The Berkeley Parser at the EVALITA 2009 Constituency Parsing Task*

Alberto Lavelli[1] and Anna Corazza[2]

[1] FBK-irst,
via Sommarive 18, I-38123 Povo (TN), Italy
lavelli@fbk.eu
[2] Department of Physics, University "Federico II" of Naples,
via Cinthia, I-80126 Napoli, Italy
corazza@na.infn.it

Abstract. In this paper we describe our participation at the EVALITA 2009 Constituency Parsing Task. We used the Berkeley Parser, obtaining the best F_1, that is 78.73. This result corresponds to an increment of 15.85% with respect to the best result obtained at EVALITA 2007 by the Bikel's parser ($F_1 = 67.96$). A further important advantage of the Berkeley parser is that it does not require any language adaptation in addition to the need of retraining it on the new treebank. For comparison, we also report the results obtained by the Bikel's parser on the 2009 treebank.

Keywords: Constituency parsing, statistical parsing, Italian.

1 Introduction

Our participation at the EVALITA 2009 constituency parsing task is part of a wider research effort devoted to the application of state-of-the-art statistical parsing techniques to Italian (see [1] for preliminary outcomes of such an effort). Statistical parsers can be ported to new languages by retraining them on a treebank for the new language. Quite often, they also require some knowledge about the new language, such as rules for the choice of heads in the grammar. We therefore compared the different tools not only on performance, but also regarding the manual interventions necessary for the porting.

We started from the baseline obtained at the 2007 EVALITA competition [2], where we compared the Collins' parser [3], as implemented by Dan Bikel[3] [4], and the Stanford parser[4] [5, 6] (for more details on our participation at EVALITA 2007, see [7]). Adaptation of the Bikel's parser to the Turin University Treebank (TUT) included the identification of rules for finding lexical heads, and the selection of a lower threshold for unknown words (as the amount of available data is much lower). As we did not aim at introducing language-dependent adaptations, no tree transformations analogous

* We thank Dan Bikel and Slav Petrov for making available their parsers and for kindly answering our questions about their usage.

[3] http://www.cis.upenn.edu/~dbikel/#stat-parser

[4] http://nlp.stanford.edu/downloads/lex-parser.shtml

to those introduced by Collins for the PennTreeBank were applied to TUT. Regarding the Stanford parser adaptation to Italian, in the spirit of avoiding any language-specific adaptation, we only considered the basic available annotations, i.e., parent annotation for both nonterminals and PoS tags and horizontal Markovization (see [6] for details about the annotations). The head identification rules were the same as for the Bikel's parser.

For the 2009 participation, we started from the results obtained with the Bikel's parser at EVALITA 2007, reported in Table 1 [7]. During the development phase of EVALITA 2007, we compared the performance of the Bikel's parser and of the Stanford parser, and the Bikel's parser was chosen for performing the official run on the 2007 test set.

The first row in Table 1 reports the results obtained in the official EVALITA 2007 evaluation. It should be noted that 26 sentences could not be evaluated, due to misalignment errors (i.e., sentences having a different number of words in the gold standard and in the parser output). Such errors were caused by the presence of multi-word expressions, which are usually taken into account during preprocessing. After the gold standard was released, we have run further experiments with multi-word expressions represented as single tokens. Such results are reported in the second and third rows of Table 1. In the fourth and fifth rows we show the results of the Bikel's parser in the leave-one-out (LOO) experiment on the development set, both considering all sentences and considering only sentences with less than 40 words. In the same table, we also report performance on the test set for some significant configurations of the Stanford parser (both on all sentences and only on sentences with less than 40 words): (i) the treebank grammar without any transformation (Stanford TG); (ii) the configuration which obtained the best value of F_1 (i.e., with parent annotation, Stanford PA); (iii) the configuration which obtained the best Exact Match Rate (i.e., parent annotation on both nonterminals and PoS tags and hMarkov=2, Stanford best).

Table 1. EVALITA 2007: Results on TUT using Parseval measures (LR: labeled recall; LP: labeled precision) and Exact Match Rate (EMR).

	LR	LP	F_1	EMR
EVALITA 2007	70.81	63.35	67.96	
Bikel test	71.73	69.88	70.79	9.05
Bikel test < 40	72.04	70.08	71.05	9.84
Bikel LOO	73.42	72.49	72.95	18.43
Bikel LOO < 40	76.68	75.47	76.07	21.67
Stanford TG	54.15	60.41	57.11	3.50
Stanford TG < 40	56.15	62.35	59.09	4.37
Stanford PA	61.00	62.12	61.56	6.50
Stanford PA < 40	62.92	64.15	63.53	7.07
Stanford best	61.19	62.25	61.72	5.00
Stanford best < 40	63.03	64.23	63.62	5.43

The results on the test set clearly confirm our previous experiments on ISST [1], with the Bikel's parser outperforming the Stanford parser. Therefore, in 2009 edition we did not consider the Stanford parser.

2 Participation at EVALITA 2009

For the 2009 EVALITA edition we took into consideration a new parser, that is the Berkeley parser[5] [8], and compared its performance with the Bikel's parser, which obtained the best performance in the 2007 edition. As discussed above, in addition to performance we are also interested in the effort necessary to port the parser on a new language, that is Italian. Berkeley parser seemed extremely interesting from this point of view as it requires no additional effort apart from the availability of a treebank. Therefore, as soon as development data became available, in order to decide which of the two parsers had better performance on the TUT treebank at EVALITA 2009, we have run experiments with a 10-fold cross validation set up.

The Berkeley parser is based on a hierarchical coarse-to-fine parsing, where a sequence of grammars is considered, each being the refinement, namely a partial splitting, of the preceding one. Its performance is at the state of the art for English on the Penn Treebank and it outperforms other parsers in languages different from English, namely German and Chinese [8]. Indeed, a good compromise between efficiency and accuracy is obtained by a node splitting procedure, where splits which do not help accuracy are immediately pruned. Training is based on a discriminative framework, as discussed in [9]. As we aim at maximizing F_1, we used the parser version without reranking according to likelihood.

The Bikel's parser can be viewed in the framework of the lexicalized grammar approaches traditionally considered for probabilistic context-free grammars (PCFGs). Each parse tree is represented as the sequence of decisions corresponding to the head-centered, top-down derivation of the tree. Probabilities for each decision are conditioned on the lexical head.

For both parsers, we specialized the PUNCT PoS tag associated to punctuation to more specific PoS tags, similarly to what is done in the PennTreeBank annotation.

3 Results at EVALITA 2009

3.1 EVALITA 2009 dataset

The TUT version used in 2009 as training set consisted of 2,200 sentences, 1,100 taken from the Italian civil law and 1,100 taken from newspaper articles. The test set was composed by 200 sentences (100 civil law, 100 newspaper). The PoS tag set consists of 19 basic tags (68 including morphological features) and 29 nonterminal symbols.

As we did in 2007, we specialized the PUNCT PoS tag associated to punctuation to more specific PoS tags, similarly to what is done in the PennTreeBank annotation.

[5] `http://nlp.cs.berkeley.edu/Main.html#Parsing`

3.2 Experimental Results on the 2009 Training Set

First of all, we report the results obtained on the training set, used to choose the parser for performing the official run on the test set. As said above, two different parsers were compared, i.e. the Bikel's parser and the Berkeley parser. For the Berkeley parser the performance of the two grammars obtained after 4 or 5 refinement iterations are considered.

The chosen experimental set-up was 10-fold cross validation (using LOO as in 2007 was not a viable option because of the time needed by the Berkeley parser to perform training). Two different settings were compared with respect to the PoS tag set. The former employed all the 68 PoS tags (including morphological information) while the latter used only 19 basic PoS tags. The rationale of the latter setting was to reduce data sparsity. In Table 2 the results obtained performing the training of the parser using full PoS tags are shown, while in Table 3 the results obtained performing the training of the parser using basic PoS tags are displayed. In both cases the evaluation was done on the original treebank with full PoS tags (EVALB does not consider PoS accuracy when calculating Labeled Precision and Recall) and without considering punctuation.

Table 2. EVALITA 2009: Parser trained using full PoS tags and evaluated on the original treebank with full PoS tags. Results obtained using 10-fold cross validation on the training set.

	LR	LP	F_1	EMR
Bikel	72.37	71.27	71.82	16.29
Bikel < 40	76.03	74.63	75.32	19.83
Berkeley - iteration #5	75.94	75.58	75.76	23.13
Berkeley - iteration #5 < 40	79.70	79.18	79.44	28.09
Berkeley - iteration #4	73.99	74.74	74.37	18.10
Berkeley - iteration #4 < 40	76.72	77.39	77.05	21.83

Table 3. EVALITA 2009: Parser trained using basic PoS tags and evaluated on the original treebank with full PoS tags. Results obtained using 10-fold cross validation on the training set.

	LR	LP	F_1	EMR
Bikel	71.65	70.89	71.27	15.61
Bikel < 40	75.18	74.12	74.64	18.89
Berkeley - iteration #5	78.05	78.59	78.32	25.49
Berkeley - iteration #5 < 40	81.65	82.00	81.82	30.93
Berkeley - iteration #4	78.51	79.47	78.99	25.93
Berkeley - iteration #4 < 40	81.75	82.45	82.10	31.35

3.3 Experimental Results on the 2009 Test Set

The Berkeley parser with the grammar obtained at iteration #4 was chosen for performing the official run on the 2009 test set. In [10] the official results for the constituency parsing task can be found. Our system obtained the best result (F_1: 78.73; R: 80.02; P: 77.48). In Table 4 the results obtained on the test set by the Bikel's parser and by the Berkeley parser with grammars at different iterations are shown. The results confirm that the Berkeley parser outperforms the Bikel's parser. Note that in the official evaluation punctuation was taken into account and this greatly affected the performance of Bikel's parser (see Table 5 for results on the test set without considering punctuation).

Table 4. EVALITA 2009: Results obtained by the Bikel's parser and by the Berkeley parser on the test set.

	LR	LP	F_1	EMR
Bikel	68.51	64.45	66.42	14.00
Bikel < 40	68.99	65.03	66.95	14.81
Berkeley - iteration #5	79.60	76,63	78.09	21.50
Berkeley - iteration #5 < 40	79.53	76.94	78.21	22.75
Berkeley - iteration #4	80.02	77.48	78.73	21.00
Berkeley - iteration #4 < 40	79.90	77.92	78.90	22.22

Table 5. EVALITA 2009: Results obtained by the Bikel's parser and by the Berkeley parser on the test set without considering punctuation.

	LR	LP	F_1	EMR
Bikel	74.08	69.70	71.82	14.00
Bikel < 40	74.74	70.45	72.53	14.81
Berkeley - iteration #5	79.75	76,77	78.23	21.50
Berkeley - iteration #5 < 40	79.70	77.11	78.38	22.75
Berkeley - iteration #4	80.20	77.65	78.90	21.00
Berkeley - iteration #4 < 40	80.11	78.12	79.10	22.22

4 Conclusions

The results show that the Berkeley parser performs better than the Bikel's parser and moreover do not require any language-specific adaptation. This is in line with what reported in the [11] where different parsers are compared on French and the Berkeley parser wins over the other parsers.

We plan to keep working on improving parsing results on Italian, experimenting e.g. the Charniak reranking parser [12] and the use of self training both with reranking [13,

14] and without reranking [15]. Moreover, we would like to integrate the parser in the TextPro tool suite [16] to make it usable within other more complex systems (e.g., textual entailment, question answering, ...).

References

1. Corazza, A., Lavelli, A., Satta, G., Zanoli, R.: Analyzing an Italian treebank with state-of-the-art statistical parsers. In: Proceedings of the Third Workshop on Treebanks and Linguistic Theories (TLT 2004). Tübingen, Germany (2004)
2. Bosco, C., Mazzei, A., Lombardo, V., Attardi, G., Corazza, A., Lavelli, A., Lesmo, L., Satta, G., Simi, M.: Comparing italian parsers on a common treebank: the EVALITA experience. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation. Marrakech, Morocco (2008)
3. Collins, M.: Head-Driven Statistical Models for Natural Language Parsing. In: PhD thesis, University of Pennsylvania (1999)
4. Bikel, D.M.: Intricacies of Collins' parsing model. Computational Linguistics, vol. 30, issue 4, pp. 479–511 (2004)
5. Klein, D., Manning, C.D.: Fast exact inference with a factored model for natural language parsing. In: Advances in Neural Information Processing Systems 15 (NIPS 2002) (2002)
6. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics. Sapporo, Japan (2003)
7. Corazza, A., Lavelli, A., Satta, G.: Phrase-based statistical parsing. In: Proceedings of the EVALITA 2007 Workshop on Evaluation of NLP Tools for Italian (2007)
8. Petrov, S., Klein, D.: Improved inference for unlexicalized parsing. In: Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics, pp. 404–411. Rochester, New York (2007)
9. Petrov, S., Klein, D.: Discriminative log-linear grammars with latent variables. In: Proceedings of NIPS 2008 (2008)
10. Bosco, C., Mazzei, A., Lombardo, V.: Evalita'09 Parsing Task: constituency parsers and the Penn format for Italian. In: Proceedings of EVALITA 2009 (2009)
11. Seddah, D., Candito, M., Crabbé, B.: Cross parsers evaluation : a French treebanks study. In: Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09). Paris, France (2009)
12. Charniak, E., Johnson, M.: Coarse-to-fine n-best parsing and maxent discriminative reranking. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), pp. 173–180. Ann Arbor, Michigan (2005)
13. McClosky, D., Charniak, E., Johnson, M.: Effective self-training for parsing. In: Proceedings of the Human Language Technology Conference of the NAACL, Main Conference, pp. 152–159. New York City, USA (2006)
14. McClosky, D., Charniak, E., Johnson, M.: Reranking and self-training for parser adaptation. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. Sydney, Australia (2006)
15. Reichart, R., Rappoport, A.: Self-training for enhancement and domain adaptation of statistical parsers trained on small datasets. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 616–623. Prague, Czech Republic (2007)
16. Pianta, E., Girardi, C., Zanoli, R.: The TextPro tool suite. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation. Marrakech, Morocco (2008)