Analyzing the CoNLL–X Shared Task from a Sentence Accuracy Perspective

Analizando la CoNLL–X Shared Task con Medidas Basadas en Precisión por Frase

Miguel Ballesteros†, Jesús Herrera†, Virginia Francisco‡, Pablo Gervás‡
†Departamento de Ingeniería del Software e Inteligencia Artificial
‡Instituto de Tecnología del Conocimiento
Universidad Complutense de Madrid
C/ Profesor José García Santesmases s/n, Madrid, Spain
{miballes, jesus.herrera, virginia}@fdi.ucm.es, pgervas@sip.ucm.es

Resumen: Hoy en día, dada la relevancia de las CoNLL shared tasks para Análisis de Dependencias, las medidas más usadas son las que allí se computaron. Esas medidas, están basadas en calcular globalmente la precisión palabra por palabra (o token por token) para todo el conjunto de frases. En nuestra opinión el usuario final de un analizador de dependencias podría esperar una precisión local basada en evaluar la precisión frase a frase. En estos casos, unas medidas diferentes pueden añadir algo de información que podría ser relevante acerca de que analizador devuelve un mejor resultado. Es por ello que presentamos el estudio de este artículo con la intención de enriquecer la descripción del comportamiento de los analizadores de dependencias.
Palabras clave: Análisis sintáctico de dependencias, CoNLL Shared Tasks, Precisión por frase.

Abstract: Nowadays, because of the relevance of the CoNLL shared tasks on Dependency Parsing, the most used evaluation measures are the ones computed in them. These measures, which are token–based, are computed globally for a whole big set of texts considering token by token. But a final user of a dependency parser would expect a high and stable accuracy for every parsed piece of text (usually one sentence). In this cases sentence–based measures add some information that could be relevant. This is why we developed the present study, which is addressed to get a richer description of the performance of dependency parsers.
Keywords: Dependency parsing, CoNLL–X Shared Task, Sentence Accuracy

1 Introduction

In the CoNLL shared tasks on Dependency Parsing the following token–based evaluation measures were computed (Buchholz and Marsi, 2006): LAS (Labelled Attachment Score), UAS (Unlabelled Attachment Score) and LA (Label Accuracy). Since these tasks on Dependency Parsing have been very relevant in the area, now this set of measures has become a *de facto* standard when evaluating dependency parsers. Although Yamada and Matsumoto (2003) proposed a sentence–based measure, described in their work as *Complete Rate* measure, moreover, some recent works have used complete match measures to evaluate, such as Goldberg and Elhadad (2010). Therefore, our work aims to attract attention to sentence–based measures, as a way to get a richer description of the performance of dependency parsers combining them with token–based measures. To this end, we reevaluated the participation of the 19 parsers in the CoNLL–X Shared Task by computing a pair of sentence–based measures over its 13 test corpora.

2 Background

The CoNLL–X Shared Task was the first of a series of evaluation campaigns devoted to Dependency Parsing. We took the material for developing the present work from that first task, so we give a brief outline of it.

2.1 The CoNLL–X Shared Task

Every year the CoNLL conference features a shared task. The 10th edition was devoted to Multilingual Syntactic Dependency parsing. The aim of this task was to extend the state–of–the–art available at that time in Dependency Parsing. Participants were asked to label dependency structures by means of fully automatic dependency parsers. This Shared Task provided a benchmark for evaluating the participating parsers accross 13 languages. Systems were scored with the following token–based measures: LAS, UAS and LA.

For the purposes of the Shared Task 13 annotated source corpora, one for each proposed language, were provided. We used all of them to develop our experiment: **Arabic** (Hajič and Zemánek, 2004), **Czech** (Böhmová et al., 2001), **Danish** (Kromann, 2003), **Slovene** (Džeroski et al., 2006), **Swedish** (Nilsson et al., 2005), **Turkish** (Oflazer et al., 2003), **Chinese** (Chen et al., 2003), **Dutch** (van der Beek et al., 2002), **German** (Brants et al., 2002), **Japanese** (Kawata and Bartels, 2000), **Portuguese** (Afonso et al., 2002), **Bulgarian** (Simov et al., 2005) and **Spanish** (Palomar et al., 2004). In Table 1 we show the sizes of the training corpora.

The following authors presented parsers to the Shared Task: Attardi (2006), Bick (2006), Canisius (2006), Carreras (2006), Chang et al. (2006), Cheng et al. (2006), Corston-Oliver and Aue (2006), Dreyer et al. (2006), Johansson and Nugues (2006), Liu et al. (2006), McDonald et al. (2006), Nivre et al. (2006), Riedel's (2006), Schiehlen's (2006), Shimizu's (2006), Yuret (2006), Wu et al. (2006). O'Neil and Sagae did not publish their papers, but their results were computed in the Shared Task and are computed in the present work.

2.2 Evaluation Measures

One way to evaluate dependency parsers is to consider parsed texts as sets of wordforms (tokens) and to compute how many tokens are correctly attached, labelled or both things at the same time. Thus we have measures such as Labelled Attachment Score (LAS), Unlabelled Attachment Score (UAS) and Label Accuracy (LA). These were used for evaluation in the CoNLL Shared Tasks (Buchholz and Marsi, 2006; Nivre et al., 2007) on Dependency Parsing. This set of measures is known as token–based measures.

But there are also measures that consider parsed texts as sets of sentences. These measures can take into account either the whole unlabelled graph (only links between wordforms) or the whole labelled graph (links and labels), for every sentence in the test set. We also consider macro-averaging attachment scores over sentences that seem to be a more informative measure. Since the Shared Task provided labelled parsing, we consider the following evaluation measures:

- Macro–Average LAS (MacroLas) is the percentage of "scoring" tokens in the test set with correct attachment and labelling averaged per sentence.
- Labelled Complete–Match (LCM) is the percentage of sentences in the test set with correct labelled graph.

3 Why do we Think that Sentence Accuracy Measures should be Considered?

Nowadays[1] dependency parsers usually show a high overall parsing accuracy when evaluated for LAS, UAS or LA. This means that a high percentage of the processed tokens are correctly linked and/or these links are correctly labelled. But all these tokens pertain to different sentences and generally speaking, only a small percentage of these sentences is actually parsed without any errors. So high values of LAS, UAS and LA mean a high performance from a computational point of view. Nonetheless, the unit of language with proper meaning is the sentence. Then, a human end user eventually would prefer a high percentage of sentences parsed without errors (and a small percentage with several errors), rather than one or two errors for each parsed sentence. Then, the more sentences without errors the more usefulness for a human end user. Under these considerations, sentence–based measures should be considered to add more information to the peformance of dependency parsers.

Therefore, the reasons given above led us to study the enrichment of token–based evaluation processes with sentence–based mea-

[1]Some parsers presented at the Shared Task are constantly renovated, for instance, the last version of MaltParser is dated March 2012

	Arab	Bulg	Chin	Czech	Dan	Dutch	Germ	Jap	Port	Slov	Span	Swed	Turk
#Sentences	1,479	12,823	57,333	72,703	5,190	13,349	39,216	17,044	9,071	1,534	3,306	11,042	4,997
#Tokens	54,379	190,217	338,897	1,249,408	94,386	195,069	699,610	151,461	206,678	28,750	89,334	191,467	57,510
Av.S	37.2	14.8	5.9	17.2	18.2	14.6	17.8	8.9	22.8	18.7	27.0	17.3	11.5

Table 1: Number of sentences and wordforms of each training corpus of the CoNLL–X Shared Task. Av.S means average sentence length.

sures. This is why we developed the reevaluation described in the present Work.

4 Reevaluating the parsers of the CoNLL–X Shared Task with Sentence–Based Measures

To illustrate our proposal we reevaluated the participations of all CoNLL–X systems[2] computing sentence–based measures. Then, we evaluated each parser by computing MacroLAS and LCM for each test set provided in the Shared Task. The results of this reevaluation are shown in the Tables 2 and 3.

The results for LCM are normally around 30%, but we must take into account the difficult task that is to annotate sentences that could contain an important number of tokens combined in very different syntactic structures.

MSTParser (McDonald's) and MaltParser (Nivre's) results were really close and the best in the Shared Task. McDonald's parser is the best when considering MacroLAS measure due to the MSTParser's accuracy predicting arcs, but Nivre's parser is the best when considering LCM due to the better accuracy predicting dependency labels, as shown in (McDonald and Nivre, 2011). Again, Nivre's and McDonald's systems are the best, and the MacroLAS results demonstrate that they are really accurate when measuring the results sentence by sentence. Nevertheless, it seems that Nivre's parser could be considered a bit better because the differences are wider, more than 2 percentage points, in favour of this parser when considering LCM and the results for MacroLAS are only 0.3 percentage points worse.

Besides that, it is important to remark that the results with MaltParser and MSTParser are similar considering LCM and MacroLAS and they follow a very similar behaviour for every language. Therefore, it can be concluded that both trends on data–driven dependency parsers are accurate and eligible for parsing complex syntactic purposes. Note that the MacroLAS results are quite similar to the LAS results published in the Shared Task, nonetheless, the parsers that showed better behavior in the Shared Task, obtain much better MacroLAS data and the parsers that showed worse results in the Shared Task obtain much worse results for MacroLAS. It is quite obvious that MacroLAS will yield results close to LAS, since both are averaging the number of correct labelled attachments.

Longer sentences are an interesting issue to tackle because most of the parsers show difficulties parsing them, as it is mentioned in (McDonald and Nivre, 2011), which means that the results for languages with a longer average sentence length are directly affected by this fact. Most testing data–sets contain sentences of very different lengths, with the exception of Japanese and Chinese, in which the average sentence length is really small and most of the sentences are similar in terms of sentence length. Thus, it is also important to take into account that the languages with a shorter average sentence length in the testing data set are the ones with a higher LCM after parsing. For instance, the average sentence length for Chinese is 5.78 words and LCM is 49.58. For Arabic, the average sentence length is 36.80 words and LCM is 6.24. In the Figure 1 we show the correlation between average sentence length and LCM that corroborates the strong correlation between sentence–based measures and the average sentence length. Table 1 shows the average sentence length in each corpus.

Besides that, it seems that there are some remarkable differences between models trained with corpora that contain sentences in the same average sentence length, for instance, models trained with the Slovene corpus (18.7 average sentence length) and German corpus (17.8 average sentence length) produced very different results, but it can be explained over the training corpus size of Slovene (29k tokens) and German (700k to-

[2]Using the outputs published in the CoNLL–X Shared Task website

Parser	Arab	Bulg	Chin	Czech	Dan	Dutch	Germ	Japa	Port	Slov	Span	Swed	Turk	Tot
McD.	**71.11**	88.29	88.40	**82.24**	85.95	80.35	**89.13**	95.43	87.63	**75.96**	**83.58**	85.33	75.06	**83.73**
Niv.	70.33	88.61	89.56	79.87	**86.38**	**80.96**	88.08	**96.06**	**88.45**	71.02	82.94	**86.64**	**76.25**	83.47
O'N.	71.06	**86.63**	89.50	78.74	83.95	79.16	87.87	95.42	85.69	73.91	81.87	84.50	70.23	82.19
Che.$^{\uparrow}$	69.89	87.47	87.51	78.14	83.55	74.59	86.70	95.07	85.74	73.90	81.47	83.74	73.74	81.65
Rie.$^{\downarrow}$	70.80	–	**92.13**	70.77	85.26	79.39	88.62	95.40	85.37	74.25	79.17	83.26	71.03	81.29
Sag.$^{\downarrow}$	67.47	–	87.60	78.83	83.99	77.73	87.19	95.28	87.06	72.84	78.40	84.45	74.60	81.29
Cor.	68.33	84.48	83.05	77.08	82.54	73.90	85.33	95.12	85.63	75.14	82.40	82.37	73.13	80.65
Car.$^{\uparrow}$	65.72	84.23	86.76	71.62	81.07	70.34	84.33	94.18	84.13	71.04	79.20	81.40	70.36	78.80
Cha.$^{\downarrow}$	58.56	–	87.49	69.00	81.13	75.38	86.53	94.73	82.19	71.31	80.62	84.37	71.97	78.61
Wu.$^{\uparrow}$	67.34	81.40	78.47	54.82	79.59	73.16	79.95	95.25	82.31	70.05	73.50	75.72	67.77	75.33
Bic.$^{\uparrow}$	58.58	80.36	80.56	66.07	76.79	72.32	76.63	92.11	76.20	66.49	73.36	77.44	66.30	74.09
Can.	53.57	79.93	83.93	56.20	79.93	77.40	81.73	93.64	74.08	57.43	68.67	81.62	65.36	73.35
Shi.$^{\uparrow}$	67.30	–	–	–	76.94	–	–	–	–	66.33	74.84	82.10	67.13	72.44
Joh.$^{\downarrow}$	68.68	–	74.29	71.50	81.87	74.59	81.17	87.26	84.01	68.15	76.39	78.51	72.82	70.71
Liu.$^{\uparrow}$	56.66	69.00	80.00	61.31	80.34	63.91	72.60	84.68	72.28	60.12	66.49	67.96	53.17	68.34
Yur.$^{\downarrow}$	49.36	75.04	78.09	47.31	73.50	69.30	67.85	92.17	66.49	53.27	71.01	68.96	71.85	68.02
Sch.	43.14	–	71.66	51.47	76.87	72.44	72.26	91.59	66.55	49.00	48.34	74.52	61.98	64.99
Dre.$^{\downarrow}$	53.95	74.56	76.36	62.91	66.78	66.36	73.34	91.09	74.63	61.53	66.88	68.76	56.47	63.83
Att.$^{\downarrow}$	50.12	70.06	51.60	55.25	64.90	49.37	66.45	44.07	72.20	56.33	65.48	63.84	44.65	58.02
Av	62.21	80.77	81.50	67.40	79.54	72.81	80.88	**90.48**	80.04	66.74	74.45	78.71	67.57	*74.78*

Table 2: Results of the CoNLL–X Shared Task for Macro–Average LAS (MacroLAS). The arrows show the reclassification when considering MacroLAS compared with the LAS results published in the Shared Task.

Parser	Arab	Bulg	Chin	Czech	Dan	Dutch	Germ	Japa	Port	Slov	Span	Swed	Turk	Tot
Niv$^{\uparrow}$	9.59	**32.91**	68.05	27.12	**26.09**	**27.46**	34.73	**75.32**	**31.60**	18.41	**17.96**	**32.13**	19.26	**32.36**
McD.$^{\downarrow}$	9.59	30.15	62.51	**27.95**	24.22	25.91	34.73	72.92	23.96	18.91	17.48	27.76	**19.42**	30.42
Sag.$^{\uparrow}$	8.22	–	61.25	23.01	23.91	23.06	**36.69**	71.23	27.78	**20.40**	12.62	27.76	19.10	29.59
Che.$^{\uparrow}$	9.59	29.15	59.63	23.01	20.19	19.43	32.49	71.51	20.83	18.91	14.08	26.48	17.50	27.91
Rie.$^{\downarrow}$	9.59	–	**72.09**	13.42	21.12	22.28	32.49	71.65	21.53	13.93	10.19	23.91	13.80	27.17
O'N.$^{\downarrow}$	9.59	26.63	62.63	20.55	18.94	21.50	31.93	71.79	21.18	15.17	11.17	25.96	13.32	26.95
Cor.	**10.27**	23.87	46.83	20.82	16.15	18.65	28.85	71.79	22.57	18.16	15.05	24.16	15.25	25.57
Cha.	2.74	–	61.59	1.64	17.39	19.95	34.17	71.51	19.10	5.72	15.05	27.25	14.44	24.21
Car.$^{\uparrow}$	8.22	20.10	58.71	17.26	15.22	17.36	25.21	67.70	19.79	14.68	15.53	20.82	13.80	24.18
Wu.$^{\uparrow}$	8.22	23.62	47.29	0.00	13.35	17.88	24.37	72.21	21.18	13.43	7.77	14.91	11.71	21.23
Bic.$^{\uparrow}$	8.22	13.82	43.83	11.51	10.87	18.13	17.37	62.20	4.51	7.71	9.22	16.97	10.11	18.04
Can.	0.00	14.07	46.25	0.00	12.11	18.91	22.97	65.73	0.00	0.00	4.85	18.25	10.11	16.40
Joh.$^{\downarrow}$	8.22	–	33.10	7.94	11.94	15.80	16.25	50.63	14.58	7.96	6.31	14.40	9.47	16.38
Liu.$^{\uparrow}$	7.53	9.30	42.10	9.59	12.11	13.99	14.29	53.74	7.99	5.22	4.85	13.11	5.62	15.34
Yur.	0.00	10.55	44.87	0.00	9.32	16.58	12.32	63.47	0.00	0.00	5.34	11.31	14.44	14.48
Dre.$^{\downarrow}$	0.00	5.28	39.10	8.77	0.62	14.51	12.89	59.80	6.25	5.47	2.42	7.71	4.17	12.84
Shi.$^{\uparrow}$	8.90	–	–	–	12.11	–	–	–	–	8.21	6.31	23.91	9.15	11.43
Sch.$^{\uparrow}$	0.00	–	40.72	0.00	3.73	13.73	8.12	0.28	0.00	0.00	0.00	9.51	0.00	6.34
Att.$^{\downarrow}$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Av	6.24	18.42	49.48	11.81	14.18	18.06	23.33	**59.64**	14.60	10.12	9.27	19.28	11.61	*20.04*

Table 3: Results of the CoNLL–X Shared Task for Labelled Complete Match (LCM). The arrows show the reclassification when considering LCM compared with the LAS results published in the Shared Task.

kens). Moreover, Czech and German produced similar differences, in this case the Czech corpus is really big (1,249k tokens), but this situation can be explained due to the complexities of the Czech language, such as word–order or irregular grammar, which is a well known issue in dependency parsing.

5 *Conclusions*

As shown in Section 3 and taking into account the results discussed in Section 4, the use of sentence–based measures might give another view on the following question: which dependency parser is better? Considering only token scores the answer may not be enough in some cases, where the user could want to know if a Complete–Match accuracy (or close to complete) can be expected or not.

In summation, it is clear that these measures might be considered when we need a high accuracy per sentence and it is normally needed for a task in which the potential usefulness of dependency parsing is required. We believe that this study shows the importance of sentence accuracy analysis and we would like to aim researchers to show the results and data considering them in order to be able to study the accuracy in a deeper way and tak-

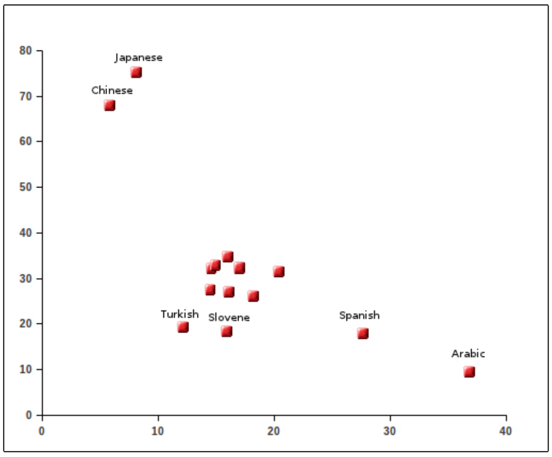

Figure 1: Correlation between Average Sentence Length (in the testing data–sets) and the LCM measure when parsed by MaltParser.

ing into consideration all the facts that are involved.

It is worth to mention, that the reclassification of the parsers is wider for LCM than for MacroLAS, as shown in Tables 2 and 3. Therefore, some parsers have difficulties parsing whole sentences, for instance, Attardi's parser is not able to parse correctly any of the sentences and this knowledge is more than useful when we need to select a parser as a tool to address a task.

Finally, taking into consideration the sentence length factors exposed in the previous section, it is also important to make the results directly comparable by building testing data–sets that contain sentences of the same average sentence length and not only containing a similar number of tokens. Moreover, this fact also affects token–based measures because one of the most frequent reasons of errors are due to the dependency length, but it is more evidenced when measuring with LCM, which shows again how sentence–based measures provide non–redundant information.

Acknowledgments

This research is funded by the Spanish Ministry of Education and Science (TIN2009-14659-C03-01 Project), Universidad Complutense de Madrid and Banco Santander Central Hispano (GR58/08 Research Group Grant).

References

Susana Afonso, Eckhard Bick, Renato Haber, and Diana Santos. 2002. Floresta sintá(c)tica: A treebank for Portuguese. In *LREC*.

Giuseppe Attardi. 2006. Experiments with a multilanguage non-projective dependency parser. In *CoNLL-X*.

Eckhard Bick. 2006. Lingpars, a linguistically inspired, language-independent machine learner for dependency treebanks. In *CoNLL-X*.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2001. The PDT: a 3-level annotation scenario. chapter 7. Kluwer Academic Publishers.

Sabine Brants, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The tiger treebank. In *TLT*.

Sabine Buchholz and Erwin Marsi. 2006. Conll-x shared task on multilingual dependency parsing. In *CoNLL-X*.

Sander Canisius, Toine Bogers, Antal Van Den Bosch, and Jeroen Geertzen. 2006. Dependency parsing by inference over high-recall dependency predictions. In *CoNLL-X*.

Xavier Carreras, Mihai Surdeanu, and Lluís Màrquez. 2006. Projective dependency parsing with perceptron. In *CoNLL-X*.

Ming Wei Chang, Quang Do, and Dan Roth. 2006. A pipeline model for bottom-up dependency parsing. In *CoNLL-X*.

Keh-Jiann Chen, Chi-Ching Luo, Ming-Chung Chang, Feng-Yi Chen, Chao-Jan Chen, Chu-Ren Huang, and Zhao-Ming Gao. 2003. Sinica treebank: Design criteria, representational issues and implementation.

Yuchang Cheng, Masayuki Asahara, and Yuji Matsumoto. 2006. Multi-lingual dependency parsing at NAIST. In *CoNLL-X*.

Simon Corston-Oliver and Anthony Aue. 2006. Dependency parsing with reference to slovene, spanish and swedish. In *CoNLL-X*.

Markus Dreyer, David A. Smith, and Noah A. Smith. 2006. Vine parsing and minimum risk reranking for speed and precision. In *CoNLL-X*.

Sasŏ Džeroski, Tomaž Erjavec, Nina Ledinek, Petr Pajas, Zdenek Žabokrtsky, and Andreja Žele. 2006. Towards a Slovene dependency treebank. In *LREC*.

Yoav Goldberg and Michael Elhadad. 2010. An efficient algorithm for easy-first non-directional dependency parsing. In *NAACL*.

Jan Hajič and Petr Zemánek. 2004. Prague arabic dependency treebank: Development in data and tools. In *Proc. of the NEMLAR Intern. Conf. on Arabic Language Resources and Tools*.

Richard Johansson and Pierre Nugues. 2006. Investigating Multilingual Dependency Parsing. In *CoNLL–X*.

Yasuhiro Kawata and Julia Bartels. 2000. Stylebook for the Japanese treebank in VERBMOBIL. Verbmobil-Report 240, Seminar für Sprachwissenschaft, Universität Tübingen.

Matthias T. Kromann. 2003. The Danish dependency treebank and the underlying linguistic theory. In Joakim Nivre and Erhard Hinrichs, editors, *TLT2*, Växjö, Sweden.

Ting Liu, Jinshan Ma, Huijia Zhu, and Sheng Li. 2006. Dependency parsing based on dynamic local optimization. In *CoNLL-X*.

Ryan McDonald and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. *Computational Linguistics*.

Ryan McDonald, K. Lerman, and Fernando Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *CoNLL–X*.

Jens Nilsson, Johan Hall, and Joakim Nivre. 2005. MAMBA meets TIGER: Reconstructing a Swedish treebank from antiquity. In *NODALIDA Special Session on Treebanks*.

Joakim Nivre, Johan Hall, Jens Nilsson, G. Eryiğit, and S. Marinov. 2006. Labeled pseudo–projective dependency parsing with support vector machines. In *CoNLL–X*.

Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *CoNLL 2007*.

Kemal Oflazer, Bilge Say, Dilek Zeynep Hakkani-Tür, and Gökhan Tür. 2003. Building a Turkish treebank.

Manuel Palomar, Montserrat Civit, Arantza Díaz, Lidia Moreno, Empar Bisbal, Marí J. Aranzabe, Alicia M. Ageno, M$^{\text{a}}$ Antonia Martí Martí, and Borja Navarro. 2004. 3lb: Construcción de una base de datos de árboles sintáctico–semánticos para el catalán, euskera y español. In *SEPLN*.

Sebastian Riedel, Ruket Çakici, and Ivan Meza-Ruiz. 2006. Multi-lingual dependency parsing with incremental integer linear programming. In *CoNLL-X*.

Michael Schiehlen and Kristina Spranger. 2006. Language independent probabilistic context-free parsing bolstered by machine learning. In *CoNLL-X*.

Nobuyuki Shimizu. 2006. Maximum spanning tree algorithm for non-projective labeled dependency parsing. In *CoNLL-X*.

Kiril Simov, Petya Osenova, Alexander Simov, and Milen Kouylekov. 2005. Design and implementation of the Bulgarian HPSG-based treebank. *Journal of Research on Language and Computation*.

Leonoor van der Beek, Gosse Bouma, Robert Malouf, and Gertjan van Noord. 2002. The Alpino dependency treebank. In *Computational Linguistics in the Netherlands (CLIN)*.

Yu-Chieh Wu, Yue-Shi Lee, and Jie-Chi Yang. 2006. The Exploration of Deterministic and Efficient Dependency Parsing. In *CoNLL–X*.

Hiroyasu Yamada and Yuji Matsumoto. 2003. Statistical dependency analysis with support vector machines. In *IWPT*.

Deniz Yuret. 2006. Dependency parsing as a classification problem. In *CoNLL-X*.