

# Survey on parsing three dependency representations for English

Angelina Ivanova    Stephan Oepen    Lilja Øvrelid

University of Oslo, Department of Informatics  
{angelii|oe|liljao}@ifi.uio.no

## Abstract

In this paper we focus on practical issues of data representation for dependency parsing. We carry out an experimental comparison of (a) three syntactic dependency schemes; (b) three data-driven dependency parsers; and (c) the influence of two different approaches to lexical category disambiguation (aka tagging) prior to parsing. Comparing parsing accuracies in various setups, we study the interactions of these three aspects and analyze which configurations are easier to learn for a dependency parser.

## 1 Introduction

Dependency parsing is one of the mainstream research areas in natural language processing. Dependency representations are useful for a number of NLP applications, for example, machine translation (Ding and Palmer, 2005), information extraction (Yakushiji et al., 2006), analysis of typologically diverse languages (Bunt et al., 2010) and parser stacking (Øvrelid et al., 2009). There were several shared tasks organized on dependency parsing (CoNLL 2006–2007) and labeled dependencies (CoNLL 2008–2009) and there were a number of attempts to compare various dependencies intrinsically, e.g. (Miyao et al., 2007), and extrinsically, e.g. (Wu et al., 2012).

In this paper we focus on practical issues of data representation for dependency parsing. The central aspects of our discussion are (a) three dependency formats: two ‘classic’ representations for dependency parsing, namely, *Stanford Basic* (SB) and *CoNLL Syntactic Dependencies* (CD), and bilinear dependencies from the HPSG English Resource Grammar (ERG), so-called *DELPH-IN Syntactic Derivation Tree* (DT), proposed recently by Ivanova et al. (2012); (b) three state-of-the-art statistical parsers: Malt (Nivre et al., 2007), MST

(McDonald et al., 2005) and the parser of Bohnet and Nivre (2012); (c) two approaches to word-category disambiguation, e.g. exploiting common PTB tags and using supertags (i.e. specialized ERG lexical types).

We parse the formats and compare accuracies in all configurations in order to determine how parsers, dependency representations and grammatical tagging methods interact with each other in application to automatic syntactic analysis.

SB and CD are derived automatically from phrase structures of Penn Treebank to accommodate the needs of fast and accurate dependency parsing, whereas DT is rooted in the formal grammar theory HPSG and is independent from any specific treebank. For DT we gain more expressivity from the underlying linguistic theory, which challenges parsing with statistical tools. The structural analysis of the schemes in Ivanova et al. (2012) leads to the hypothesis that CD and DT are more similar to each other than SB to DT. We recompute similarities on a larger treebank and check whether parsing results reflect them.

The paper has the following structure: an overview of related work is presented in Section 2; treebanks, tagsets, dependency schemes and parsers used in the experiments are introduced in Section 3; analysis of parsing results is discussed in Section 4; conclusions and future work are outlined in Section 5.

## 2 Related work

Schwartz et al. (2012) investigate which dependency representations of several syntactic structures are easier to parse with supervised versions of the Klein and Manning (2004) parser, ClearParser (Choi and Nicolov, 2009), MST Parser, Malt and the Easy First Non-directional parser (Goldberg and Elhadad, 2010). The results imply that all parsers consistently perform better when (a) coordination has one of the conjuncts as the head rather than the coordinating conjunction;

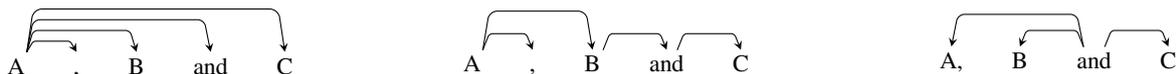


Figure 1: Annotation of coordination structure in SB, CD and DT (left to right) dependency formats

(b) the noun phrase is headed by the noun rather than by determiner; (c) prepositions or subordinating conjunctions, rather than their NP or clause arguments, serve as the head in prepositional phrase or subordinated clauses. Therefore we can expect (a) Malt and MST to have fewer errors on coordination structures parsing SB and CD than parsing DT, because SB and CD choose the first conjunct as the head and DT chooses the coordinating conjunction as the head; (b,c) no significant differences for the errors on noun and prepositional phrases, because all three schemes have the noun as the head of the noun phrase and the preposition as the head of the prepositional phrase.

Miwa et al. (2010) present intrinsic and extrinsic (event-extraction task) evaluation of six parsers (GDep, Bikel, Stanford, Charniak-Johnson, C&C and Enju parser) on three dependency formats (Stanford Dependencies, CoNLL-X, and Enju PAS). Intrinsic evaluation results show that all parsers have the highest accuracies with the CoNLL-X format.

### 3 Data and software

#### 3.1 Treebanks

For the experiments in this paper we used the **Penn Treebank** (Marcus et al., 1993) and the **DeepBank** (Flickinger et al., 2012). The latter is comprised of roughly 82% of the sentences of the first 16 sections of the Penn Treebank annotated with full HPSG analyses from the English Resource Grammar (ERG). The DeepBank annotations are created on top of the raw text of the PTB. Due to imperfections of the automatic tokenization, there are some token mismatches between DeepBank and PTB. We had to filter out such sentences to have consistent number of tokens in the DT, SB and CD formats. For our experiments we had available a training set of 22209 sentences and a test set of 1759 sentences (from Section 15).

#### 3.2 Parsers

In the experiments described in Section 4 we used parsers that adopt different approaches and implement various algorithms.

**Malt** (Nivre et al., 2007): transition-based dependency parser with local learning and greedy search.

**MST** (McDonald et al., 2005): graph-based dependency parser with global near-exhaustive search.

**Bohnet and Nivre (2012) parser**: transition-based dependency parser with joint tagger that implements global learning and beam search.

#### 3.3 Dependency schemes

In this work we extract DeepBank data in the form of bilinear syntactic dependencies, **DELPH-IN Syntactic Derivation Tree (DT)** format. We obtain the exact same sentences in **Stanford Basic (SB)** format from the automatic conversion of the PTB with the Stanford parser (de Marneffe et al., 2006) and in the **CoNLL Syntactic Dependencies (CD)** representation using the LTH Constituent-to-Dependency Conversion Tool for Penn-style Treebanks (Johansson and Nugues, 2007).

SB and CD represent the way to convert PTB to bilinear dependencies; in contrast, DT is grounded in linguistic theory and captures decisions taken in the grammar. Figure 1 demonstrates the differences between the formats on the coordination structure. According to Schwartz et al. (2012), analysis of coordination in SB and CD is easier for a statistical parser to learn; however, as we will see in section 4.3, DT has more expressive power distinguishing structural ambiguities illustrated by the classic example *old men and women*.

#### 3.4 Part-of-speech tags

We experimented with two tag sets: **PTB tags** and lexical types of the ERG grammar - **supertags**.

PTB tags determine the *part of speech* (PoS) and some *morphological features*, such as number for nouns, degree of comparison for adjectives and adverbs, tense and agreement with person and number of subject for verbs, etc.

Supertags are composed of *part-of-speech*, *valency* in the form of an ordered sequence of complements, and *annotations* that encompass category-internal subdivisions, e.g. mass vs. count vs. proper nouns, intersective vs. scopal adverbs,

or referential vs. expletive pronouns. Example of a supertag: *v\_np.is.le* (verb “is” that takes noun phrase as a complement).

There are 48 tags in the PTB tagset and 1091 supertags in the set of lexical types of the ERG.

The state-of-the-art accuracy of PoS-tagging on in-domain test data using gold-standard tokenization is roughly 97% for the PTB tagset and approximately 95% for the ERG supertags (Ytrestøl, 2011). Supertagging for the ERG grammar is an ongoing research effort and an off-the-shelf supertagger for the ERG is not currently available.

## 4 Experiments

In this section we give a detailed analysis of parsing into SB, CD and DT dependencies with Malt, MST and the Bohnet and Nivre (2012) parser.

### 4.1 Setup

For Malt and MST we perform the experiments on gold PoS tags, whereas the Bohnet and Nivre (2012) parser predicts PoS tags during testing.

Prior to each experiment with Malt, we used MaltOptimizer to obtain settings and a feature model; for MST we exploited default configuration; for the Bohnet and Nivre (2012) parser we set the beam parameter to 80 and otherwise employed the default setup.

With regards to evaluation metrics we use labelled attachment score (LAS), unlabeled attachment score (UAS) and label accuracy (LACC) excluding punctuation. Our results cannot be directly compared to the state-of-the-art scores on the Penn Treebank because we train on sections 0-13 and test on section 15 of WSJ. Also our results are not strictly inter-comparable because the setups we are using are different.

### 4.2 Discussion

The results that we are going to analyze are presented in Tables 1 and 2. Statistical significance was assessed using Dan Bikel’s parsing evaluation comparator<sup>1</sup> at the 0.001 significance level. We inspect three different aspects in the interpretation of these results: parser, dependency format and tagset. Below we will look at these three angles in detail.

From the **parser** perspective Malt and MST are not very different in the traditional setup with gold

PTB tags (Table 1, *Gold PTB tags*). The Bohnet and Nivre (2012) parser outperforms Malt on CD and DT and MST on SB, CD and DT with PTB tags even though it does not receive gold PTB tags during test phase but predicts them (Table 2, *Predicted PTB tags*). This is explained by the fact that the Bohnet and Nivre (2012) parser implements a novel approach to parsing: beam-search algorithm with global structure learning.

MST “loses” more than Malt when parsing SB with gold supertags (Table 1, *Gold supertags*). This parser exploits context features “*POS tag of each intervening word between head and dependent*” (McDonald et al., 2006). Due to the far larger size of the supertag set compared to the PTB tagset, such features are sparse and have low frequencies. This leads to the lower scores of parsing accuracy for MST. For the Bohnet and Nivre (2012) parser the complexity of supertag prediction has significant negative influence on the attachment and labeling accuracies (Table 2, *Predicted supertags*). The addition of gold PTB tags as a feature lifts the performance of the Bohnet and Nivre (2012) parser to the level of performance of Malt and MST on CD with gold supertags and Malt on SB with gold supertags (compare Table 2, *Predicted supertags + gold PTB*, and Table 1, *Gold supertags*).

Both Malt and MST benefit slightly from the combination of gold PTB tags and gold supertags (Table 1, *Gold PTB tags + gold supertags*). For the Bohnet and Nivre (2012) parser we also observe small rise of accuracy when gold supertags are provided as a feature for prediction of PTB tags (compare *Predicted PTB tags* and *Predicted PTB tags + gold supertags* sections of Table 2).

The parsers have different running times: it takes minutes to run an experiment with Malt, about 2 hours with MST and up to a day with the Bohnet and Nivre (2012) parser.

From the point of view of the **dependency format**, SB has the highest LACC and CD is first-rate on UAS for all three parsers in most of the configurations (Tables 1 and 2). This means that SB is easier to label and CD is easier to parse structurally. DT appears to be a more difficult target format because it is both hard to label and attach in most configurations. It is not an unexpected result, since SB and CD are both derived from PTB phrase-structure trees and are oriented to ease dependency parsing task. DT is not custom-designed

<sup>1</sup><http://nextens.uvt.nl/depparse-wiki/SoftwarePage#scoring>

<i>Gold PTB tags</i>							<i>Predicted PTB tags</i>			
	LAS		UAS		LACC			LAS	UAS	LACC
	Malt	MST	Malt	MST	Malt	MST		Bohnet and Nivre (2012)		
SB	<b>89.21</b>	88.59	90.95	90.88	<b>93.58</b>	<b>92.79</b>	SB	89.56	92.36	<b>93.30</b>
CD	88.74	<b>88.72</b>	<b>91.89</b>	<b>92.01</b>	91.29	91.34	CD	<b>89.77</b>	<b>93.01</b>	92.10
DT	85.97	86.36	89.22	90.01	88.73	89.22	DT	88.26	91.63	90.72

<i>Gold supertags</i>							<i>Predicted supertags</i>			
	LAS		UAS		LACC			LAS	UAS	LACC
	Malt	MST	Malt	MST	Malt	MST		Bohnet and Nivre (2012)		
SB	87.76	85.25	90.63	88.56	92.38	90.29	SB	85.41	89.38	<b>90.17</b>
CD	88.22	87.27	<b>91.17</b>	90.41	91.30	90.74	CD	<b>86.73</b>	<b>90.73</b>	89.72
DT	<b>89.92</b>	<b>89.58</b>	90.96	<b>90.56</b>	<b>92.50</b>	<b>92.64</b>	DT	85.76	89.50	88.56

<i>Gold PTB tags + gold supertags</i>							<i>Pred. PTB tags + gold supertags</i>			
	LAS		UAS		LACC			LAS	UAS	LACC
	Malt	MST	Malt	MST	Malt	MST		Bohnet and Nivre (2012)		
SB	90.32 <sup>1</sup>	89.43 <sup>1</sup>	91.90 <sup>1</sup>	91.84 <sup>2</sup>	<b>94.48<sup>1</sup></b>	93.26 <sup>1</sup>	SB	90.32	93.01	<b>93.85</b>
CD	89.59 <sup>1</sup>	89.37 <sup>2</sup>	<b>92.43<sup>1</sup></b>	<b>92.77<sup>2</sup></b>	92.32 <sup>1</sup>	92.07 <sup>2</sup>	CD	90.55	<b>93.56</b>	92.79
DT	<b>90.69<sup>1</sup></b>	<b>91.19<sup>2</sup></b>	91.83 <sup>1</sup>	92.33 <sup>2</sup>	93.10 <sup>1</sup>	<b>93.69<sup>2</sup></b>	DT	<b>91.51</b>	92.99	<b>93.88</b>

Table 1: Parsing results of Malt and MST on Stanford Basic (SB), CoNLL Syntactic Dependencies (CD) and DELPH-IN Syntactic Derivation Tree (DT) formats. Punctuation is excluded from the scoring. *Gold PTB tags*: Malt and MST are trained and tested on gold PTB tags. *Gold supertags*: Malt and MST are trained and tested on gold supertags. *Gold PTB tags + gold supertags*: Malt and MST are trained on gold PTB tags and gold supertags. <sup>1</sup> denotes a feature model in which gold PTB tags function as PoS and gold supertags act as additional features (in CPOSTAG field); <sup>2</sup> stands for the feature model which exploits gold supertags as PoS and uses gold PTB tags as extra features (in CPOSTAG field).

<i>Pred. supertags + gold PTB</i>			
	LAS	UAS	LACC
	Bohnet and Nivre (2012)		
SB	87.20	90.07	<b>91.81</b>
CD	<b>87.79</b>	<b>91.47</b>	90.62
DT	86.31	89.80	89.17

Table 2: Parsing results of the Bohnet and Nivre (2012) parser on Stanford Basic (SB), CoNLL Syntactic Dependencies (CD) and DELPH-IN Syntactic Derivation Tree (DT) formats. Parser is trained on gold-standard data. Punctuation is excluded from the scoring. *Predicted PTB*: parser predicts PTB tags during the test phase. *Predicted supertags*: parser predicts supertags during the test phase. *Predicted PTB + gold supertags*: parser receives gold supertags as feature and predicts PTB tags during the test phase. *Predicted supertags + gold PTB*: parser receives PTB tags as feature and predicts supertags during test phase.

to dependency parsing and is independent from parsing questions in this sense. Unlike SB and CD, it is linguistically informed by the underlying, full-fledged HPSG grammar.

The Jaccard similarity on our training set is 0.57 for SB and CD, 0.564 for CD and DT, and 0.388 for SB and DT. These similarity values show that CD and DT are structurally closer to each other than SB and DT. Contrary to our expectations, the accuracy scores of parsers do not suggest that CD and DT are particularly similar to each other in terms of parsing.

Inspecting the aspect of **tagset** we conclude that traditional PTB tags are compatible with SB and CD but do not fit the DT scheme well, while ERG supertags are specific to the ERG framework and do not seem to be appropriate for SB and CD. Neither of these findings seem surprising, as PTB tags were developed as part of the treebank from which CD and SB are derived; whereas ERG supertags are closely related to the HPSG syntactic structures captured in DT. PTB tags were designed to simplify PoS-tagging whereas supertags were developed to capture information that is required to analyze syntax of HPSG.

For each PTB tag we collected corresponding supertags from the gold-standard training set. For open word classes such as nouns, adjectives, adverbs and verbs the relation between PTB tags and supertags is many-to-many. Unique one-to-many correspondence holds only for possessive wh-pronoun and punctuation.

Thus, supertags do not provide extra level of detalization for PTB tags, but PTB tags and supertags are complementary. As discussed in section 3.4, they contain bits of information that are different. For this reason their combination results in slight increase of accuracy for all three parsers on all dependency formats (Table 1, *Gold PTB tags + gold supertags*, and Table 2, *Predicted PTB + gold supertags* and *Predicted supertags + gold PTB*). The Bohnet and Nivre (2012) parser predicts supertags with an average accuracy of 89.73% which is significantly lower than state-of-the-art 95% (Ytrestøl, 2011).

When we consider punctuation in the evaluation, all scores raise significantly for DT and at the same time decrease for SB and CD for all three parsers. This is explained by the fact that punctuation in DT is always attached to the nearest token which is easy to learn for a statistical parser.

### 4.3 Error analysis

Using the CoNLL-07 evaluation script<sup>2</sup> on our test set, for each parser we obtained the error rate distribution over CPOSTAG on SB, CD and DT.

**VBP, VBZ and VBG.** VBP (verb, non-3rd person singular present), VBZ (verb, 3rd person singular present) and VBG (verb, gerund or present participle) are the PTB tags that have error rates in 10 highest error rates list for each parser (Malt, MST and the Bohnet and Nivre (2012) parser) with each dependency format (SB, CD and DT) and with each PoS tag set (PTB PoS and supertags) when PTB tags are included as CPOSTAG feature. We automatically collected all sentences that contain 1) attachment errors, 2) label errors, 3) attachment and label errors for VBP, VBZ and VBG made by Malt parser on DT format with PTB PoS. For each of these three lexical categories we manually analyzed a random sample of sentences with errors and their corresponding gold-standard versions.

In many cases such errors are related to the root of the sentence when the verb is either treated as complement or adjunct instead of having a root status or vice versa. Errors with these groups of verbs mostly occur in the complex sentences that contain several verbs. Sentences with coordination are particularly difficult for the correct attachment and labeling of the VBP (see Figure 2 for an example).

**Coordination.** The error rate of Malt, MST and the Bohnet and Nivre (2012) parser for the coordination is not so high for SB and CD ( 1% and 2% correspondingly with MaltParser, PTB tags) whereas for DT the error rate on the CPOSTAGS is especially high (26% with MaltParser, PTB tags). It means that there are many errors on incoming dependency arcs for coordinating conjunctions when parsing DT. On outgoing arcs parsers also make more mistakes on DT than on SB and CD. This is related to the difference in choice of annotation principle (see Figure 1). As it was shown in (Schwartz et al., 2012), it is harder to parse coordination headed by coordinating conjunction.

Although the approach used in DT is harder for parser to learn, it has some advantages: using SB and CD annotations, we cannot distinguish the two cases illustrated with the sentences (a) and (b):

<sup>2</sup><http://nextens.uvt.nl/depparse-wiki/SoftwarePage#scoring>

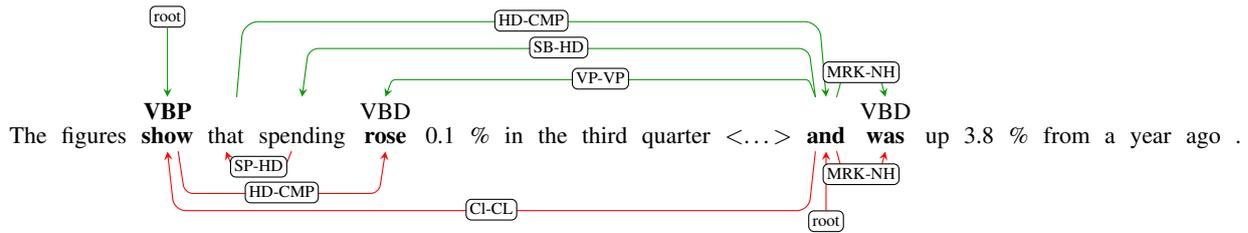


Figure 2: The gold-standard (*in green above the sentence*) and the incorrect Malt’s (*in red below the sentence*) analyses of the utterance from the DeepBank in DT format with PTB PoS tags

- The fight is putting a tight squeeze on profits of many, threatening to drive the smallest ones out of business and straining relations between *the national fast-food chains and their franchisees*.
- Proceeds from the sale will be used for remodelling and refurbishing projects, as well as for *the planned MGM Grand hotel/casino and theme park*.

In the sentence a) “the national fast-food” refers only to the conjunct “chains”, while in the sentence b) “the planned” refers to both conjuncts and “MGM Grand” refers only to the first conjunct.

The Bohnet and Nivre (2012) parser succeeds in finding the correct conjuncts (shown in bold font) on DT and makes mistakes on SB and CD in some difficult cases like the following ones:

- <...> investors **hoard** gold and **help** underpin its price <...>
- Then **take** the expected return and **subtract** one standard deviation.

CD and SB wrongly suggest “gold” and “help” to be conjoined in the first sentence and “return” and “deviation” in the second.

## 5 Conclusions and future work

In this survey we gave a comparative experimental overview of (i) parsing three dependency schemes, viz., Stanford Basic (SB), CoNLL Syntactic Dependencies (CD) and DELPH-IN Syntactic Derivation Tree (DT), (ii) with three leading dependency parsers, viz., Malt, MST and the Bohnet and Nivre (2012) parser (iii) exploiting two different tagsets, viz., PTB tags and supertags.

From the *parser* perspective, the Bohnet and Nivre (2012) parser performs better than Malt and MST not only on conventional formats but also on the new representation, although this parser solves a harder task than Malt and MST.

From the dependency *format* perspective, DT appears to be a more difficult target dependency representation than SB and CD. This suggests that the expressivity that we gain from the grammar theory (e.g. for coordination) is harder to learn with state-of-the-art dependency parsers. CD and DT are structurally closer to each other than SB and DT; however, we did not observe sound evidence of a correlation between structural similarity of CD and DT and their parsing accuracies

Regarding the *tagset* aspect, it is natural that PTB tags are good for SB and CD, whereas the more fine-grained set of supertags fits DT better. PTB tags and supertags are complementary, and for all three parsers we observe slight benefits from being supplied with both types of tags.

As future work we would like to run more experiments with predicted supertags. In the absence of a specialized supertagger, we can follow the pipeline of (Ytrestøl, 2011) who reached the state-of-the-art supertagging accuracy of 95%.

Another area of our interest is an extrinsic evaluation of SB, CD and DT, e.g. applied to semantic role labeling and question-answering in order to find out if the usage of the DT format grounded in the computational grammar theory is beneficial for such tasks.

## Acknowledgments

The authors would like to thank Rebecca Dridan, Joakim Nivre, Bernd Bohnet, Gertjan van Noord and Jelke Bloem for interesting discussions and the two anonymous reviewers for comments on the work. Experimentation was made possible through access to the high-performance computing resources at the University of Oslo.

## References

- Bernd Bohnet and Joakim Nivre. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *EMNLP-CoNLL*, pages 1455–1465. ACL.
- Harry Bunt, Paola Merlo, and Joakim Nivre, editors. 2010. *Trends in Parsing Technology*. Springer Verlag, Stanford.
- Jinho D Choi and Nicolas Nicolov. 2009. K-best, locally pruned, transition-based dependency parsing using robust risk minimization. *Recent Advances in Natural Language Processing V*, pages 205–216.
- Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure trees. In *LREC*.
- Yuan Ding and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 541–548, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Daniel Flickinger, Yi Zhang, and Valia Kordoni. 2012. DeepBank: a Dynamically Annotated Treebank of the Wall Street Journal. In *Proceedings of the Eleventh International Workshop on Treebanks and Linguistic Theories*, pages 85–96. Edies Colibri.
- Yoav Goldberg and Michael Elhadad. 2010. An efficient algorithm for easy-first non-directional dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 742–750, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Angelina Ivanova, Stephan Oepen, Lilja Øvrelid, and Dan Flickinger. 2012. Who did what to whom? a contrastive study of syntacto-semantic dependencies. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 2–11, Jeju, Republic of Korea, July. Association for Computational Linguistics.
- Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*, pages 105–112, Tartu, Estonia, May 25–26.
- Dan Klein and Christopher D. Manning. 2004. Corpus-based induction of syntactic structure: models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330, June.
- Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 523–530, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Ryan McDonald, Kevin Lerman, and Fernando Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, pages 216–220, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Makoto Miwa, Sampo Pyysalo, Tadayoshi Hara, and Jun'ichi Tsujii. 2010. Evaluating dependency representations for event extraction. In Chu-Ren Huang and Dan Jurafsky, editors, *COLING*, pages 779–787. Tsinghua University Press.
- Yusuke Miyao, Kenji Sagae, and Jun'ichi Tsujii. 2007. Towards framework-independent evaluation of deep linguistic parsers. In Ann Copestake, editor, *Proceedings of the GEAF 2007 Workshop*, CSLI Studies in Computational Linguistics Online, page 21 pages. CSLI Publications.
- Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chaney, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.
- Lilja Øvrelid, Jonas Kuhn, and Kathrin Spreyer. 2009. Cross-framework parser stacking for data-driven dependency parsing. *TAL*, 50(3):109–138.
- Roy Schwartz, Omri Abend, and Ari Rappoport. 2012. Learnability-based syntactic annotation design. In *Proc. of the 24th International Conference on Computational Linguistics (Coling 2012)*, Mumbai, India, December. Coling 2012 Organizing Committee.
- Xianchao Wu, Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2012. A Comparative Study of Target Dependency Structures for Statistical Machine Translation. In *ACL (2)*, pages 100–104. The Association for Computer Linguistics.
- Akane Yakushiji, Yusuke Miyao, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2006. Automatic construction of predicate-argument structure patterns for biomedical information extraction. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 284–292, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Gisle Ytrestøl. 2011. Cutforce: deep deterministic HPSG parsing. In *Proceedings of the 12th International Conference on Parsing Technologies*, IWPT '11, pages 186–197, Stroudsburg, PA, USA. Association for Computational Linguistics.