Semantic role labeling tools trained on the Cast3LB-CoNNL-SemRol corpus

Roser Morante

ILK-Tilburg University
Postbus 90153, 5000 LE Tilburg
R.Morante@uvt.nl

Abstract

In this paper we present the Cast3LB–CoNLL–SemRol corpus, currently the only corpus of Spanish annotated with dependency syntax and semantic roles, and the tools that have been trained on the corpus: an ensemble of parsers and two dependency-based semantic role labelers that are the only semantic role labelers based on dependency syntax available for Spanish at this moment. One of the systems uses information from gold standard syntax, whereas the other one uses information from predicted syntax. The results of the first system (86 F_1) are comparable to current state of the art results for constituent-based semantic role labeling of Spanish. The results of the second are 11 points lower. This work has been carried out as part of the project *Técnicas semiautomáticas para el etiquetado de roles semánticos en corpus del español*.

1. Introduction

In this paper we present the Cast3LB–CoNLL–SemRol corpus, currently the only corpus of Spanish annotated with dependency syntax and semantic roles, and the tools that have been trained on the corpus: an ensemble of parsers and two dependency-based semantic role labelers that are the only semantic role labelers based on dependency syntax available for Spanish at this moment. One of the systems uses information from gold standard syntax, whereas the other one uses information from predicted syntax. The results of the first system (86 F_1) are comparable to current state of the art results for constituent-based semantic role labeling of Spanish. The results of the second are 11 points lower. This work has been carried out as part of the project *Técnicas semiautomáticas para el etiquetado de roles semánticos en corpus del español*.

Automatic semantic role labeling (ASRL) is a natural language processing task that consists in identifying the arguments of the predicates in a sentence and assigning a semantic role to them. The notion of semantic role is originally due to (Fillmore, 1968). Semantic roles are symbolic entities that describe the function of the participants in an event from the point of view of the situation in the real world. They allow to know who did what to whom when where how, etc. A predicate with a certain meaning assigns certain semantic roles to the participants in the event that the predicate denotes, regardless of the syntactic form of the sentence. For example, the predicate *annotate* would be assigned the same semantic roles in the two sentences in Example 1 despite the syntactic differences (active versus passive construction). So, a semantic role can be expressed with different syntactic structures and can occupy different positions in the sentence, especially in Spanish, where order constraints are less strict than in English. In Example 1, ARG0 is a postverbal prepositional phrase in (a), whereas it is a preverbal noun phrase in (b).

(1) a. [The corpus$_{ARG1}$] has been annotated [by three annotators$_{ARG0}$].

b. [Three annotators$_{ARG0}$] annotated [the corpus$_{ARG1}$].

Most of the semantic role labelers developed for English use information from constituent syntax. An exception is (Hacioglu, 2004). The basic referents are the systems participating in the CoNLL Shared Tasks 2004 (Màrquez and Carreras, 2004) and 2005 (Carreras and Màrquez, 2005). Some additional relevant systems are those in (Gildea and Jurafsky, 2002; Pradhan et al., 2005; Toutanova et al., 2005; Surdeanu et al., 2007). Most systems for English are trained on the PropBank (Palmer et al., 2005) corpus. In Section 2. we will point out the differences between the annotation of semantic roles in PropBank and in the Cast3LB–CoNLL–SemRol corpus.

For Catalan and Spanish two systems were developed for Task 9 Multilevel Semantic Annotation of Catalan and Spanish (Màrquez et al., 2007) in the framework of SemEval 2007, and a combined system that implements joint learning strategies is the one in (Surdeanu et al., 2008). The systems that we present in this paper are trained on the Cast3LB-CoNLL-SemRol corpus (Morante, 2006). The main difference between the systems that we present in this paper and the systems mentioned above is that they use information from dependency syntax, instead of constituent syntax.

Apart from semantic role labelers, the corpus has also been used to train three dependency parsers: Nivre's MaltParser (Nivre, 2006; Nivre et al., 2006), Canisius' memory-based constraint-satisfaction inference parser (Canisius and Tjong Kim Sang, 2007), and a new memory-based parser that operates with a single word-pair relation classifier. Since parser combination has proved to improve the performance of individual parsers (Henderson and Brill, 1999; Zeman and Žabokrtský, 2005; Sagae and Lavie, 2006), we also experimented with an ensemble of parsers that integrates the three individual parsers. As far as we know, this is the first ensemble of dependency parsers for Spanish.

The contents of the paper are organised as follows. Section 2. presents the Cast3LB–CoNLL-SemRol corpus. Section 3. describes the ensemble of parsers and Section 4. the semantic role labelers. Finally, in Section 5. we put forward some conclusions.

1	Asimismo	asimismo	r	rg	_	2	MOD	_
2	defiende	defender	v	vm	num=s,per=3,mod=i,tmp=p	0	ROOT	_
3	la	el	d	da	num=s,gen=f	4	ESP	_
4	financiación	financiación	n	nc	num=s,gen=f	2	CD	ARG1
5	pública	pública	a	aq	num=s,gen=f	4	CN	_
6	de	de	s	sp	for=s	4	CN	_
7	la	el	d	da	num=s,gen=f	8	ESP	_
8	investigación	investigación	n	nc	num=s,gen=f	6	_	_
9	básica	básico	a	aq	num=s,gen=f	8	CN	_
10	y	y	c	cc	_	2	CTE	_
11	pone	poner	v	vm	num=s,per=3,mod=i,tmp=p	10	CDO	_
12	de	de	s	sp	for=s	11	CC	ARG_ST
13	manifiesto	manifiesto	n	nc	gen=m,num=s	12	_	_
14	que	que	c	cs	_	18	_	_
15	las	el	d	da	gen=f,num=p	16	ESP	_
16	empresas	empresa	n	nc	gen=f,num=p	18	SUJ	ARG1
17	se	él	p	p0	per=3	18	_	_
18	centran	centrar	v	vm	num=p,per=3,mod=i,tmp=p	11	CD	ARG1
19	más	más	r	rg	_	20	CC	ARG_EXT
20	en	en	s	sp	for=s	18	CREG	ARG_LOC
21	la	el	d	da	num=s,gen=f	22	ESP	_
22	I+D	I+D	n	np	_	20	_	_
23	con	con	s	sp	for=s	18	CC	ARG_PRP
24	objetivos	objetivo	n	nc	gen=m,num=p	23	_	_
25	de	de	s	sp	for=s	24	CN	_
26	mercado	mercado	n	nc	gen=m,num=s	25	_	_
27	.	.	F	Fp	_	2	PUNC	_

Table 1: Example sentence of the Cast3LB–CoNLL corpus of Spanish

2. The Cast3LB–CoNLL corpus of Spanish

The Cast3LB–CoNLL corpus of Spanish is a manually revised version of the Cast3LB treebank (Civit et al., 2006) used in the CoNLL Shared Task 2006 that on dependency parsing (Buchholz and Marsi, 2006). The revision of the corpus was necessary due to the existence of errors, mostly caused by the conversion of the Cast3LB treebank, in constituent format, to the CoNLL shared task dependency format. The revision consisted of correcting errors in the dependency assignment and labeling. Examples of systematic errors were tagging the head of a noun phrase as a child of a prenominal adjective in that noun phrase, or tagging the verb head of a subordinate sentence as a child of a complement of that verb. In a few cases, wrong assignment of syntactic functions was also corrected, like cases of punctuation elements being assigned a syntactic function.

The Cast3LB–CoNLL–SemRol (Morante, 2006) is the Cast3LB–CoNLL corpus with an additional manually annotated layer of semantic roles. Only the semantic roles of predicates expressed by verbs have been annotated. The set of semantic roles has been defined with the purpose of generalising as much as possible the semantic relation that holds between a predicate and its complements. The annotation is not meant to code information about the syntactic structure. We defined 26 semantic roles that were sufficient to annotate the corpus (details and examples can be found in Morante (2006)):

- **ARG0** is assigned to prototypical agents, usually the subject of accusative verbs, ergative verbs in ergative constructions, and unergative verbs. We do not distinguish between different types of agents, like for example experiencers.
- **ARG1** is a prototypical theme, usually the object of accusative verbs, the subject of passive constructions, or the subject of ergative verbs in unaccusative constructions. We do not distinguish between different types of ARG1 like themes or patients.
- **ARGM** is the role assigned per default, when no other roles can be assigned.
- **ARG_ATR** is assigned to attributes, the complements required by the verbs *ser* (“to be”) and *estar* (“to be”) in their attributive meaning.
- **ARG_BEN** is assigned to arguments of the verb that refer to the entity that benefits from an action or to whom the action is directed. Most of the times it is the indirect object of ditransitive verbs like *dar* (“give”) or *decir* (“say”).
- **ARG_CAU** is assigned to the complements that refer to the entity or to the fact that causes an event to happen. It can be the subject of causative constructions, a prepositional phrase introduced by prepositions *por* or *al*, or a subordinate clause introduced by subordinate conjunctions *por lo que*, *pues*, *porque*.
- **ARG_COMP** is assigned to complements that refer to entities that have a relation of company, coordination, or collaboration with the entity that performs, experiences or suffers the event. ARG_COMP constituents are usually introduced by prepositions *con* or *junto con*.
- **ARG_CONC** is assigned to the complement of a verb that expresses a circumstance, despite which the event expressed by the verb holds or has taken place. ARG_CONC constituents are usually introduced by preposition *a pesar de* or subordinate conjunction *aunque*.
- **ARG_COND** is assigned to complements of the verb that express a condition. ARG_COND constituents are usually

introduced by the subordinate conjunction *si*.

- **ARG_CONS** is assigned to complements of the verb that express a consequence.
- **ARG_DEST** is used with events in which an entity moves along a path, physically or metaphorically. ARG_DEST is assigned to complements that express final end of the path, or destination. It can also express direction, or the entity into which something is converted with predicates that express change of state.
- **ARG_EXT** is assigned to complements that express the frequency or intensity with which something happens. It is usually assigned to adverbial constituents.
- **ARG_INSTR** is assigned to complements that refer to an entity used as instrument to perform the action, literally or metaphorically. It is usually assigned to constituents introduced by prepositions *con* or *gracias a*.
- **ARG_LOC** is assigned to complements of the predicate that refer to the location where the event happens.
- **ARG_MNR** is assigned to constituents that express the manner in which an event happens.
- **ARG_MEANS** is assigned to constituents that express the means by which an event happens.
- **ARG_OP** is assigned to constituents that express a term of comparison, an entity, event, or property that is compared or opposed to the main entity, event, or property.
- Like ARG_DEST, **ARG_OR** is used with events in which an entity moves along a path, physically or metaphorically. It is assigned to complements that express the origin of the path. It can also refer to the location from which the event is performed.
- **ARG_PRED** is assigned to predicative complements.
- **ARG_PRP** is assigned to constituents that express the purpose or goal of an event. It is usually introduced by prepositions *para, a, con el fin de, a fin de*, etc.
- **ARG_QUANT** is assigned to constituents that express quantity.
- **ARG_RES** is assigned to constituents that express the result of an event.
- **ARG_SRC** is assigned to the constituents that express the source of one of the entities involved in the event. It is not related to events that express a path.
- **ARG_ST** is assigned to constituents that express the state in which an entity affected by the event is.
- **ARG_TMP** is assigned to constituents that express temporal aspects of an event.
- **ARG_TOP** is assigned to constituents that express what the event is about, on what does the event focus, or what is the topic of the event.

The corpus contains 89199 words in 3303 sentences, from which 11023 are verbal forms corresponding to 1443 verb lemmas. 1369 verbs appear less than 20 times; 54 verbs, from 20 to 50 times; 12 verbs, 50 to 100 times: *tratar* (51), *dejar* (53), *acabar* (55), *pasar* (59), *parecer* (62), *seguir* (62), *quedar* (67), *encontrar* (68), *llevar* (68), *poner* (68), *deber* (75), *querer* (78), *dar* (86). 6 verbs, from 100 to 300 times: *saber* (101), *llegar* (107), *ver* (121), *ir* (132), *decir* (210), *tener* (243), *hacer* (253), *poder* (282), *estar* (296); and 2 verbs appear more than 800 times: *ser*, 1348 times and *haber*, 812 times.

Table 1 shows an example sentence of the corpus. Like in the CoNLL Shared Task 2006, sentences are separated by a blank line and fields are separated by a single tab character. A sentence consists of tokens, each one starting on a new line. A token consists of the following 8 fields that contain information about morphosyntactic features and non-projective dependencies:

1. ID: token counter, starting at 1 for each new sentence.
2. FORM: word form or punctuation symbol.
3. LEMMA: lemma of word form.
4. CPOSTAG: coarse-grained part-of-speech tag.
5. POSTAG: fine-grained part-of-speech tag.
6. FEATS: unordered set of syntactic and/or morphological features, separated by a vertical bar. If features are not available, the value of the feature is an underscore. The complete description of the CPOSTAG, POSTAG, and FEATS tags can be found in (Civit, 2002).
7. HEAD: head of the current token, which is either a value of ID or zero ('0') for the sentence root.
8. DEPREL: dependency relation to the HEAD. The set of tags is described in (Morante, 2006).

2.1. Related work

Semantic role labeling systems for English are usually trained on the PropBank corpus (Palmer et al., 2005). In the PropBank project, a layer of predicate–argument information was added to the Penn Treebank. The verb semantic arguments are numbered beginning by 0. Arg0 is the prototypical Agent and Arg1 is the prototypical Patient. For the higher numbers, as (Palmer et al., 2005, p.4) indicate, it is not possible to generalize across verbs. Sentence (a) in Example 2 would be annotated as in 2b.

(2) a. He wouldn't accept anything of value from those he was writing about.

b. [A0 He] [AM-MOD would] [AM-NEG n't] [V accept] [A1 anything of value] from [A2 those he was writing about]

The Cast3LB–CoNLL-SemRol corpus is annotated differently because the goal of the annotation is to capture generalisations of the semantic relation that holds between a predicate and its complements across verbs in order to check if the systems can learn these generalisations. A difference with PropBank is that we do not make a distinction between arguments subcategorised by the verb and adjuncts. All complements are annotated with the same set of roles. Another difference with PropBank is that what in PropBank are arguments with a number higher than 1 (Arg2, Arg3, etc) are notional semantic roles without a number in the Cast3LB–CoNLL-SemRol corpus. The distinction between Arg0 and Arg1 is kept. For example, A2 would be replaced by ARG_OR in Example 2b, whereas

ARG2-at would be replaced by ARG_LOC in Example 3[1], capturing the fact that ARG_LOC and ARG_OR are different semantic notions. This distinction cannot be made if the A2 label is used.

(3) He said that if [ARG1 little cocoa] [ARGM-MNR actually] has [rel arrived] at [ARG2-at the ports] , shipping delays could result .

The corpus Cast3LB–CoNLL-SemRol is similar to the corpora of Spanish that were used in Task 9 of SemEval 2007 (Màrquez et al., 2007). The main difference is that Cast3LB–CoNLL-SemRol is annotated with dependency syntax whereas the Task 9 corpora are annotated with constituent syntax.

3. Ensemble of dependency parsers

The corpus Cast3LB–CoNLL has been used to train an ensemble of dependency parsers that integrates three parsers: the MaltParser (Nivre, 2006), a parser based on constraint satisfaction (Canisius and Tjong Kim Sang, 2007), and a memory–based parser (Morante, 2008). Like in (Sagae and Lavie, 2006), the ensemble that we present works in two stages. In the first stage, each of the three parsers analyzes an input sentence and produces a syntactic structure. The unlabeled attachment scores in this stage range from 82 to 86 %, according to the evaluation metrics used in the CoNLL Shared Task 2006. In the second stage, a voting algorithm is applied that takes into account the results of the parsers in the first stage in order to provide a final solution.

3.1. Individual parsers

3.1.1. MaltParser (MP)

The MaltParser 0.4[2] (Nivre, 2006; Nivre et al., 2006) is an inductive dependency parser that uses four essential components: a deterministic algorithm for building labeled projective dependency graphs; history-based feature models for predicting the next parser action; support vector machines for mapping histories to parser actions; and graph transformations for recovering non-projective structures.
For our experiments we trained the parser using the support vector machines algorithm (LIBSVM (Chang and Lin, 2005)), with the same parameter options used by (Nivre et al., 2006) in the CoNLL Shared Task 2006. The parser algorithm used was Nivre, with the options arc order eager, shift before reduce and allow reduction of unattached tokens.

3.1.2. Memory-based constraint satisfaction parser (MB1)

The memory-based constraint satisfaction parser (Canisius and Tjong Kim Sang, 2007) uses three memory-based classifiers that predict weighted soft-constraints on the structure of the parse tree. Each predicted constraint covers a small part of the complete dependency tree, and overlap between them ensures that global output structure is taken into account. A dynamic programming algorithm for dependency parsing is used to find the optimal solution to the constraint satisfaction problem thus obtained.

3.1.3. Memory-based constraint satisfaction parser (MB2)

The memory-based single classifier parser (Morante, 2008) consists of a single classifier that predicts the relation between two words in a sentence, and a decision heuristics that chooses among the dependency relations that the classifier has predicted for one word, based on information from the classifier output.

3.1.4. Results

The global results of the three parsers are shown in Table 2 in terms of Labeled Attachment Score (LAS), Unlabeled Attachment Score (UAS), and Label Accuracy (LAc) according to the evaluation metrics used in the CoNLL Shared Task 2006 (Buchholz and Marsi, 2006). The MP performs significantly better than MB1 and MB2, whereas MB1 and MB2 perform similarly in spite of the fact that their approach to memory-based learning is different: MB1 applies constraint satisfaction, and MB2 is based on only one classifier and heuristics that rely on the distance of the predicted class to the nearest neighbor and on the class distribution.

	MP	MB1	MB2
LAS	80.45 %	75.74 %	75.44 %
UAS	87.42 %	82.44 %	82.75 %
LAc	85.12 %	81.95 %	81.35 %

Table 2: Results of the individual parsers.

3.2. Ensemble system

The ensemble system operates in two stages. In the first stage, each of the three parsers analyzes an input sentence and produces a dependency graph. The results of the individual parsers were presented in Table 2 in the previous section. In the second stage, a voting system distills a final dependency graph out of the three first-stage dependency graphs. Voting techniques have been previously applied to dependency parsing (Sagae and Lavie, 2006; Zeman and Žabokrtský, 2005).
We provide results of three different voting systems, that take into account agreement among classifiers and/or the normalized F1 value of each classifier for each dependency relation:

- **VS1**: the system votes for the solution of the single classifier that has the higher F1 for the dependency relation that the single classifier predicts.
- **VS2**: the system votes for the solution of the MP, unless MB1 and MB2 agree, in which case the MB1 and MB2 solution is chosen.
- **VS3**: the system votes for the solution of the MP, unless MB1 and MB2 agree or the three parsers disagree. In the first case, the MB1 and MB2 solution is chosen, and in the second, the system votes for the solution of the single classifier that has the higher F1 for the syntactic function that the single classifier predicts.

[1]Example taken from http://www.cs.rochester.edu/~gildea/PropBank/a/arrive.html.

[2]Web page of MaltParser 0.4: http://w3.msi.vxu.se/∼nivre/research/MaltParser.html.

	VS1	dif.MP	VS2	dif.MP	VS3	dif.MP	VS4	dif.MP
LAS	80.53%	+0.08	81.04%	+0.59	81.09%	+0.64	79.71%	-0.74
UAS	87.43%	+0.01	87.68%	+0.26	87.68%	+0.26	86.07%	-1.35
LAc	85.22%	+0.10	85.71%	+0.59	85.78%	+0.66	85.92%	+0.80

Table 3: LAS, UAS, and LAc of the different versions of the ensemble of parsers compared to the MaltParser.

- **VS4**: the system votes for system VS1 unless two single systems agree. In this case, the system votes for the solution agreed by them.

The results of the different versions of the ensemble system are presented in Table 3 as well as the improvement over the MP. Results show that combined systems VS1, VS2 and VS3 perform better than the best parser, although the difference is insignificant, since it reduces the error of MP in less than 5% (4.44%). Combined system VS4 improves only in accuracy over the results of the best system.
VS1 is the system that improves the least because the MP has the better F1 scores for 19 of the 25 dependency relations. That VS2 and VS3 do no improve significantly might be due to the fact that some agreement cases between MB1 and MB2 can be errors.
VS3 is the voting system that performs better: by voting for the agreement between MB1 and MB2, or for the system with higher F1 in case of complete disagreement, more errors are eliminated than errors are introduced. For further research it would be interesting to analyze if it is possible to eliminate more errors by introducing specific voting strategies per dependency relation.

3.3. Related work

The related work we are aware of deals with languages other than Spanish. (Zeman and Žabokrtský, 2005) tested several approaches for combining dependency parsers for Czech. They found that the best method was accuracy-aware voting, which reduced the error of the best parser in 13%. Differences between their approach an ours are that they experiment with seven parsers, they perform stacking, and they check that the resulting structure is a well-formed tree.
(Sagae and Lavie, 2006) experiment with six parsers on the Wall Street Journal corpus. They apply a two stage procedure of reparsing focusing on unlabeled dependencies. In the first stage, *m* different parsers analyze an input sentence. In the second stage, a parsing algorithm is applied taking into account the analysis produced by each parser in the first stage. They reparse the sentence based on the output of *m* parsers in order to maximize the number of votes for a well-formed dependency structure. Their experiments increase the accuracy of the best parser in 1.7%.
(Nivre et al., 2007) combined the outputs of the parsers participating in the CoNLL Shared Task 2007 on dependency parsing using the method of (Sagae and Lavie, 2006). They show that accuracy never falls below the performance of the top three systems, although it degrades after ten different parsers have been added.

4. Semantic role labelers

Two semantic role labelers have been trained on the Cast3LB-CoNLL-SemRol corpus. The only difference between the systems is that one uses information from gold standard syntax, whereas the other one uses information from predicted syntax. The engine of the two semantic role labelers is a memory-based classifier. Memory-based language processing (Daelemans and van den Bosch, 2005) is based on the idea that NLP problems can be solved by storing annotated examples of the problem in their literal form in memory, and applying similarity-based reasoning on these examples in order to solve new ones.

4.1. System based on gold standard syntax (SRL-GS)

The system based on gold standard syntax solves the task in three phases:

1. A pre-processing phase that consists of identifying the potential candidates to be assigned a semantic role or a semantic verb class. The system starts by detecting a target verb and the clause boundaries in order to look for the siblings of the verb that exist within the same clause. These tokens will be the focal elements of the examples in each training set. For a sentence like the one presented in Table 1 the focal elements would be the ones in Table 4. Each of the rows in the table would be an instance, with its corresponding features.

Focal element	Verb
asimismo	defiende
financiacin	defiende
manifiesto	pone
centran	pone
empresas	centran
se	centran
ms	centran
I+D	centran
objetivos	centran

Table 4: Focal elements of the sentence in Table 1.

2. A classification phase, i.e. the actual assignment of roles and verb classes.

   We use the IB1 classifier as implemented in TiMBL (version 6.0) (Daelemans et al., 2007), a supervised inductive algorithm for learning classification tasks based on the k-nearest neighbor classification rule (Cover and Hart, 1967). In IB1, similarity is defined by a feature-level distance metric between a test instance and a memorized example. The metric combines a per-feature value distance metric with global

feature weights that account for relative differences in discriminative power of the features. The IB1 algorithm is parameterized by using Jeffrey Divergence as the similarity metric, gain ratio for feature weighting, using 11 k-nearest neighbors, and weighting the class vote of neighbors as a function of their inverse linear distance.

3. A postprocessing phase, in which the predictions of some semantic roles are corrected by taking into consideration all the predictions in the clause. For example, if the system has predicted two ARG0 for the same predicate, one of the predictions is modified.

4.2. Features

There are three groups of features: about the sibling in focus, about the verb, and about the clause.

- Sibling in focus (26): content word, content word lemma, gender and number; is content word a named entity? a temporal adverb? a locative adverb?; POS of the two previous words to the content word and of the three next words; POS and lemma of the three first words of the sibling; preposition; POS and POS type of the head; syntactic function; relative position to the verb; relative position to the verb of the next sibling; string with the POS of all words in the sibling; string with all nouns, adjectives, adverbs and verbs in the sibling.

- Verb (12): distance to the sibling in focus; word, lemma, POS type, two previous and next words; concordance in gender with the content word of the sibling; is the verb causative? pronominal? passive?

- Clause (12): number of constituents with function CC (adverbial complement); number of constituents; relative position to the verb of siblings with syntactic function SUJ, CD, CAG, CI, ATR, CPRED.CD, CPRED.SUJ, CREG; string with the POS of the head of all siblings; string with the syntactic function of all siblings.

4.3. Results

We divided the corpus in train, development and test sets. Feature selection was performed by starting with a set of basic features (essentially the identity and the parts-of-speech tags of the head words involved, in their local context) and gradually adding new features. Table 5 shows the results of the system for the test set. It achieves 0.86 F_1, 0.88 precision and 0.84 recall. The system performs very well with very frequent roles (ARG0, ARG1) and with roles that have morphosyntactic markers (ARG_ATR, ARG_PRED, ARG_BEN). With unfrequent roles performance is variable (ARG_COMP 0.77 versus ARG_OP 0.33). The system performs less well with some roles that are relatively frequent, that do not have syntactic markers and that might encode a diversity of semantic concepts (ARG_MNR).

We computed the effect of removing groups of features. Removing the features with information about the sibling in focus causes a decrease of 12.12 in F_1; removing the features about the verb, a decrease of 1.31; the features about the clause, a decrease of 0.97, and the features about the content word, a decrease of 0.64. So the features about the sibling in focus seem to be the most informative ones.

Semantic Role	Total	Precision	Recall	F_1
ARG_CONC	9	0.83	0.55	0.66
ARG_TMP	192	0.80	0.86	0.83
ARG_MEANS	2	0.00	0.00	0.00
ARG_BEN	78	0.86	0.88	0.87
ARG_OP	4	0.50	0.25	0.33
ARG_TOP	23	0.63	0.22	0.32
ARG_DEST	36	0.60	0.66	0.63
ARG0	285	0.91	0.93	0.92
ARG_CONS	6	1.00	0.17	0.28
ARGM	53	0.50	0.23	0.31
ARG_OR	21	0.92	0.52	0.67
ARG_COND	9	1.00	0.55	0.71
ARG_ST	5	1.00	0.20	0.33
ARG_LOC	124	0.91	0.78	0.84
ARG_CAU	34	0.67	0.85	0.75
ARG_COMP	14	0.83	0.71	0.77
ARG_ATR	140	0.98	1.00	0.99
ARG_RES	1	0.00	0.00	0.00
ARG_PRP	33	0.79	0.69	0.74
ARG_PRED	45	0.91	0.91	0.91
ARG_SRC	2	0.50	0.50	0.50
ARG_EXT	22	0.92	0.54	0.68
ARG1	735	0.94	0.93	0.94
ARG_INSTR	14	0.80	0.28	0.42
ARG_MNR	88	0.65	0.62	0.63
Overall	1975	0.88	0.84	0.86

Table 5: Results of the SRL-GS system per role.

4.4. System based on predicted syntax (SRL-PS)

The system based on predicted syntax is exactly the same as the previous system, except for the fact that it uses information from syntax predicted with the MaltParser as presented in Subsection 3.1.1..

4.4.1. Results

The results of this system are presented in Table 6: 0.75 F_1, 0.82 precision, and 0.68 recall. Compared to the SRL-GS system, performance drops 0.11 points, mainly due to a decrease in recall (-0.16). Precision decreases in 0.6 points. Performance decreases around 0.10 with the most frequent roles (ARG0, ARG1). It does not decrease with two roles that have very clear morphosyntactic markers (ARG_ATR, ARG_PRED), and it decreases considerably (-0.24) with one ARG_BEN). The decrease with unfrequent roles is irregular (from nothing to 0.51).

Removing the features with information about the sibling in focus causes a decrease of 6.17 in F1; removing the features about the verb, a decrease of 2.23; the features about the clause, a decrease of 0.95, and about the content word, a decrease of 0.76. So, compared to the SRL-GS system, the decrease caused by removing the features about the sibling

Semantic Role	Total	Precision	Recall	F_1	dif
ARG_CONC	9	1.00	0.11	0.20	-0.46
ARG_TMP	192	0.82	0.67	0.74	-0.09
ARG_MEANS	2	0.00	0.00	0.00	0.00
ARG_BEN	78	0.77	0.53	0.63	-0.24
ARG_OP	4	1	0.25	0.40	+0.09
ARG_TOP	23	0.71	0.21	0.33	+0.01
ARG_DEST	36	0.44	0.30	0.36	-0.27
ARG0	285	0.81	0.83	0.82	-0.10
ARG_CONS	6	0.25	0.16	0.20	-0.08
ARGM	53	0.25	0.07	0.11	-0.20
ARG_OR	21	1.00	0.23	0.38	-0.29
ARG_COND	9	1.00	0.11	0.20	-0.51
ARG_ST	5	0.00	0.00	0.00	-0.33
ARG_LOC	124	0.70	0.70	0.70	-0.14
ARG_CAU	34	0.60	0.50	0.54	-0.21
ARG_COMP	14	0.33	0.21	0.26	-0.51
ARG_ATR	140	0.98	1.00	0.99	0.00
ARG_RES	1	0.00	0.00	0.00	0.00
ARG_PRP	33	0.71	0.30	0.42	-0.32
ARG_PRED	45	0.86	0.88	0.87	0.04
ARG_SRC	2	0.50	0.50	0.50	0.00
ARG_EXT	22	0.81	0.40	0.54	-0.14
ARG1	735	0.87	0.78	0.82	-0.12
ARG_INSTR	14	0.85	0.42	0.57	+0.15
ARG_MNR	88	0.76	0.36	0.49	-0.14
Overall	1975	0.82	0.68	0.75	-0.11

Table 6: Results of the SRL-PS system per role compared to the results of the SRL-GS system.

in focus is lower. This can be explained by the errors in the syntactic tree that the SRL-PS system has as input.

4.5. Related work

The results of the SRL-GS system are comparable to the results of the existing semantic role labeling systems for Spanish that use information from gold standard constituent syntax: the highest score of the systems participating in Task 9 of SemEval 2007 (Màrquez et al., 2007) was 0.84; (Morante and van den Bosch, 2007) report a maximum F_1 of 0.85 with a memory-based system very similar to SRL-GS. (Surdeanu et al., 2008) report 0.86 with a combined system that implements joint learning strategies. Because the corpus used and the annotation is not exactly the same as in the mentioned systems, the results are not completely comparable. However, they give an indication that the performance of systems that use information from dependency syntax is similar to that of the systems that use information from constituent syntax. (Hacioglu, 2004) describes a system for English that achieves 0.84 F_1 using information from dependency syntax converted from constituent syntax. We cannot compare the results of the SRL-PS system because we are not aware of semantic role labelers of Spanish that use predicted syntax.

5. Conclusions

In this paper we presented the Cast3LB–CoNLL–SemRol corpus, currently the only corpus of Spanish annotated with dependency syntax and semantic roles, and the tools that have been trained on the corpus: an ensemble of parsers and two dependency-based semantic role labelers that are the only semantic role labelers based on dependency syntax available for Spanish at this moment.

The results of the ensemble of parsers are only slightly better than the results of the best parser; the error reduction of the label accuracy score reaches 4.44%. This is due to the fact that there are only three parsers, one of which performs clearly better than the other two, which perform very similarly. The best results were obtained by the voting system that gives priority to the decisions of the best parser, unless the other two parsers agree, in which case their solution is chosen, or the three parsers disagree, in which case the system votes for the solution of the single classifier that has the higher F1 for the dependency relation that the single classifier predicts. We consider the results to be promising enough to continue our research.

The results of the semantic role labelers shows that the performance of the system that uses information from predicted syntax decreases in 0.11 compared to the performance of the system that uses gold standard syntax. In both systems the features from the sibling in focus are very informative, though the effect of removing these features is lower in the system based on predicted syntax.

The results of the system based on gold standard dependency syntax are similar to the results obtained by existing systems based on gold standard constituent syntax. We conclude that syntactic information increases performance of a semantic role labeler regardless of the type of syntax used (constituent or dependency). However, if a system uses information from predicted syntax, other features have to be found in order to compensate for errors in the syntactic tree. Using the same features provokes a considerable decrease in performance.

Further research will focus on improving the system based on predicted syntax by incorporating an ensemble of parsers, instead of a single parser, and on engineering different features that are more robust to syntactic errors.

Acknowledgements

This research has been funded by the grant EX2005-1145 awarded by the Spanish Ministerio de Educación y Ciencia to the project *Técnicas semiautomáticas para el etiquetado de roles semánticos en corpus del español*.

6. References

S. Buchholz and E. Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the X CoNLL Shared Task*. SIGNLL.

S. Canisius and E. Tjong Kim Sang. 2007. A constraint satisfaction approach to dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 1124–1128.

Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 152–164, Ann Arbor, Michigan, June. Association for Computational Linguistics.

C.C. Chang and C.J. Lin. 2005. LIBSVM: A library for support vector machines. URL:http:// www.csie.ntu.edu.tw/ ∼cjlin/ papers/ libsvm.pdf.

M. Civit, M.A. Martí, and N. Bufí, 2006. *Advances in Natural Language Processing (LNAI, 4139)*, chapter Cat3LB and Cast3LB: from constituents to dependencies, pages 141–153. Springer Verlag, Berlin.

M. Civit. 2002. Guía para la anotación morfológica del corpus CLiC-TALP (versión 3). X-TRACT-II WP-00-06, CliC–UB.

T. M. Cover and P. E. Hart. 1967. Nearest neighbor pattern classification. *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, 13:21–27.

W. Daelemans and A. van den Bosch. 2005. *Memory-based language processing.* Cambridge University Press, Cambridge, UK.

W. Daelemans, J. Zavrel, K. Van der Sloot, and A. Van den Bosch. 2007. TiMBL: Tilburg memory based learner, version 5.1, reference guide. Technical Report Series 07-03, ILK, Tilburg, The Netherlands.

Ch. Fillmore, 1968. *Universals in Linguistic Theory*, chapter The case for case, pages 1–88. Holt, Rinehart, London.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

K. Hacioglu. 2004. Semantic role labeling using dependency trees. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, Morristown, NJ, USA. ACL.

J. Henderson and E. Brill. 1999. Exploiting diversity in natural language processing: combining parsers. In *Proceedings of the Fourth Conference on Empirical Methods in Natural Language Processing (EMNLP)*, College Park, Maryland.

Ll. Màrquez and X. Carreras. 2004. Introduction to the CoNLL-2004 Shared Task: Semantic role labeling. In *Proceedings of the 8th Conference on Computational Natural Language Learning*, pages 89–97, Boston. ACL.

Ll. Màrquez, L. Villarejo, M.A. Martí, and M. Taulé. 2007. Semeval-2007 task 09: Multilevel semantic annotation of Catalan and Spanish. In *Proc. of SemEval*.

R. Morante and A. van den Bosch. 2007. Memory–based semantic role labelling. In *Proc. of the International Conference on Recent Advances in Natural Language Processing (RANLP-2007)*, pages 388–394, Borovets, Bulgaria.

R. Morante. 2006. Semantic role annotation in the Cast3LB-CoNNL-SemRol corpus. Induction of Linguistic Knowledge Research Group Technical Report ILK 06-03, Tilburg University, Tilburg.

R. Morante. 2008. Experiments with an ensemble of spanish dependency parsers. *Procesamiento del Lenguaje Natural*, 40.

J. Nivre, J. Hall, J. Nilsson, G. Eryigit, and S. Marinov. 2006. Labeled pseudo–projective dependency parsing with support vector machines. In *Proceedings of the Tenth Conference on Computational Natural Language Learning, CoNLL-X*, New York City, NY, June.

J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The CoNLL-2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague.

J. Nivre. 2006. *Inductive Dependency Parsing*. Springer.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105.

S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J. Martin, and D. Jurafsky. 2005. Support vector learning for semantic argument classification. *Machine Learning*, 60(1–3):11–39, September.

K. Sagae and A. Lavie. 2006. Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference on the North American Chapter of the ACL*, pages 129–132, New York. ACL.

Mihai Surdeanu, Lluís Màrquez, Xavier Carreras, and Pere R. Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research (JAIR)*, 29:105–151.

M. Surdeanu, R. Morante, and Ll. Màrquez., 2008. *Proceedings of the Computational Linguistics and Intelligent Text Processing 9th International Conference, CICLing 2008*, volume 4919/2008 of *Lecture Notes in Computer Science*, chapter Analysis of Joint Inference Strategies for the Semantic Role Labeling of Spanish and Catalan, pages 206–218. Springer, Berlin/Heidelberg.

K. Toutanova, A. Haghighi, and Ch.D. Manning. 2005. Joint learning improves semantic role labeling. In *Proceedings of the 43th Annual Conference of the Association for Computational Linguistics (ACL-05)*, Ann Arbor, Michigan.

D. Zeman and Z. *Ž*abokrtský. 2005. Improving parsing accuracy by combining diverse dependency parsers. In *Proceedings of the International Workshop on Parsing Technologies*, Vancouver, Canada.