Investigating Multilingual Dependency Parsing

Richard Johansson
Department of Computer Science
LTH, Lund University
221 00 Lund, Sweden
Richard.Johansson@cs.lth.se

Pierre Nugues
Department of Computer Science
LTH, Lund University
221 00 Lund, Sweden
Pierre.Nugues@cs.lth.se

Abstract

In this paper, we describe a system for the CoNLL-X shared task of multilingual dependency parsing. It uses a baseline Nivre's parser (Nivre, 2003) that first identifies the parse actions and then labels the dependency arcs. These two steps are implemented as SVM classifiers using LIBSVM. Features take into account the static context as well as relations dynamically built during parsing.

We experimented two main additions to our implementation of Nivre's parser: N-best search and bidirectional parsing. We trained the parser in both left-right and right-left directions and we combined the results. To construct a single-head, rooted, and cycle-free tree, we applied the Chu-Liu/Edmonds optimization algorithm. We ran the same algorithm with the same parameters on all the languages.

1 Nivre's Parser

Nivre (2003) proposed a dependency parser that creates a projective and acyclic graph. The parser is an extension to the shift–reduce algorithm. As with the regular shift–reduce, it uses a stack S and a list of input words W. However, instead of finding constituents, it builds a set of arcs G representing the graph of dependencies.

Nivre's parser uses two operations in addition to shift and reduce: left-arc and right-arc. Given a sequence of words, possibly annotated with their part of speech, parsing simply consists in applying a sequence of operations: left-arc (`la`), right-arc (`ra`), reduce (`re`), and shift (`sh`) to the input sequence.

2 Parsing an Annotated Corpus

The algorithm to parse an annotated corpus is straightforward from Nivre's parser and enables us to obtain, for any projective sentence, a sequence of actions taken in the set $\{\texttt{la}, \texttt{ra}, \texttt{re}, \texttt{sh}\}$ that parses it. At a given step of the parsing process, let TOP be the top of the stack and $FIRST$, the first token of the input list, and arc, the relation holding between a head and a dependent.

1. **if** $arc(TOP, FIRST) \in G$, **then** `ra`;
2. **else if** $arc(FIRST, TOP) \in G$, **then** `la`;
3. **else if** $\exists k \in Stack, arc(FIRST, k) \in G$ or $arc(k, FIRST) \in G$, **then** `re`;
4. **else** `sh`.

Using the first sentence of the Swedish corpus as input (Table 1), this algorithm produces the sequence of 24 actions: `sh`, `sh`, `la`, `ra`, `re`, `la`, `sh`, `sh`, `sh`, `la`, `la`, `ra`, `ra`, `sh`, `la`, `re`, `ra`, `ra`, `ra`, `re`, `re`, `re`, `re`, and `ra` (Table 2).

3 Adapting Nivre's Algorithm to Machine–Learning

3.1 Overview

We used support vector machines to predict the parse action sequence and a two step procedure to

Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-X),
pages 206–210, New York City, June 2006. ©2006 Association for Computational Linguistics

Table 1: Dependency graph of the sentence *Äktenskapet och familjen är en gammal institution, som funnits sedan 1800-talet* 'Marriage and family are an old institution that has been around from the 19th century'.

ID	**Form**	**POS**	**Head**	**Rel.**
1	Äktenskapet	NN	4	SS
2	och	++	3	++
3	familjen	NN	1	CC
4	är	AV	0	ROOT
5	en	EN	7	DT
6	gammal	AJ	7	AT
7	institution	NN	4	SP
8	,	IK	7	IK
9	som	PO	10	SS
10	funnits	VV	7	ET
11	sedan	PR	10	TA
12	1800-talet	NN	11	PA
13	.	IP	4	IP

produce the graph. We first ran the classifier to select unlabeled actions, `la`, `ra`, `sh`, `re`. We then ran a second classifier to assign a function to `ra` and `la` parse actions.

We used the LIBSVM implementation of the SVM learning algorithm (Chang and Lin, 2001). We used the Gaussian kernel throughout. Optimal values for the parameters (C and γ) were found using a grid search. The first predicted action is not always possible, given the parser's constraints. We trained the model using probability estimates to select the next possible action.

3.2 Feature Set

We used the following set of features for the classifiers:

- Word and POS of *TOP* and *FIRST*
- Word and POS of the second node on the stack
- Word and POS of the second node in the input list
- POS of the third and fourth nodes in the input list
- The dependency type of *TOP* to its head, if any
- The word, POS, and dependency type of the leftmost child of *TOP* to *TOP*, if any
- The word, POS, and dependency type of the rightmost child of *TOP* to *TOP*, if any
- The word, POS, and dependency type of the leftmost child of *FIRST* to *FIRST*, if any

For the POS, we used the Coarse POS, the Fine POS, and all the features (encoded as boolean flags). We did not use the lemma.

Table 2: Actions to parse the sentence *Äktenskapet och familjen är en gammal institution, som funnits sedan 1800-talet*.

Ac.	**Top word**	**First word**	**Rel.**
sh	*nil*	Äktenskapet	
sh	Äktenskapet	och	
la	och	familjen	++
ra	Äktenskapet	familjen	CC
re	familjen	är	
la	Äktenskapet	är	SS
sh	nil	är	
sh	är	en	
sh	en	gammal	
la	gammal	institution	AT
la	en	institution	DT
ra	är	institution	SP
ra	institution	,	IK
sh	,	som	
la	som	funnits	SS
re	,	funnits	
ra	institution	funnits	ET
ra	funnits	sedan	TA
ra	sedan	1800-talet	PA
re	1800-talet	.	
re	sedan	.	
re	funnits	.	
re	institution	.	
ra	är	.	IP

4 Extensions to Nivre's Algorithm

4.1 N-best Search

We extended Nivre's original algorithm with a beam search strategy. For each action, `la`, `ra`, `sh` and `re`, we computed a probability score using LIBSVM. These scores can then be used to carry out an N-best search through the set of possible sequences of actions.

We measured the improvement over a best-first strategy incrementing values of N. We observed the largest difference between N = 1 and N = 2, then leveling off and we used the latter value.

4.2 Bidirectionality and Voting

Tesnière (1966) classified languages as centrifuge (head to the left) and centripetal (head to the right) in a table (page 33 of his book) that nearly exactly fits corpus evidence from the CONLL data. Nivre's parser is inherently left-right. This may not fit all the languages. Some dependencies may be easier to capture when proceeding from the reverse direction. Jin et al. (2005) is an example of it for Chinese, where the authors describe an adaptation of Nivre's parser to bidirectionality.

We trained the model and ran the algorithm in both directions (left to right and right to left). We used a voting strategy based on probability scores. Each link was assigned a probability score (simply by using the probability of the `la` or `ra` actions for each link). We then summed the probability scores of the links from all four trees. To construct a single-head, rooted, and cycle-free tree, we finally applied the Chu-Liu/Edmonds optimization algorithm (Chu and Liu, 1965; Edmonds, 1967).

5 Analysis

5.1 Experimental Settings

We trained the models on "projectivized" graphs following Nivre and Nilsson (2005) method. We used the complete annotated data for nine langagues. Due to time limitations, we could not complete the training for three languages, Chinese, Czech, and German.

5.2 Overview of the Results

We parsed the 12 languages using exactly the same algorithms and parameters. We obtained an average score of 74.93 for the labeled arcs and of 80.39 for the unlabeled ones (resp. 74.98 and 80.80 for the languages where we could train the model using the complete annotated data sets). Table 3 shows the results per language. As a possible explanation of the differences between languages, the three lowest figures correspond to the three smallest corpora. It is reasonable to assume that if corpora would have been of equal sizes, results would have been more similar. Czech is an exception to this rule that applies to all the participants. We have no explanation for this. This language, or its annotation, seems to be more complex than the others.

The percentage of nonprojective arcs also seems to play a role. Due to time limitations, we trained the Dutch and German models with approximately the same quantity of data. While both languages are closely related, the Dutch corpus shows twice as much nonprojective arcs. The score for Dutch is significantly lower than for German.

Our results across the languages are consistent with the other participants' mean scores, where we are above the average by a margin of 2 to 3% except for Japanese and even more for Chinese where we obtain results that are nearly 7% less than the average for labeled relations. Results are similar for unlabeled data. We retrained the data with the complete Chinese corpus and you obtained 74.41 for the labeled arcs, still far from the average. We have no explanation for this dip with Chinese.

5.3 Analysis of Swedish and Portuguese Results

5.3.1 Swedish

We obtained a score of 78.13% for the labeled attachments in Swedish. The error breakdown shows significant differences between the parts of speech. While we reach 89% of correct head and dependents for the adjectives, we obtain 55% for the prepositions. The same applies to dependency types, 84% precision for subjects, and 46% for the OA type of prepositional attachment.

There is no significant score differences for the left and right dependencies, which could attributed to the bidirectional parsing (Table 4). Distance plays a dramatic role in the error score (Table 5). Prepositions are the main source of errors (Table 6).

5.3.2 Portuguese

We obtained a score 84.57% for the labeled attachments in Portuguese. As for Swedish, error distribution shows significant variations across the

Table 3: Summary of results. We retrained the Chinese* model after the deadline.

Languages	Unlabeled	Labeled
Completed training		
Arabic	75.53	64.29
Chinese*	79.13	74.41
Danish	86.59	81.54
Dutch	76.01	72.67
Japanese	87.11	85.63
Portuguese	88.4	84.57
Slovene	74.36	66.43
Spanish	81.43	78.16
Swedish	84.17	78.13
Turkish	73.59	63.39
$\overline{x}$	80.80	74.98
σ	5.99	8.63
Noncompleted training		
Chinese	77.04	72.49
Czech	77.4	71.46
German	83.09	80.43
$\overline{x}$ all languages	80.39	74.93
σ all languages	5.36	7.65

parts of speech, with a score of 94% for adjectives and only 67% for prepositions.

As for Swedish, there is no significant score differences for the left and right dependencies (Table 7). Distance also degrades results but the slope is not as steep as with Swedish (Table 8). Prepositions are also the main source of errors (Table 9).

5.4 Acknowledgments

This work was made possible because of the annotated corpora that were kindly provided to us: Arabic (Hajič et al., 2004), Bulgarian (Simov et al., 2005; Simov and Osenova, 2003), Chinese (Chen et al., 2003), Czech (Böhmová et al., 2003), Danish (Kromann, 2003), Dutch (van der Beek et al., 2002), German (Brants et al., 2002), Japanese (Kawata and Bartels, 2000), Portuguese (Afonso et al., 2002), Slovene (Džeroski et al., 2006), Spanish (Civit Torruella and Martí Antonín, 2002), Swedish (Nilsson et al., 2005), and Turkish (Oflazer et al., 2003; Atalay et al., 2003).

Table 4: Precision and recall of binned HEAD direction. Swedish.

Dir.	Gold	Cor.	Syst.	R	P
to_root	389	330	400	84.83	82.50
left	2745	2608	2759	95.01	94.53
right	1887	1739	1862	92.16	93.39

Table 5: Precision and recall of binned HEAD distance. Swedish.

Dist.	Gold	Cor.	Syst.	R	P
to_root	389	330	400	84.83	82.50
1	2512	2262	2363	90.05	95.73
2	1107	989	1122	89.34	88.15
3-6	803	652	867	81.20	75.20
7-...	210	141	269	67.14	52.42

Table 6: Focus words where most of the errors occur. Swedish.

Word	POS	Any	Head	Dep	Both
till	PR	48	20	45	17
i	PR	42	25	34	17
på	PR	39	22	32	15
med	PR	28	11	25	8
för	PR	27	22	25	20

Table 7: Precision and recall of binned HEAD direction. Portuguese.

Dir.	Gold	Cor.	Syst.	R	P
to_root	288	269	298	93.40	90.27
left	3006	2959	3020	98.44	97.98
right	1715	1649	1691	96.15	97.52

Table 8: Precision and recall of binned HEAD distance. Portuguese.

Dist.	Gold	Cor.	Syst.	R	P
to_root	288	269	298	93.40	90.27
1	2658	2545	2612	95.75	97.43
2	1117	1013	1080	90.69	93.80
3-6	623	492	647	78.97	76.04
7-...	323	260	372	80.50	69.89

Table 9: Focus words where most of the errors occur. Portuguese.

Word	POS	Any	Head	Dep	Both
em	prp	66	38	47	19
de	prp	51	37	35	21
a	prp	46	30	39	23
e	conj	28	28	0	0
para	prp	21	13	18	10

References

A. Abeillé, editor. 2003. *Treebanks: Building and Using Parsed Corpora*, volume 20 of *Text, Speech and Language Technology*. Kluwer Academic Publishers, Dordrecht.

S. Afonso, E. Bick, R. Haber, and D. Santos. 2002. “Floresta sintá(c)tica”: a treebank for Portuguese. In *Proc. of the Third Intern. Conf. on Language Resources and Evaluation (LREC)*, pages 1698–1703.

N. B. Atalay, K. Oflazer, and B. Say. 2003. The annotation process in the Turkish treebank. In *Proc. of the 4th Intern. Workshop on Linguistically Interpreteted Corpora (LINC)*.

A. Böhmová, J. Hajič, E. Hajičová, and B. Hladká. 2003. The PDT: a 3-level annotation scenario. In Abeillé (Abeillé, 2003), chapter 7.

S. Brants, S. Dipper, S. Hansen, W. Lezius, and G. Smith. 2002. The TIGER treebank. In *Proc. of the First Workshop on Treebanks and Linguistic Theories (TLT)*.

Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*.

K. Chen, C. Luo, M. Chang, F. Chen, C. Chen, C. Huang, and Z. Gao. 2003. Sinica treebank: Design criteria, representational issues and implementation. In Abeillé (Abeillé, 2003), chapter 13, pages 231–248.

Y.J. Chu and T.H. Liu. 1965. On the shortest arborescence of a directed graph. *Science Sinica*, 14:1396–1400.

M. Civit Torruella and M^{a} A. Martí Antonín. 2002. Design principles for a Spanish treebank. In *Proc. of the First Workshop on Treebanks and Linguistic Theories (TLT)*.

S. Džeroski, T. Erjavec, N. Ledinek, P. Pajas, Z. Žabokrtsky, and A. Žele. 2006. Towards a Slovene dependency treebank. In *Proc. of the Fifth Intern. Conf. on Language Resources and Evaluation (LREC)*.

J. Edmonds. 1967. Optimum branchings. *Journal of Research of the National Bureau of Standards*, 71B:233–240.

J. Hajič, O. Smrž, P. Zemánek, J. Šnaidauf, and E. Beška. 2004. Prague Arabic dependency treebank: Development in data and tools. In *Proc. of the NEMLAR Intern. Conf. on Arabic Language Resources and Tools*, pages 110–117.

Meixun Jin, Mi-Young Kim, and Jong-Hyeok Lee. 2005. Two-phase shift-reduce deterministic dependency parser of Chinese. In *Proceedings of the Second International Joint Conference on Natural Language Processing*.

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

M. T. Kromann. 2003. The Danish dependency treebank and the underlying linguistic theory. In *Proc. of the Second Workshop on Treebanks and Linguistic Theories (TLT)*.

J. Nilsson, J. Hall, and J. Nivre. 2005. MAMBA meets TIGER: Reconstructing a Swedish treebank from antiquity. In *Proc. of the NODALIDA Special Session on Treebanks*.

Joakim Nivre and Jens Nilsson. 2005. Pseudo-projective dependency parsing. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 99–106, Ann Arbor, June.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT 03)*, pages 149–160, Nancy, 23-25 April.

K. Oflazer, B. Say, D. Zeynep Hakkani-Tür, and G. Tür. 2003. Building a Turkish treebank. In Abeillé (Abeillé, 2003), chapter 15.

K. Simov and P. Osenova. 2003. Practical annotation scheme for an HPSG treebank of Bulgarian. In *Proc. of the 4th Intern. Workshop on Linguistically Interpreteted Corpora (LINC)*, pages 17–24.

K. Simov, P. Osenova, A. Simov, and M. Kouylekov. 2005. Design and implementation of the Bulgarian HPSG-based treebank. In *Journal of Research on Language and Computation – Special Issue*, pages 495–522. Kluwer Academic Publishers.

Lucien Tesnière. 1966. *Éléments de syntaxe structurale*. Klincksieck, Paris, 2e edition.

L. van der Beek, G. Bouma, R. Malouf, and G. van Noord. 2002. The Alpino dependency treebank. In *Computational Linguistics in the Netherlands (CLIN)*.