the attachment of several POS and lemma tags to certain tokens. After checking manually its annotations on a sample corpus, it was observed that around 19% of the tokens had an ambiguous grammatical category and lemma tag (more than one tag for a given token). This level of ambiguity reduced to some extent the reliability of its annotations.

These three tools were licensed to our research group by 2002 (at that time, no freely available –and downloadable– POS tagger could be found for Spanish). As explained, when used separately, they have been yielding some rather poor results at lemma (and POS) tagging. However, they are expensive tools and, therefore, their cost had to be recouped. A first attempt to overcome this problem was to find out whether they could interoperate to improve their separate results. This was assessed by means of the experiment described below.

3. Methodology

The three tools were run on a set of HTML web pages from the domain of the cinema reviews. These pages were previously transformed into plain text files, where all their HTML labels had been removed.

Next, their corresponding output files were fed into an implementation of the architecture described in Aguado de Cea *et al.* (2003), shown in Figure 1, called OntoTagger. This architecture specifies a number of pipelined (sequential) phases that enable the comparison and the combination of different annotations of a certain input text.

First of all, due to the different annotation assumptions, scopes, schemas and formats of the set of tools chosen, a mechanism for the standardisation of their annotations had to be provided. This was achieved by means of a set of XML files containing the mappings of each tool tagset into a sort of standardised tagset. This standardised tagset was included in a group of ontologies (Aguado de Cea *et al.*, 2004a; Aguado de Cea *et al.*, 2004b) that formalise the EAGLES (1996a; 1996b) recommendations for the morphosyntactic and the syntactic annotation of corpora. It also complies with the standards being currently developed within the ISO TC37 SC4 subcommittee[3], such as LAF/GrAF (ISO, 2008a) and MAF (ISO, 2008b).

Second, the morphosyntactic category and the lemma tags of the input words were separated (decanted) from the rest of phenomena being annotated by each tool. Both types of tags were obviously interrelated. Thus, changing or improving the annotation of one of them entailed changing or improving the annotation of the other. That is why they were stuck together in this process of decanting.

Third, a manual checking of the annotations was performed, in order to assess their correctness. Besides, the different outputs were automatically compared, and a disagreement report file was generated for its study. This helped (a) determine the error rate of each tool at lemma tagging; (b) find out the types of tokens that each tool failed to tag (correctly) systematically or most frequently; and (c) develop the rules to correct these types of errors. These rules generally used some contextual information not considered by the tools when annotating or simply selected the lemma tag coming from the tool less likely to yield that type of error in that context.

Fourth, these error correction rules were implemented into a module that was pipelined after the decanting phase described above (these rules are presented in Section 5). The resulting pipeline was applied to the combination of the standardised and decanted annotations of the three tools for another set of HTML web pages of the same input domain. The final output of this pipeline was a unique and improved lemma (and POS) annotation of these pages, implemented into three different W3C[4] standard languages, namely XML, RDF(S) and OWL.

Finally, the resulting lemma annotations (the ones corresponding to the three original tools and the one obtained from their output combination) were manually checked and their correctness was assessed once again. The results of this assessment are shown in Section 6.

[2] http://www.um.es/grupos/grupo-lacell/quees.php

[3] http://www.tc37sc4.org/

[4] http://www.w3.org/

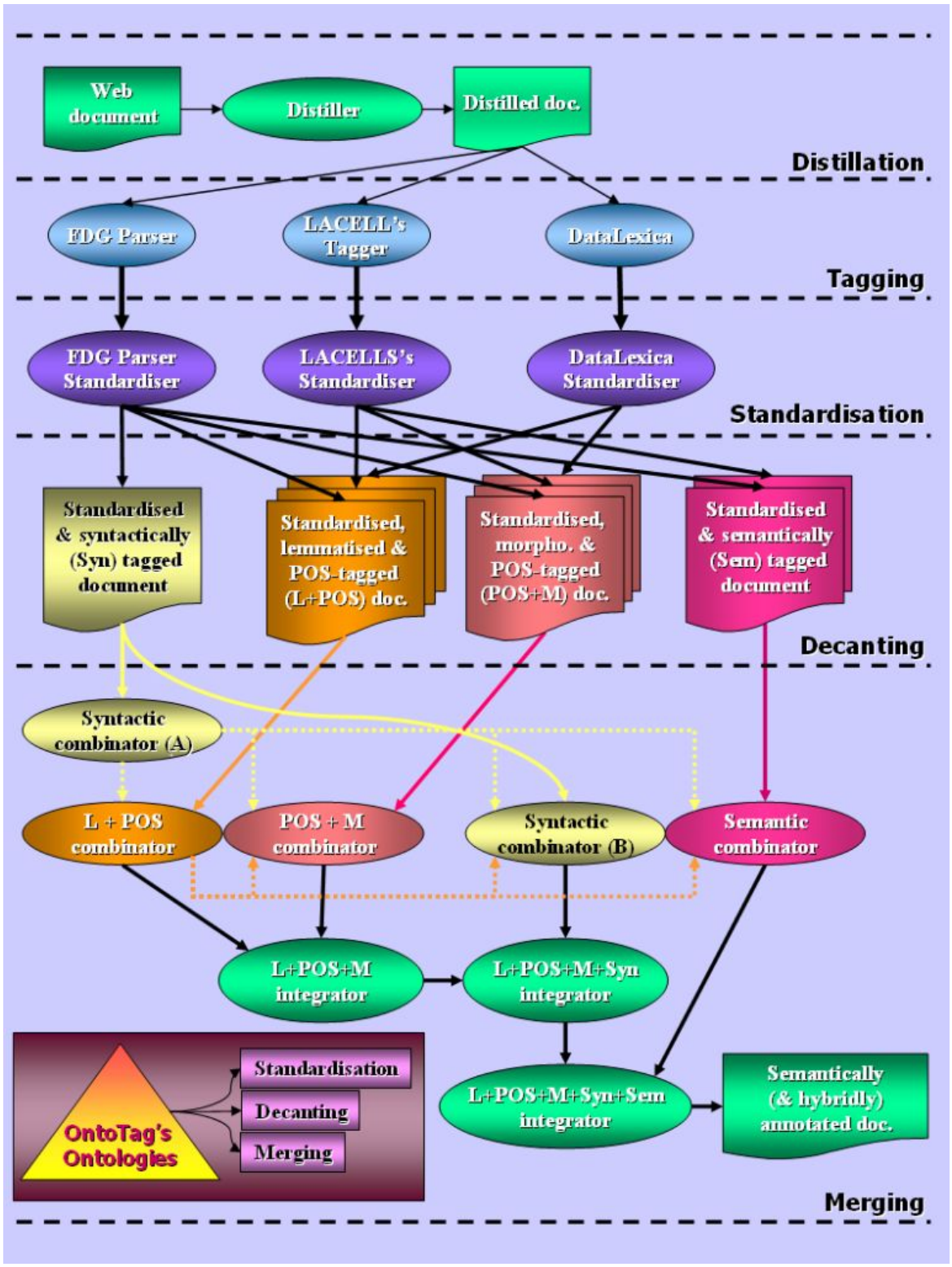

Figure 1: The software configuration used in this experiment

4. The rules for lemma combination

As mentioned above, the rules for lemma combination of OntoTagger were determined empirically, checking and comparing the annotations coming from each linguistic tool incorporated into the configuration. This was done with the help of an output file generated *ad hoc* by OntoTagger, which summarises the discordances between these lemma tags. These rules for lemma combination solve many particular (and sometimes inexplicable) malfunctions of the linguistic annotation tools.

To begin with, a sort of or meta-rule for lemma

combination was identified when studying the output file containing the lemma tagging discordances aforementioned. A specification of this meta-rule is shown in Figure 2.

PROCEDURE *LemmaCombination(*OUT $lemma_1$, IN $lemma_2$*)*
  IF $lemma_2$ contains **only one value** THEN
    $lemma_1$ ← the unique value in $lemma_2$
  ELSE (* $lemma_2$ contains **more than one value** *)
    IF the token has a **verbal** (sub)category tag THEN
      IF *Category_Match*(CAT_2^{COMB}, CAT_2^{FDG}) THEN
        $lemma_1$ ← the (only) value in $lemma_2^{FDG}$
      ELSIF *Category_Match*(CAT_2^{COMB}, CAT_2^{UM}) THEN
        $lemma_1$ ← the (only) value in $lemma_2^{UM}$
      ELSE
        $lemma_1$ ← the first irreflexive lemma in $lemma_2^{DL}$
    ELSE
        $lemma_1$ ← the first value stored in $lemma_2^{DL}$

Figure 2: The general meta-rule for lemma combination applied in the experiment

In this meta-rule, (1) *CAT* denotes a list of triples <*tool*, *token*, *category*>, where *category* is the POS tag assigned by its associated *tool* to the *token* in question. Besides, whereas the (input) parameter $lemma_1$ represents a simple STRING, the (output) parameter $lemma_2$ represents a list of quadruples <*tool*, *token*, *category*, *lemma*>, where each of these quadruples describes one of the (alternative) POS (*category*) and *lemma* tags assigned by a given *tool* to a certain *token*. Lastly, (1) *DL* stands for DataLexica; (2) *FDG* stands for Connexor's FDG Parser; (3) *UM* stands for LACELL's POS tagger; and (4) *COMB* stands for resulting (or combined).

This meta-rule was formulated taking into account that

1. Most frequently, once the morphosyntactic category has been correctly determined, the best choice as for its lemma is the one associated to that morphosyntactic category in the annotations of *DL* (except when it is a verbal (sub)category). In effect, the annotations coming from *DL* are the most accurate ones once a word form and its corresponding morphosyntactic category have been fixed. For this reason, by default, $lemma_2$ is assigned the lemmas coming from *DL*.
2. However, the lemma tagging of *DL* for verbs is highly ambiguous, since it often includes at least two very similar lemmas, namely the one associated to a reflexive use of the verb and the one associated to its non-reflexive use. Accordingly, the lemma tag chosen for a token in this case is the one assigned to it by a tool (either *FDG* or *UM*) that also assigned to it a correct morphosyntactic category (which should be already included in CAT_2^{COMB}).

The remaining rules used for lemma combination in OntoTagger are, in fact, exceptions to the meta-rule shown in Figure 2. A couple of them are presented, respectively, in Table 1 and in Table 2, together with a corresponding example of application. In each of these tables, we have included (1) a natural language description of the RULE being presented; (2) the TOKEN whose lemma is combined using the associated rule, designated by (2.a) the number of file (#FILE), (2.b) the token identifier (#TOKEN), consisting of its paragraph number, its sentence number within its paragraph and its number of token within its sentence, separated by '_'; (3) its WORD FORM in the input document (TEXT); (4) the TOOL (DL = DataLexica; FDG = Connexor's FDG; UM = LACELL's POS tagger) which produced the TOOL CATEGORY and the TOOL LEMMA included in the following two columns; (5) its combined morphosyntactic category tag, obtained previously (COMBINED CATEGORY); (6) the COMBINED LEMMA obtained after the application of the rule; (7) the CONTEXT of the token in the input file where it appears; (8) a TRANSLATION into English of this Spanish context phrase; and (9) some appropriate COMMENTS, when necessary.

5. The role of ontologies in the experiment

As shown in Figure 1, ontologies play a significant role in the standardization, decanting and merging (that is, both the combination and the integration) of the results of the three tools interoperating in this experiment.

As far as standardization is concerned, our ontologies (Aguado de Cea *et al.*, 2004a; Aguado de Cea *et al.*, 2004b) defined the common and standardized vocabulary and/or the tagset into which the different annotations of the three tools were mapped. This was achieved by means of a dedicated mapping process for each of the tools included in the experiment. Each of these mapping processes used an XML file that defined the correspondences between (1) the tool-dependent tags and (2) the tool-independent and standard-compliant (EAGLES, 1996a; 1996b; ISO, 2008b) concepts, attributes, values and instances of the ontologies. Once these mapping processes had been developed, the first step towards the standardization of the different combined annotations had already been taken.

Then, the second step was taken, and the annotations were re-expressed according to a LAF/GrAF-compliant (that is, standardized) annotation scheme (ISO, 2008a), specified in terms of <*subject*, *predicate*, *object*> triples. This, in turn, was achieved by specifying three different schemata, each one based on one of the main Semantic Web and/or ontology-oriented W3C standard languages (that is XML, RDF(S) and OWL). The previous translation of the tool-dependent tags into an ontology-based tagset enabled a straightforward integration of these annotations (as well as of the combined annotations) into the three schemata mentioned above. This completed the ontology-based standardization of the annotations involved in the experiment.

As regards decanting, once the different tags had been mapped into a standardized and ontology-based tagset, the underlying hierarchy of the ontologies supported an easy separation of the annotations according to the type of phenomena they were related to. Hence, ontologies also helped classify the annotations and decant (that is, separate) them according to their linguistic level (morphological, morphosyntactic, syntactic and semantic)

and type (lemma-related, category-related, morphological, etc.).

Finally, as for merging, the use of ontologies contributed significantly to re-expressing all the tags used by the three linguistic tools, in a unified and common tagset (or vocabulary). This enabled an straightforward comparison among them and, hence, also their combination. The hierarchical structure of the ontologies helped refine as much as possible the combined tags, and the final merging was facilitated by the XML, RDF(S) and OWL schemata introduced above.

RULE A		**IF** THE COMBINED CATEGORY (POS) TAG FOR THE TOKEN IS 'RU' (RESIDUAL UNASSIGNED), THAT IS, THE MOST GENERAL AND INACCURATE ONE, **THEN** TAKE , AS ITS COMBINED LEMMA, THE WORD FORM OF THE TOKEN IN QUESTION FROM THE INPUT TEXT						
#FILE	**#TOKEN**	**#WORD**	**TEXT**	**TOOL**	**TOOL CATEGORY**	**TOOL LEMMA**	**COMBINED CATEGORY**	**COMBINED LEMMA**
1	42_2_1	42_2_1_1	Méndez	DL	RU	méndez	RU	Méndez
				FDG	null	null		
				UM	null	null		
CONTEXT		"Crítica de F. **Méndez**-Leite"						
TRANSLATION		"F. **Méndez**-Leite's commentary"						
COMMENTS		Avoids a POS tagging error that could not be solved with the linguistic tools involved.						

Table 1: An example of application of RULE A for lemma combination

RULE B		**IF** THERE ARE AS MANY DIFFERENT CANDIDATE LEMMA TAGS FOR THE TOKEN AS TOOLS **THEN** TAKE, AS ITS COMBINED LEMMA, THE LEMMA OUTPUTTED BY THE FIRST TOOL WHOSE OUTPUT CATEGORY (POS) TAG FOR THE TOKEN IS NOT 'null'						
#FILE	**#TOKEN**	**#WORD**	**TEXT**	**TOOL**	**TOOL CATEGORY**	**TOOL LEMMA**	**COMBINED CATEGORY**	**COMBINED LEMMA**
1	84_1_5	84_1_5_1	cuarentón	DL	AJ	cuarentón	RU	cuarentón
		35_1_77_1	cuarentón	FDG	NC	cuarentón		
		84_1_5_1	null	UM	null	null		
CONTEXT		"El hijo de la novia, el derrumbe de un **cuarentón**"						
TRANSLATION		"The bride's son, the collapse of a **man in his forties**"						
COMMENTS		Avoids a POS combination failure.						

Table 2: An example of application of RULE B for lemma combination

6. Results

In this section, we present the statistical indicators of the lemma tag combination experiment described above. They give an idea of the correctness of the lemma tag that each tool associates to the different morphosyntactic units of the input text. The values of these indicators are shown in the bar chart included in Figure 3.

The results of each group of statistical indicators in this bar chart, namely *Correct*, *Wrong* and *Null*, are calculated as the arithmetic mean of the items correctly, incorrectly and not lemma tagged (respectively) by the corresponding tool. They have been grouped together, in order to show more clearly the behaviour of each tool, *i.e.* the combination tool (in dark blue), DataLexica (in maroon), FDG (in beige), and LACELL's POS tagger (in light blue), with respect to the other ones.

As can be seen in the figure, the value of the *Correct* indicator for the combination tool (94.94%) highly outperforms those of DataLexica (71.49%), FDG (75.22%) and UMurcia (49.65%). This is due to the fact that a high value of this indicator entails a higher number of correct lemma tags and, hence, points out a good performance of the tool under consideration. On the contrary, a high value of the *Wrong* indicator shows a bad performance, since it entails a higher number of wrongly annotated items. Accordingly, it can be observed that the values of this latter indicator corresponding to the combination tool (5.06%) also outperform clearly those of the other tools. In fact, they outperform five times the results of DataLexica (28.36%), and four times those of FDG (22.93%). They also surpass the results of LACELL's POS tagger (8.48%).

As for the values of the *Null* indicator, it can be observed that the combination tool (0.00%) outperforms the other three in this case as well (also in this case, a high value of the indicator points out a bad performance).

When trying to determine which of these results depended on the application of the rules and which ones depended on the interoperation of the three linguistic tools *per se*, it was found that the meta-rule presented above was applied in 56% of the cases. Taking into account that this

meta-rule was applied in those cases in which there is an inexistent or a very low degree of disagreement among the lemma tags assigned by each tool, it could be considered that 56% of the cases dealt directly with the interoperation of the tools. As for the remaining 44% cases, their correct annotation depended strongly on the rule system presented in this paper. However, almost 85% of them were correctly lemma-tagged by means of two other rules, whereas the remaining 15% was lemma-tagged by fairly secondary rules (they were very specific and domain-dependent cases).

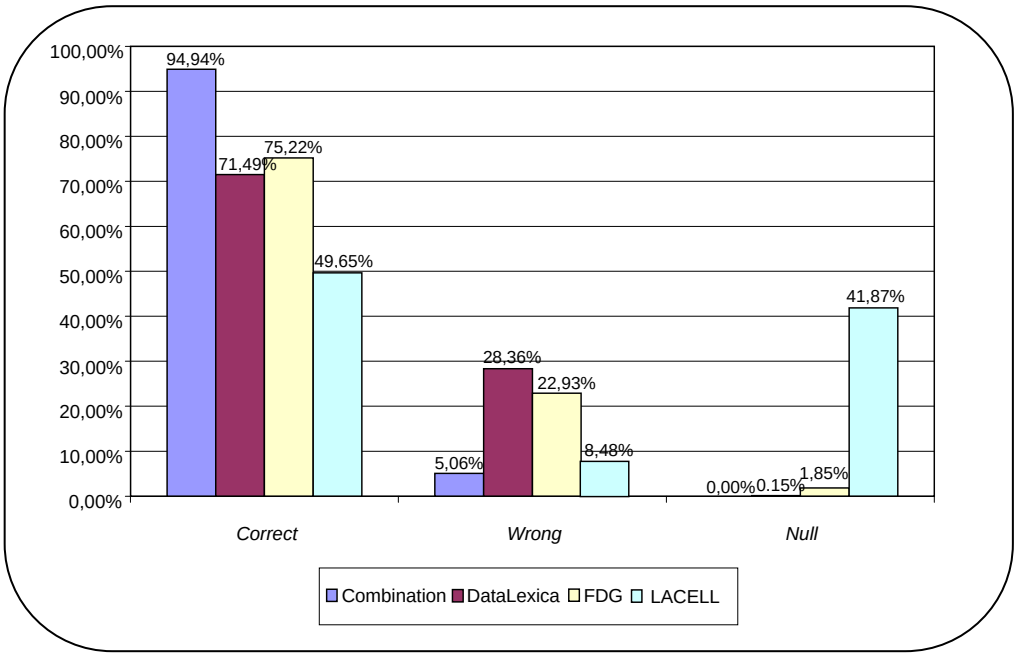

Figure 3: Comparative statistics associated to the correctness of lemma tags.

7. Conclusions

In this paper, we have shown that ontologies can be applied to map different annotation tagsets and schemas into a sort of standardised annotation schema, useful for the comparison and combination of the results of different linguistic annotation tools. We have also proved that the interoperation and the combination of lemma taggers by means of standard-compliant schemata, based on ontologies and W3C standard languages, can yield significantly better results than each of these lemma taggers separately.

Moreover, the architecture and the methodology presented in this paper help correct the errors introduced by certain linguistic annotation tools when annotating lemma (or POS) taggers. This can be done by contrasting and combining their annotations with the ones coming from other tools, not necessarily outperforming them, but built following other design criteria. This can be applied to the improvement of lemma (or POS) taggers, as well as to other linguistic annotation tools, for languages for which some annotation tools exist already, but which are not very accurate separately.

In addition, the architecture and the methodology presented can also be viewed as an automatic means of (1) assessing the correctness or the inter-annotation (dis)agreement of different types of tools developed for the annotation of the same type of linguistic phenomena; and (2) standardising (or, at least, unifying) the format and the schemas of annotation of different linguistic tools and resources.

8. Acknowledgements

This research has partly been supported by the ministry of Science and Technology grant TSI2007-65677C02 (**GeoBuddies** project) and by the European Commission's Seventh Framework Program (grant FP7-ICT-4-248458, **Monnet** project). We would also like to thank Mr. Javier Arrizabalaga for his help at the implementation of the architecture described in the present paper.

9. References

Aguado de Cea, G., Álvarez de Mon y Rego, I., Gómez-Pérez, A., Pareja-Lora, A. and Plaza-Arteche, R. (2002). "A semantic web page linguistic annotation model". *Proceedings of the AAAI'2002 Workshop: Semantic Web Meets Language Resources* (AAAI Technical Report WS-02-16), pp. 20–29.

Aguado de Cea, G., Álvarez de Mon, I., Gómez-Pérez, A., Pareja-Lora, A. (2003). "OntoTag: XML/RDF(S)/OWL Semantic Web Page Annotation in ContentWeb" in *Proceedings of the 2nd Workshop on NLP and XML (NLPXML-2003) – Language Technology and the Semantic Web* (pp. 25-32). 10th

Conference of the European Chapter of the Association for Computational Linguistics (EACL'03). Budapest, Hungary.

Aguado de Cea, G., Gómez-Pérez, A., Álvarez de Mon y Rego, I., Pareja-Lora, A. (2004a). "OntoTag's Linguistic Ontologies: Improving Semantic Web Annotations for a Better Language Understanding in Machines". *Proceedings of ITCC 2004*, vol. 2, pp 124-128.

Aguado de Cea, G., Gómez-Pérez, A., Álvarez de Mon y Rego, I., Pareja-Lora, A. (2004b). "OntoTag's linguistic ontologies: Enhancing higher level and Semantic Web annotations". *Proceedings of LREC 2004*, vol. VI, pp. 1905–1908.

Berners-Lee, T., Fischetti, M. (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Harper. San Francisco.

Borst, W. N. (1997). Construction of *Engineering Ontologies*. PhD thesis, University of Twente, Enschede.

EAGLES. (1996a). EAGLES: Recommendations for the Morphosyntactic Annotation of Corpora. EAGLES Document EAG—TCWG—MAC/R.

EAGLES. (1996b). EAGLES: Recommendations for the Syntactic Annotation of Corpora. EAGLES Document EAG—TCWG—SASG/1.8.

Gruber, T. R. (1993). "A Translation Approach to Portable Ontologies" in *Journal on Knowledge Acquisition*, Vol. 5(2), 199-220.

ISO. (2008a). Language resource management – Linguistic annotation framework. ISO/TC 37/SC 4 N522 – ISO/CD 24612.

ISO. (2008b). Language resource management – Morpho-Syntactic Annotation Framework. ISO/TC 37/SC 4 N225 – ISO/CD 24611.

Tapanainen, P., Järvinen, T. (1997). "A Non-Projective Dependency Parser". *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP).*

Vargas-Vera, M., Motta, E., Domingue, J., Shum, S. B., Lanzoni, M. (2000). Knowledge Extraction by Using an Ontology-based Annotation Tool. *Proceedings of the K-CAP'01 Workshop on Knowledge Markup and Semantic Annotation*, Victoria B.C., Canada.