# Integration of Data from a Syntactic Lexicon into Generative and Discriminative Probabilistic Parsers

**Anthony Sigogne**
Université Paris-Est, LIGM
sigogne@univ-mlv.fr

**Matthieu Constant**
Université Paris-Est, LIGM
mconstan@univ-mlv.fr

**Éric Laporte**
Université Paris-Est, LIGM
laporte@univ-mlv.fr

## Abstract

This article evaluates the integration of data extracted from a syntactic lexicon, namely the Lexicon-Grammar, into several probabilistic parsers for French. We show that by modifying the Part-of-Speech tags of verbs and verbal nouns of a treebank, we obtain accurate performances with a parser based on Probabilistic Context-Free Grammars (Petrov et al., 2006) and a discriminative parser based on a reranking algorithm (Charniak and Johnson, 2005).

## 1 Introduction

Syntactic lexicons are rich language resources that may contain useful data for parsers like subcategorisation frames, as they provide, for each lexical entry, information about its syntactic behaviors. Most of the time, these lexicons only deal with verbs. Few, like the Lexicon-Grammar (Gross, 1994), deal with other categories like nouns, adjectives or adverbs. Many works on symbolic parsing studied the use of a syntactic lexicon, in particular linguistic formalisms like Lexical-Functional Grammars [LFG] (Kaplan and Maxwell, 1994; Riezler et al., 2002; Sagot, 2006) or Tree Adjoining Grammars [TAG] (Joshi, 1987; Sagot and Tolone, 2009; de La Clergerie, 2010). For probabilistic parsing, we can cite LFG (Cahill, 2004; O'Donovan et al., 2005; Schluter and Genabith, 2008), Head-Driven Phrase Structure Grammar [HPSG] (Carroll and Fang, 2004) and Probabilistic Context-Free Grammars [PCFG] (Briscoe and Carroll, 1997; Deoskar, 2008). The latter has incorporated valence features to PCFGs and lexicons and observes slight improvements on performances. However, lexical resources that contain valence features were obtained automatically from a corpus. Furthermore, valence features are mainly used on verbs. In this paper, we will show how we can exploit information contained in the Lexicon-Grammar in order to improve probabilistic parsers. We will in particular focus on verbs and verbal nouns[1].

In section 2, we describe the probabilistic parsers used in our experiments. Section 3 briefly introduces the Lexicon-Grammar. We detail information contained in this lexicon that can be used for parsing. Then, in section 4, we present methods to integrate this information into parsers and, in section 5, we describe our experiments and discuss the obtained results.

## 2 Statistical parsers

In our experiments, we used two types of parsers: a generative parser that generates the *n-best* parses ($n$ most probable parses) for a sentence according to a PCFG; a reranker that reranks the *n-best* parses generated from the PCFG parser according to a discriminative probabilistic model.

### 2.1 Non-lexicalized PCFG parser

The PCFG parser, used into our experiments, is the Berkeley Parser (called BKY thereafter) (Petrov et al., 2006)[2]. This parser is based on a non-lexicalized PCFG model. The main problem of non-lexicalized context-free grammars is that preterminal symbols encode too general information which weakly discriminates syntactic ambiguities. BKY tries to handle the problem by generating a grammar containing complex pre-terminals. It follows the principle of latent annotations introduced by (Matsuzaki et al., 2005). It consists in creating iteratively several grammars, which have a tagset increasingly complex. For each iteration, a symbol of the grammar is splitted in several symbols

[1] Verbal nouns are nouns playing the role of a predicate in the sentence.

[2] The Berkeley parser is freely available at http://code.google.com/p/berkeleyparser/downloads/list

*Proceedings of Recent Advances in Natural Language Processing*, pages 363–370,
Hissar, Bulgaria, 12-14 September 2011.

according to the different syntactic behaviors of the symbol that occur in a treebank. Parameters of the latent grammar are estimated with an algorithm based on Expectation-Maximisation (EM). Within the framework of French, (Seddah et al., 2009) have shown that BKY produces *state-of-the-art* performances. They have also shown that several parsers, based on the lexicalized paradigm (phrasal nodes are annotated with their headword), achieved lower scores than BKY.

### 2.2 Reranking parser

We have also experimented the integration of a reranker as a post-process of BKY output. For a given sentence $s$, a reranker selects the best parse $y$ among the set of candidates $Y(s)$ according to a scoring function $V_\theta$ :

$$y^\star = argmax_{y \in Y(s)} V_\theta(y) \qquad (1)$$

The set of candidates $Y(s)$ is the *n-best* parses output of the baseline parser (BKY in our case), $Y(s) = \{y_1, y_2, ..., y_n\}$. The *n-best* parses correspond to the $n$ most probable parses according to the probability model of the parser. The scoring function $V_\theta$ is defined by the dot product of a weight vector $\theta$ and a feature vector $f$ :

$$V_\theta(y) = \theta.f(y) = \sum_{j=1}^{m} \theta_j.f_j(y) \qquad (2)$$

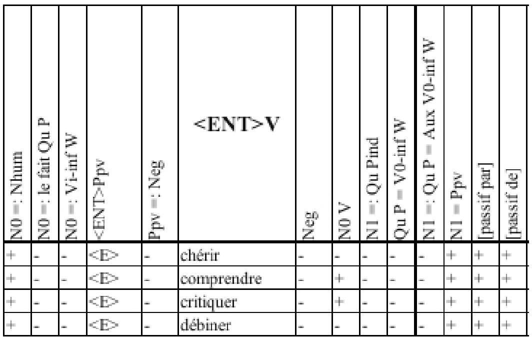
where the feature vector $f(y)$ is a vector of $m$ functions $f = (f_1, f_2, ..., f_m)$, and each feature function $f_j$ maps a parse $y$ to a real number $f_j(y)$. The first feature $f_1(y)$ is the probability of the parse given by the n-best parser (cf. (Charniak and Johnson, 2005)). All remaining features are integer values, and each of them is the number of times that the feature occurs in parse $y$. Features belong to feature schemas which are abstract schemas from which specific features are instantiated. Feature schemas that we used during our experiments are specified in the table 1. For example, a feature $f_{10}(y)$, which is an instance of the feature schema *Rule*, counts the number of times that a nominal phrase in $y$ is the head of a rule which has a determinant and a noun as children. The weight vector $\theta$ can be estimated by a machine learning algorithm from a treebank corpus which contains the gold parse for each sentence. In our case, we will use the Maximum Entropy estimator, as in (Charniak and Johnson, 2005).

| Feature schemas | |
|---|---|
| Rule | Edges |
| Word | WordEdges |
| Heavy | Heads |
| HeadTree | WProj |
| Bigrams$^\triangle$ | NgramTree |
| Trigrams$^\triangle$ | |

Table 1: Features used in this work. Those with a $^\triangle$ are from (Collins, 2000), and others are from (Charniak and Johnson, 2005)

## 3 Lexicon-Grammar

The Lexicon-Grammar [LG] is the richest source of syntactic and lexical information for French[3] that focuses not only on verbs but also on verbal nouns, adjectives, adverbs and frozen (or fixed) sentences. Its development started in the 70's by Maurice Gross and his team (Gross, 1994). It is a syntactic lexicon represented in the form of tables. Each table encodes lexical items of a particular category sharing several syntactic properties (e.g. subcategorization information). A lexical item is a lemmatized form that can be present in one or more tables depending on its meaning and its syntactic properties. Each table row corresponds to a lexical item and a column corresponds to a property (e.g. syntactic constructions, argument distribution, and so on). A cell encodes whether a lexical item accepts a given property. Figure 1 shows a sample of verb table *12*. In this table, we can see that the verb *chérir* (*to cherish*) accepts a human subject (pointed out by a + in the property *N0 =: Nhum*) but this verb cannot be intransitive (pointed out by a − in the property *N0 V*). Recently, these tables have been made consistent and explicit (Tolone, 2011) in order to be



| N0 =: Nhum | N0 =: le fait Qu P | N0 =: Vi-inf W | <ENT>Ppv | Ppv =: Neg | <ENT>V | Neg | N0 V | N1 =: Qu Pind | Qu P = V0-inf W | N1 =: Qu P = Aux V0-inf W | N1 = Ppv | [passif par] | [passif de] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| + | - | - | <E> | - | chérir | - | - | - | - | - | + | + | + |
| + | - | - | <E> | - | comprendre | - | + | - | - | - | + | + | + |
| + | - | - | <E> | - | critiquer | - | + | - | - | - | + | + | + |
| + | - | - | <E> | - | débiner | - | - | - | - | - | + | + | + |

Figure 1: Sample of verb table *12*

[3] We can also cite lexicons like LVF (Dubois and Dubois-Charlier, 1997), Dicovalence (Eynde and Piet, 2003) and Lefff (Sagot, 2010).

exploitable for NLP. They also have been transformed in a XML-structured format (Constant and Tolone, 2008)[4]. Each lexical entry is associated with its table identifier, its possible arguments and its syntactic constructions.
For the verbs, we manually constructed a hierarchy of the tables on several levels[5]. Each level contains classes which group LG tables which may not share all their defining properties but have a relatively similar syntactic behavior. Figure 2 shows a sample of the hierarchy. The tables *4, 6* and *12* are grouped into a class called *QTD2* (transitive sentence with two arguments and sentential complements). Then, this class is grouped with other classes at the superior level of the hierarchy to form a class called *TD2* (transitive sentence with two arguments). The characteristics of each level are given in the table 2[6] (level 0 represents the set of tables of the LG). We can state that there are 5,923 distinct verbal forms for 13,862 resulting entries in tables of verbs. The column *#classes* specifies the number of distinct classes. The columns *AVG_1* and *AVG_2* respectively indicate the average number of entries per class and the average number of classes per distinct verbal form.

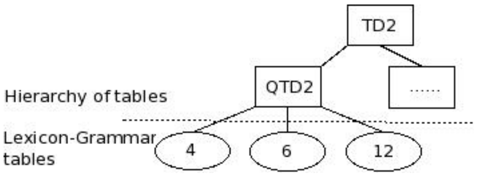


Figure 2: Sample of the hierarchy of verb tables

| Level | #classes | AVG_1 | AVG_2 |
|---|---|---|---|
| 0 | 67 | 207 | 2.15 |
| 1 | 13 | 1,066 | 1.82 |
| 2 | 10 | 1,386 | 1.75 |
| 3 | 4 | 3,465 | 1.44 |

Table 2: Characteristics of the hierarchy of verb tables

The hierarchy of tables have the advantage of reducing the number of classes associated with each verb of the tables. We will see that this ambiguity reduction is crucial in our experiments.

## 4 Exploitation of the Lexicon-Grammar data

Many experiments about parsing, within the framework of French (Crabbé and Candito, 2008; Seddah et al., 2009), have shown that refining the tagset of the training corpus improves performances of the parser. We will follow their works by integrating information from the Lexicon-Grammar to part-of-speech tags. In this article, we will only focus on tables of verbs and verbal nouns.

Table identifiers of the lexical entries are important hints about their syntactic behaviors. For example, the table *31R* indicates that all verbs belonging to this table are intransitive. The first experiment, called **AnnotTable**, consists in augmenting the part-of-speech tag with the table identifier(s) associated with the noun or the verb. For example, the verb *chérir* (*to cherish*) belongs to the table *12*. Therefore, the induced tag is *#tag_12*, where *#tag* is the POS tag associated with the verb. For an ambiguous verb like *sanctionner* (*to punish*), belonging to two tables *6* and *12*, the induced tag is *#tag_6_12*.

Then, in the case of verbs, we have done variants of the previous experiment by taking the hierarchy of verb tables into account. This hierarchy provides a tagset with a size which varies according to the level in the hierarchy. Identifiers added to tags depend on the verb and the specific level in the hierarchy. For example, the verb *sanctionner*, belonging to tables *6* and *12*, has a tag *#tag_QTD2* at level 1. In the case of ambiguous verbs, for a given level in the hierarchy, suffixes contain all classes the verb belongs to. This experiment will be called **AnnotVerbs** thereafter. In the case of verbal nouns, as such a hierarchy of tables does not exist, we experimented two other methods. The first one, called **AnnotIN**, consists in adding a suffix *IN* to the tag of a noun if this noun occurs in the syntactic lexicon, and therefore if it is a verbal noun. The second method, called **AnnotNouns**, consists in creating a hierarchy of noun tables from the table of classes of verbal nouns. This hierarchy is made accordingly to the maximum

[4] These resources are freely available at http://infolingu.univ-mlv.fr>Language_Resources> Lexicon_Grammar>Download

[5] The hierarchy of verb tables is available at : http://igm.univ-mlv.fr/∼sigogne/arbre-tables.xlsx

[6] We can also state that 3,121 verb forms (3,195 entries) are unambiguous. This means that all their entries occur in a single table.

number of arguments that a noun of a table can have according to defining properties specified for this table. As a consequence, the hierarchy has a single level. For example, nouns of the table *N_aa* can have at most 2 arguments contrary to those of table *N_an04* which can have only one. The characteristics of each level are specified in table 3[7] (level 0 represents the set of tables of the Lexicon-Grammar). We can state that there are 8,531 distinct nominal forms for 12,351 resulting entries in tables of nouns.

| Level | #classes | AVG_1 | AVG_2 |
|---|---|---|---|
| 0 | 76 | 162 | 1.43 |
| 1 | 3 | 3,413 | 1.2 |

Table 3: Characteristics of the hierarchy of noun tables

## 5 Experimental setup

For our experiments, we used the richest treebank for French, the French Treebank, (later called FTB) (Abeillé et al., 2003), containing 20,860 sentences and 540,648 words from the newspaper *Le Monde* (version of 2004). As this corpus is small, we used a *cross-validation* procedure for the evaluation. This method consists in splitting the corpus into *p* equal parts, then we compute training on *p-1* parts and evaluations on the remaining part. We can iterate this process *p* times. This allows us to calculate an average score for a sample as large as the initial corpus. In our case, we set the parameter *p* to 10. We also used the part-of-speech tagset defined in (Crabbé and Candito, 2008) containing 28 different tags describing some complementary morphological and syntactic features (e.g. verb mood, clitics, ...)[8]. Compound words have been merged in order to obtain a single token.

In the following experiments, we will test the impact of modifying the tagset of the training corpus, namely the addition of information from the Lexicon-Grammar described in the section 4. Results on evaluation parts are reported using the standard protocol called PARSEVAL (Black et al., 1991) for sentences smaller than 40 words. The score f-measure (F1) takes into account the bracketing and categories of nodes (including punctuation nodes). For each experiment, we have reported the *Baseline* results (i.e. the results of BKY trained on the original treebank without annotations from the Lexicon-Grammar). We have also indicated the percentage of distinct annotated verbs and verbal nouns in the entire corpus for each annotation method[9].

### 5.1 Annotation of verb tags

We first conducted experiments on verbs described in section 4, namely *AnnotTable* and *AnnotVerbs*. The experimental results are shown in the table 4. In the case of the method *AnnotVerbs*, we varied two parameters, *Lvl* (for Level) indicating the level of the hierarchy used and *Amb.* (for Ambiguity) indicating that a tag of a verb is changed only if this verb belongs to a number of classes less than or equal to the number specified by this parameter.

| Method | Lvl/Amb. | F1/Tagging | Absolute gains (F1) |
|---|---|---|---|
| Baseline | -/- | 85.05/97.43 | |
| AnnotTable | -/1 | 84.49/97.29 | |
| AnnotVerbs | 1/1 | 85.06/97.46 | |
| AnnotVerbs | 2/1 | 85.35/97.41 | |
| AnnotVerbs | 3/1 | **85.39/97.49** | |
| AnnotVerbs | 2/2 | 84.60/97.35 | |
| AnnotVerbs | 3/2 | 85.20/97.48 | |

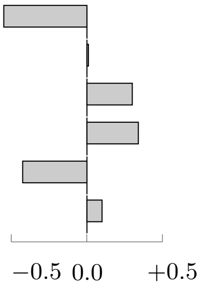


Table 4: Results from cross-validation evaluation according to verb annotation methods

| Method | Size of tagset | % annotated verbs |
|---|---|---|
| Baseline | 28 | - |
| AnnotTable | 228 | 18,6% |
| AnnotVerbs 1/1 | 89 | 21,5% |
| AnnotVerbs 2/1 | 76 | 22,5% |
| AnnotVerbs 3/1 | 47 | 33,9% |
| AnnotVerbs 2/2 | 246 | 44,7% |
| AnnotVerbs 3/2 | 75 | 55,7% |

Table 5: Size of tagset and percentage of annotated verbs according to verb annotation methods

For non-ambiguous verbs, we observe that the experiment *AnnotTable* highly deteriorates performances. This comes most probably from the

[7] The number of non-ambiguous nouns is 6126 for 6175 entries.

[8] There are 6 distinct tags for verbs and 2 distinct tags for nouns.

[9] The corpus contains 3058 distinct verbal forms and 17003 distinct nominal forms.

grammar which is too fragmented because of the significant size of the part-of-speech tagset (as shown in table 5). However, the effect is reversed as soon as we use levels of the hierarchy of tables (levels 2 and 3 only). The use of the table hierarchy causes the increase of the number of verbs annotated as non-ambiguous and the decrease of the size of the tagset. Considering ambiguous verbs do not improve performances (results are shown only for levels 2 and 3 with maximal ambiguity of 2) because of the large size of the tagset (as for experiment *AnnotTable*).

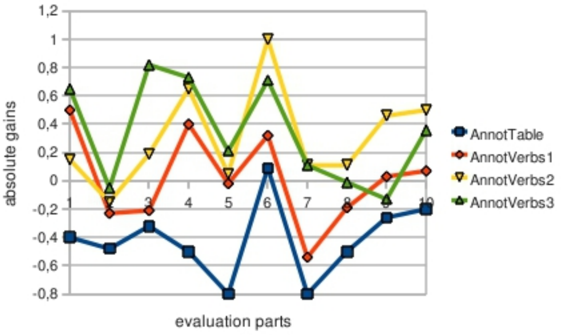


Figure 3: Absolute gains (F1) of verb annotation methods on evaluation parts (baseline is the horizontal line at 0 on y axis)

We can see on Figure 3 absolute gains according to verb annotation methods on evaluation parts. We have displayed curves for methods *AnnotTable* and *AnnotVerbs*X, where X is the level in the hierarchy (without ambiguity). Higher we are in the hierarchy of tables, the more we obtain better performances. Levels 2 and 3 are globally above the baseline for most of their evaluation parts. Therefore, this would mean that table identifiers of verbs and the hierarchy are a real help for parsing and do not produce a random effect. On table 6, we can see the top most error reductions according to phrase label, for the best verb annotation method (*AnnotVerbs* with level 3 of the hierarchy). For each phrase, the column called *Error reduction* indicates the average error reduction rate associated with the corresponding average number of error corrected (inside brackets). The NP and PP phrases are those that have the highest number of errors corrected (the low reduction rate can be explained by the fact that these two phrases have the highest number of errors). Furthermore, they are linked to each other because, generally, a PP has a NP kernel. Therefore, if a NP is corrected, the corresponding PP is also corrected (if it is the only error).

| Phrase label | Meaning | Error reduction |
|---|---|---|
| Ssub | subordinate clause | 5,3% (52) |
| Sint | internal clause | 3,6% (47) |
| PP | prepositional phrase | 3,1% (272) |
| Srel | relative clause | 2,2% (17) |
| NP | nominal phrase | 2,1% (347) |
| VPinf | infinitive phrase | 2,1% (34) |

Table 6: Top most error reductions according to phrase label

## 5.2 Annotation of noun tags

For verbal nouns, we successively conducted several experiments *AnnotTable*, *AnnotNouns* and *AnnotIN*, described in section 4. Results are given in table 7. As for verbs, we have reported the results for the experiment *AnnotNouns* with respect to the parameter *Ambiguity* (the maximum number of classes being associated with a noun is 3).

| Method | Amb. | F1/Tagging | Absolute gains (F1) |
|---|---|---|---|
| Baseline | - | 85.05/97.43 | |
| AnnotTable | 1 | 85.10/97.42 | |
| AnnotNouns | 1 | 85.13/97.48 | |
| AnnotNouns | 2 | 85.16/97.47 | |
| AnnotNouns | 3 | 85.05/97.41 | |
| AnnotIN | - | **85.20/97.54** | |
| | | | −0.5 0.0 +0.5 |

Table 7: Results from cross-validation evaluation according to noun annotation methods

| Method | Size of tagset | % annotated nouns |
|---|---|---|
| Baseline | 28 | - |
| AnnotTable | 98 | 8,6% |
| AnnotNouns 1 | 33 | 11,2% |
| AnnotNouns 2 | 38 | 16,5% |
| AnnotNouns 3 | 39 | 16,9% |
| AnnotIN | 30 | 16,9% |

Table 8: Size of tagset and percentage of annotated verbal nouns according to noun annotation methods

The various noun annotation methods slightly increase performances of the parser. Unlike verbs, the method *AnnotTable* does not degrade performances because there are much less nouns in the corpus belonging to the syntactic lexicon (less than 9% as shown in table 8), hence the limited impact of the new tagset. The use of a simple hierarchy of the noun tables, through experiment *AnnotNouns*, achieves positive gains but, here, insignificant. Moreover, we obtain a slight improvement by annotating some ambiguous nouns. Surprisingly, the method which gives the best result, despite its simplicity, is *AnnotIN*. We can see in Figure 4 absolute gains according to noun annotation methods on all evaluation parts. Unlike verbs, absolute gains are closer to the baseline. The best method *AnnotIN* is able to improve significantly 4 of 10 evaluation parts (+0,4 to +0,8).

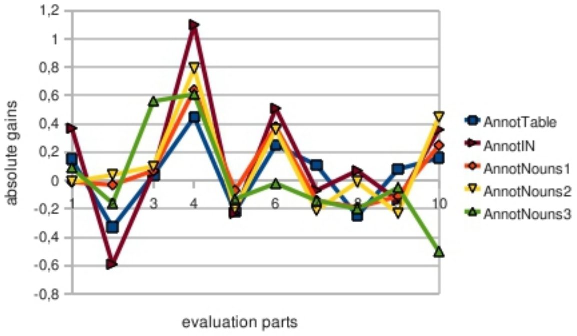


Figure 4: Absolute gains (F1) of noun annotation methods on evaluation parts (baseline is the horizontal line at 0 on y axis)

### 5.3 Combination of annotations

In a final experiment with BKY, we combined the best methods of verb and verbal noun annotations, that are *AnnotIN* for verbal nouns and *AnnotVerbs* for verbs (level 3 without ambiguity). Results are shown in table 9.

| Method | F1 |
|---|---|
| Baseline | 85.05 |
| Combination | **85.32** |

Table 9: Results from cross-validation evaluation according to combination of annotations

Combination of annotations does not increase the gains obtained with the method *AnnotVerbs* and we even observe a slight decrease.

### 5.4 Impact on a reranker

We also experimented the integration of a discriminative reranker (cf. section 2). We practically set to 10 the number of parses generated by BKY for each sentence (therefore, the 10 most probable parses). The following experiment consists in evaluating the impact of the modification of the tagset on a reranker. We called *Reranker(Baseline)* the experiment using the reranker with BKY trained on the original corpus (without annotations from the Lexicon-Grammar). *Reranker(AnnotVerbs)* is the experiment based on BKY that is trained on the corpus annotated by the best verb annotation method, *AnnotVerbs* (level 3 of hierarchy without ambiguity). Results are shown in table 10. The column named *Oracle F1/Tagging* indicates oracle scores for f-measure and tagging accuracy. An oracle score is the best global score that we could obtain whether we choose, for each input sentence, the best parse from the n-best parses. With this score, we can estimate the performance limit of a parser and the global quality of parses generated.

| Method | F1/Tagging | Oracle F1/Tagging |
|---|---|---|
| BKY(Baseline) | 85.05/97.43 | - |
| BKY(AnnotVerbs) | 85.39/97.49 | - |
| Reranker(Baseline) | 86.51/97.42 | 91,72/98.03 |
| Reranker(AnnotVerbs) | **86.71/97.49** | **91.99/98.08** |

Table 10: Results from cross-validation evaluation for reranking process.

First, we can see that *Reranker(Baseline)* improves performances with an absolute gain of +1,46 as compared with the baseline. These results are comparable to scores obtained for English (Charniak and Johnson, 2005). Then, we observe that the experiment *Reranker(AnnotVerbs)* increases the f-measure by +0,2 compared with *Reranker(Baseline)* (and to a lesser extent, the tagging accuracy by +0,07). The power of the discriminative model of the reranker implies that the gap of performances between the two experiments based on the reranker is less than the one obtained from experiments only based on BKY (+0,2 against +0,34). In addition, the oracle f-measure is improved (+0,27), which means that analyses generated by BKY are slightly better. We can see on Figure 5 absolute gains given by the reranker on all evaluation parts according to the two methods described above. Globally, the method *Reranker(AnnotVerbs)* has a curve slightly above the one of *Reranker(Baseline)*. Note that the first one outperforms the latter on 8 of 10 evaluation parts. All these observations confirm that the syntactic lexicon through the experiment *AnnotVerbs* is able to improve performances on both

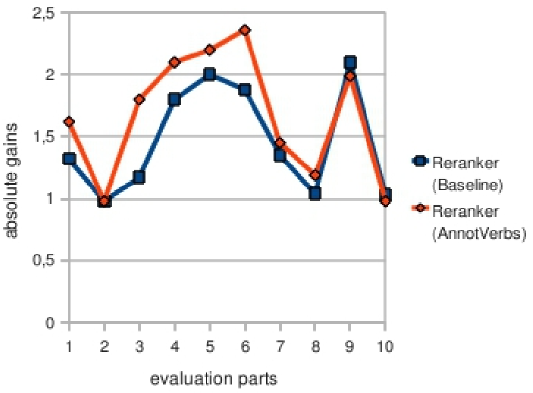


Figure 5: Absolute gains (F1) given by the reranker on evaluation parts (BKY(baseline) is the horizontal line at 0 on y axis)

a generative parser based on a PCFG grammar (BKY), and a discriminative parser (reranker).

## 6 Conclusions

The work described in this paper shows that by adding some information from a syntactic lexicon like the Lexicon-Grammar, we are able to improve performances of several probabilistic parsers. These performances are mainly obtained thanks to a hierarchy of verb tables that can limit ambiguity in terms of number of classes associated with a verb. This has the effect of increasing the coverage of verbs annotated according to the level of granularity used. However, once we include some ambiguity, performances drop. Results obtained on verbal nouns with a simple hierarchy of tables are insignificant but suggest a degree of progress with a more complex hierarchy as the one available for verbs.

In the near future, we plan to reproduce these experiments by taking into account of word clustering methods introduced by (Koo et al., 2008; Candito and Crabbé, 2009; Candito and Seddah, 2010). Thanks to a semi-supervized algorithm, these methods can reduce the size of the lexicon of the grammar by grouping words according to their behaviors in a treebank. These methods could be complementary to annotation methods described in this paper. Moreover, we plan to exploit the LFG formalism in order to use a syntactic lexicon more easily than for PCFGs, as many works have reported performance improvements for these models (Cahill, 2004; Deoskar, 2008).

## References

A. Abeillé, L. Clément, and F. Toussenel. 2003. Building a treebank for French. In Anne Abeillé, editor, *Treebanks*, Kluwer, Dordrecht.

E. Black, S.Abney, D. Flickinger, C. Gdaniec, R. Grishman, P. Harrison, D. Hindle, R. Ingria, F. Jelinek, J. Klavans, M. Liberman, M. Marcus, S. Roukos, B. Santorini, and T. Strzalkowski. 1991. A procedure for quantitatively comparing the syntactic coverage of english grammars. In *Proceedings of the DARPA Speech and Naturale Language Workshop*, pages 306–311.

T. Briscoe and J. Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Fifth Conference on Applied Natural Language Processing*, pages 356–363, Washington DC, USA.

A. Cahill. 2004. *Parsing with Automatically Acquired, Wide-Coverage, Robust, Probabilistic LFG Approximations*. Ph.D. thesis, Dublin City University, Dublin 9.

Marie Candito and B. Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technology (IWPT'09)*, pages 138–141.

Marie Candito and D. Seddah. 2010. Parsing word clusters. In *Proceedings of the first NAACL HLT Workshop on Morphologically-Rich Languages (SPRML2010)*, pages 76–84, Los Angeles, California.

J. Carroll and A. C. Fang. 2004. The automatic acquisition of verb subcategorisations and their impact on the performance of an HPSG parser. In *Proceedings of the 1st International Conference onNatural Language Processing*, Sanya City, China.

E. Charniak and M. Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics (ACL05)*.

M. Collins. 2000. Discriminative reranking for natural language parsing. In *Proceedings of the 17th ICML*, pages 175–182.

M. Constant and E. Tolone. 2008. A generic tool to generate a lexicon for NLP from Lexicon-Grammar tables. In *Actes du 27ème Colloque Lexique et Grammaire*, L'Aquila, Italie.

B. Crabbé and Marie Candito. 2008. Expériences d'analyse syntaxique statistique du français. In *Actes de la 15ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN'08)*, pages 45–54, Avignon, France.

E. de La Clergerie. 2010. Building factorized TAGs with meta-grammars. In *Proceedings of the 10th International Conference on Tree Adjoining Grammars and Related Formalisms*, pages 111–118.

T. Deoskar. 2008. Re-estimation of lexical parameters for treebank PCFGs. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 193–200, Manchester, Great Britain.

J. Dubois and F. Dubois-Charlier. 1997. *Les verbes français.* Larousse-Bordas.

K. Eynde and M. Piet. 2003. La valence: l'approche pronominale et son application au lexique verbal. *Journal of French Language studies*, pages 63–104.

M. Gross. 1994. Constructing Lexicon-grammars. In Atkins and Zampolli, editors, *Computational Approaches to the Lexicon*, pages 213–263.

A. K. Joshi. 1987. An introduction to tree adjoining grammars. In Alexis Manaster-Ramer, editor, *Mathematics of Language*, pages 87–115, Amsterdam/Philadelphia. John Benjamins Publishing Co.

R. Kaplan and J. Maxwell. 1994. *Grammar writer's workbench, version 2.0.* Rapport technique, Xerox Corporation.

T. Koo, X. Carreras, and M. Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08.*

T. Matsuzaki, Y. Miyao, and J. Tsujii. 2005. Probabilistic cfg with latent annotations. In *Proceedings of ACL-05*, pages 75–82, Ann Arbor, USA.

R. O'Donovan, A. Cahill, A. Way, M. Burke, and J. van Genabith. 2005. Large-scale induction and evaluation of lexical resources from the Penn-II and Penn-III treebanks. In *Proceedings of the First International Joint Conference on Natural Language Processing (IJCNLP'04).*

S. Petrov, L. Barrett, R. Thibaux, and D. Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia.

S. Riezler, T. King, R. Kaplan, R. Crouch, J. Maxwell, and M. Johnson. 2002. Parsing the Wall Street Journal using a lexical-functional grammar and discriminative estimation techniques. In *Proceedings of the Annual Meeting of the ACL*, University of Pennsylvania.

B. Sagot and E. Tolone. 2009. Intégrer les tables du Lexique-Grammaire à un analyseur syntaxique robuste à grande échelle. In *Actes de la 15ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN'09)*, Senlis, France.

B. Sagot. 2006. *Analyse automatique du français: lexiques, formalismes, analyseurs*. Ph.D. thesis, Université Paris VII.

B. Sagot. 2010. The lefff, a freely available, accurate and large-coverage lexicon for french. In *Proceedings of LREC 2010*, La Valette, Malte.

N. Schluter and J. Van Genabith. 2008. Treebank-based Acquisition of LFG Parsing Resources for French. In *Proceedings of LREC08*, Marrakech, Morocco.

D. Seddah, Marie Candito, and B. Crabbé. 2009. Adaptation de parsers statistiques lexicalisés pour le français : Une évaluation complète sur corpus arborés. In *Actes de la 15ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN'09)*, Senlis, France.

E. Tolone. 2011. *Analyse syntaxique à l'aide des tables du Lexique-Grammaire du français*. Ph.D. thesis, Université Paris-Est Marne-la-Vallée.