УНИВЕРСАЛЬНЫЕ ЗАВИСИМОСТИ: СРАВНЕНИЕ СИНТАКСИЧЕСКОГО АНАЛИЗА ДЛЯ ШВЕДСКОГО ЯЗЫКА

Антомонов Ф. А. (filip.antomonov.7239@student.uu.se)
Отделение лингвистики и филологии
Уппсальский университет, Уппсала, Швеция

Ключевые слова: универсальные зависимости, синтаксический анализ, шведский, корпус

UNIVERSAL DEPENDENCIES: A PARSING COMPARISON FOR SWEDISH

Filip Antomonov (filip.antomonov.7239@student.uu.se)
Department of Linguistics and Philology
Uppsala University, Uppsala, Sweden

New annotation approach in the form of Universal Dependencies aims to provide a consistent, language-independent grammatical annotation scheme for dependency treebanks. However, since UD are not related to any particular language or language group, there is an interest to investigate what impact Universal Dependencies might have on parsing quality in comparison to classic annotation schemes. This article presents results of a parsing study for Swedish, where two independent parsing systems, MaltParser and Stanford NN Parser, were trained and evaluated on the novel UD Treebank as well as on the classic Talbanken non-UD treebank. The results show that Universal Dependencies do not bring any drawbacks to parsing quality, in fact delivering a slight increase of the scores in the evaluation.

Key words: universal dependencies, parsing, swedish, treebank

1 Introduction

Universal Dependencies[1] as a project to develop consistent grammatical annotation for dependency treebanks, create new opportunities for multilingual research and development in natural language processing, in areas like cross-linguistic evaluation of empirical results and multilingual parser development. However, since UD are not related to any particular language or language group, indeed aiming at creating a common annotation scheme for potentially any human language, it is still relevant to get acquainted with any implications the new annotation scheme might have on language specific information, encoded in local treebank annotation schemes.

For this work, the aim has therefore been to answer the question if the use of UD has any impact on parsing quality in a monolingual environment, namely for Swedish. Since the main target of Universal Dependencies is in multilingual natural language processing, it may be worth investigating whether there are any costs or gains in using a UD annotated treebank in a situation where it

[1] universaldependencies.org

would not be technically required. To answer the question, models on the new UD and the classic non-UD versions of the Swedish treebank were trained with two parsers, results of which were analyzed by two different evaluation metrics.

2 Previous Work

In the field of natural language processing, and in that of syntactic parsing in particular, access to grammatically annotated treebanks is of key importance as of today. However, the annotation schemes of treebanks for different languages are often very different in structure – to the point where it is sometimes of considerable difficulty to say if performance differences are to be explained by real structural divergence of languages or mere annotation differences between treebanks (Nivre, 2015). Several steps towards a more consistent framework have been made in recent years.

In case of multilingual parsing, parallel corpora are frequently used. However, there have been some successful transfer attempts when parallel data is unavailable. McDonald et al. (2011) show a delexicalized direct transfer method, where for any training set only features like PoS tags and syntactic attachment direction are used. The model is then built from the data of the annotated source language and is used to parse the target language. Authors note that differences between annotation schemes in the treebanks are often the cause of the fact that some of the language pairs may work well together, while others – even if they are typologically similar – may sometimes not. Zeman et al. (2012) harmonize treebanks of 29 languages by means of mapping their annotation styles to a version of the scheme used by the Prague Dependency Treebank. Later, McDonald et al. (2013) showed an improvement of the results of cross-lingual direct transfer parsing by using the Universal Treebank which contains a uniformed syntactic annotation scheme for several languages, thus enabling cross-lingual training of parser models. As a baseline for model transfer, delexicalized models are proposed. Experiments show, that even while parsers, trained on data from languages in the same language group, do achieve the best results, training parsers also across language groups is certainly not pointless.

Recently, the project of *Universal Dependencies* has been gaining speed. Its aim is to develop cross-lingual treebank annotation for a large number of languages. Being an extension of several previous efforts, its goal is to find unified approaches with regard to parts-of-speech, morphosyntactic descriptions and dependency relations (Zeman, 2015). The idea is that the same construction should be annotated the same way across languages, but at the same time without annotating things not existing in a particular language simply because they may be present in other languages.

The UD morphological specification is based on three information levels: lemma, POS tag and a set of features encoding lexical and grammatical properties of word forms. The 17 POS tags are divided into open and closed class words, as well as into a class for other symbols, like punctuation. That tag inventory is fixed, but not all categories need to be used in all languages. In order to maximize parallelism across languages, UD give priority to dependency relations between content words. The motivation behind is the idea that this will help in finding parallel structures, as function words in one language often correspond to, for instance, morphological inflection in other languages. As every word depends on another word in a sentence, content words are related by dependency relations, function words are connected to the content word they specify, and punctuation is attached to the phrase's head (Nivre et al, 2016).

To speed up adoption of UD, efforts are being made to convert the existing dependency treebanks to conform with Universal Dependencies. In case of Swedish, the widely used Swedish Treebank (Nivre & Megyesi, 2007) has been converted and is freely available in an updated version in the UD repository.

In regard to parsing software, a well-known and widely used member of the community is the dependency parsing system MaltParser (Nivre et al, 2007). Being a data-driven and language-independent syntactic parser, it has been successfully used on many languages and language domains, achieving good parsing results. A recent trend in parsing lies within the field of neural networks. Chen & Manning (2014) present a way of learning a neural network classifier for use in a transition-based dependency parser. It is yet to be tested in a wide range of language domains, but the parser has already been used to achieve a notable improvement regarding labeled and unlabeled attachment scores for Chinese and English datasets, while showing fast processing speeds during parsing phase.

3 Method and NLP Tools

The main goal of this project was to investigate if the use of Universal Dependencies has any impact on parsing performance in comparison to the parsing results of the Talbanken non-UD version of the Swedish treebank[2] In order to achieve this, sets of the classic Talbanken and the novel UD version (1.2) of the Swedish treebank were trained and parsed, with results evaluated and compared.

The two parsing suits used were MaltParser (Nivre et al, 2007) and Stanford Neural Network Parser (Chen & Manning, 2014). By using two parsing systems, the idea was both to get larger comparison data, as well as to try to minimize the risk of potential parser bias in the analysis of Talbanken versus the UD Treebank, by having two grounds to base the results on.

MaltParser, as the first tool, can be used straight out of the box if the treebank is in the suitable CONLL format. However, since the parser has many configurable options and can employ several parsing algorithms, there is room for some optimization of the process to achieve better results. In order to do so, MaltParser system also provides the MaltOptimizer tool (Ballesteros & Nivre, 2012), which can be used to pick the most suitable MaltParser configuration, given the analysis of the training data of the treebank used. The configuration chosen by MaltOptimizer can then be used by MaltParser during the training phase. The parser itself does not perform the evaluation of the results, but its environment provides the MaltEval tool (Nilsson & Nivre, 2008), which can be used for comparison of the gold standard of the test set and the output of the parser, both on the level of computing labeled (LAS) and unlabeled attachment scores (UAS), as well as, for instance, by providing statistics of dependency relation labels of the sets.

The second parsing tool, the Stanford NN Parser, doesn't provide the same level of external optimization, but does compute the attachment scores at the end of the parsing phase. That stage is also clearly quicker in comparison to MaltParser. However, the training of the model is extremely slow compared to MaltParser (between 5 and 15 hours on the two machines used, versus less than 2 minutes for MaltParser on the same machines). Stanford NN Parser also requires distributed representations of words of any languages appearing in the treebanks, in the form of a word embeddings file. The authors state that it is not absolutely necessary for all words in the treebank to be covered in such a vector file, but note that parser's performance does improve with more comprehensive embeddings. For experiments presented in this article, the vector representation used is the 25-dimensional Swedish word embeddings file, produced during the SPMRL'13 Shared Task workshop (Cirik & Şensoy, 2013).

For computing labeled and unlabeled attachment scores, the mentioned MaltEval tool was used. Since it only requires gold and parse files to be in the same format, it can be used for any parser as long as that requirement is met. However, that metric itself, even though widely used otherwise, isn't particularly well suited for the task at hand, which is the parsing comparison of two closely related, but representatively different treebanks. Therefore, since representations in the training sets of Talbanken and the UD Treebank are not equivalent, it is unreasonable to simply compare attachment scores between the treebanks. Hence, in addition to the usual metrics, the experiments were also evaluated

[2] stp.lingfil.uu.se/~nivre/swedish_treebank

with the TedEval tool (Tsarfaty et al, 2011), whose evaluation metrics take into account different annotation schemes across multiple parsing experiments, providing a more objective measure of parsing performance, while allowing for direct comparison of parsing results across the two treebanks and the two parsers.

The treebanks themselves are slightly different in their setup, which also reflects in minor differences of treebank layout requirements across the two parsers. Talbanken consists of a test and a training set, which is fine for MaltParser since it creates the development set internally during training. For Stanford NN Parser however, there is a need for a separate development set, which required cutting off the latter part of the training set for use as development. The UD Treebank is instead split up into three parts, therefore the situation is a mirror image – because it consists of both a development and a training set, it was necessary to instead combine those sets for use with MaltParser. In case of TedEval, which requires that the sentence composition is exactly the same across both treebanks' test/parse sets, a couple of differing sentences from those sets were removed to facilitate consistency.

4 Results and Discussion

Training phases of the parsers generated four models, giving way for four parsed output sets, which were compared to two gold standards, one for each of the treebanks. The MaltEval generated labeled and unlabeled attachment scores of the comparison experiments for the two parsers over the two treebanks are presented in *Table 1*. Because of differences between training sets of the treebanks, attachment scores should not be compared to each other across the treebanks (even though some patterns can be seen), but rather between parsers. The comparison clearly shows that MaltParser is doing a better parsing analysis than Stanford NN Parser both for the new UD Treebank (ver. 1.2), as well as for the classic Talbanken. The score differences in regards to that are quite consistent, ranging from 1.4 % (Talbanken LAS), to 3.7 % (UD Treebank UAS) – all being in favor of MaltParser. Generally, the scores straightforwardly drop, starting from MaltParser UD Treebank UAS to Stanford NN Parser Talbanken LAS – with only one exception, that being Stanford NN Parser Talbanken UAS, which actually is higher than UD Treebank UAS score for the same parser.

	UD Treebank	**Talbanken**
	UAS / LAS	**UAS / LAS**
MaltParser	86.5 / 83.2	85.3 / 79.2
Stanford NN Parser	82.8 / 80.1	83.6 / 77.8

Table 1. Parser attachment scores across treebanks.

The inferior results of Stanford NN Parser in comparison to MaltParser, despite the former showing a noticeable attachment score increase in work presented by its authors, praising its neural network approach, were subject to some investigation. One idea was that MaltParser could theoretically make use of more linguistic information, present in the treebanks. That is, the CONLL format, being quite rich in its data encoding capabilities, could possibly not been fully utilized by Stanford NN Parser, with the parser missing to make use of some of the data columns in the treebanks. In fact, Stanford NN Parser, while making use of the fine-grained POSTAG column, does not utilize the LEMMA and FEATS columns in the treebanks, while MaltParser does. To test whether the results of MaltParser could drop to the level of Stanford NN Parser, or perhaps below, MaltParser was retrained on a version of the UD Treebank where the said columns were inactivated by a script. However, the parsing scores of such a model (possibly due to redundancy between, for instance, POSTAG and FEATS columns) weren't very different for MaltParser (86.8 / 83.5), stating that the problem should be searched for elsewhere. Results of other neural network parsers have in similar studies shown to be responsive to the size of the training set, and since the Swedish UD Treebank is relatively small, that could be the reason for score degradation. On a wider scale, that could suggest that neural network parsers overall require larger treebank sizes to be able to show their full potential.

As an additional experiment in connection to the study, Talbanken treebank was also parsed using automatic part-of-speech tags, with an aim to show any implications that they might have on parsing scores. For that experiment, the test set of the treebank was tagged by Stagger PoS tagger (Östling, 2013), previously showing great tagging results for Swedish, with treebank's gold coarse-grained PoS tags replaced by automatically generated ones. While MaltParser's results (82.6 / 75.9), originally being higher, dropped more than Stanford NN Parser's (81.7 / 75.1), the overall drop is perhaps stronger than expected, highlighting the importance of part-of-speech tagging. This area should be explored further in future work.

Some statistics of dependency relation labels were collected through MaltEval for the two treebanks, presented in *Table 3.* Examples of labels in the UD Treebank which appeared to be especially difficult for both parsers were parataxis, adjectival clause (acl), appositional modifier (appos), clausal passive subject (csubjpass), fronted or postposed element (dislocated). Labels which the parsers passed satisfactory were ones like compounding of proper nouns (name), punctuation (punct), coordinating conjunction (cc), possessive nominal modifier (nmod:poss), negation modifier (neg). In case of Talbanken labels, difficult examples were apposition (an), predicative attribute (pt), infinitive object complement (vo), free subjective predicative complement (fp) and comparative adverbial (ka), while negation adverbial (na), various types of punctuation (iu, ip, i?), verb particle (pl), determiner (dt) and adjectival pre-modifier (at) turned out well.

The scores computed by TedEval are presented in *Table 2*. These can be directly compared to each other in all directions, ultimately shedding light on the initial question of the study, answering it in a confident way: the use of Universal Dependencies has a clear positive impact on parsing quality. At least for the parsers used, the results can also be shown to be parser-independent. In fact, the percentage difference amidst treebanks is exactly the same between parsers (3.6 %), both in favor of the UD Treebank. Clearly, the results show that there are no losses in using UD, but there needs to be some explanation of the gains. For that, at least one logical gain interpretation may lie in the fact that the UD Treebank has gone through an extra revision by human annotators, taking care of any bugs present in the old version, Talbanken. This higher level of consistency, together with the use of a more fine-grained tagset and the treatment of coordination, would then explain the increase in parsing scores.

	UD Treebank	**Talbanken**
MaltParser	93.9	90.3
Stanford NN Parser	93.3	89.7

Table 2. TedEval scores of the treebanks.

5 Conclusion

The aim of this work has been to investigate the impact on parsing performance of the new treebank annotation scheme, the Universal Dependencies. Any concerns, related to whether such a language-independent annotation approach could have negative impact on parsing quality in a monolingual environment, can likely be dropped: Universal Dependencies in fact increase parsing quality for Swedish by a small margin, the results which are consistent across both tested parsers. Overall, the Universal Dependencies, if widely adopted, are a clear step forward for the usefulness of treebanks in natural language processing, especially in a multilingual setting.

Acknowledgments

The author would like to thank Professor Joakim Nivre, who has provided outstanding guidance and valuable feedback during the semester, making the project at hand possible.

MaltParser: UD Treebank

precision	recall	fscore	Deprel
1	1	1	name
0.998	0.994	0.996	punct
0.988	0.99	0.989	aux
0.993	0.982	0.988	cc
0.981	0.991	0.986	nmod:poss
0.987	0.982	0.985	cop
0.994	0.951	0.972	neg
0.96	0.975	0.968	case
0.969	0.958	0.963	det
0.926	0.982	0.953	nummod
0.935	0.97	0.952	amod
0.955	0.94	0.948	mark
0.987	0.906	0.945	compound:prt
0.902	0.923	0.912	advmod
0.908	0.912	0.91	nmod
0.895	0.897	0.896	root
0.904	0.88	0.892	nsubj
0.886	0.895	0.89	nsubjpass
1	0.8	0.889	auxpass
0.854	0.893	0.873	dobj
0.852	0.861	0.856	acl:relcl
0.791	0.807	0.799	conj
0.752	0.849	0.798	expl
0.835	0.732	0.78	mwe
0.714	0.7	0.707	xcomp
0.704	0.691	0.697	ccomp
0.665	0.679	0.672	advcl
1	0.5	0.667	compound
0.739	0.567	0.642	csubj
0.741	0.541	0.625	iobj
0.618	0.579	0.598	acl
0.467	0.452	0.459	dislocated
0.419	0.487	0.451	parataxis
0.5	0.375	0.429	csubjpass
0.391	0.34	0.364	appos
0.143	0.071	0.095	nmod:agent

MaltParser: Talbanken

precision	recall	fscore	Deprel
1	1	1	I?
1	1	1	IP
1	1	1	IU
1	1	1	XT
0.97	0.978	0.974	IF
0.938	1	0.968	IQ
0.935	1	0.966	IR
0.935	1	0.966	JR
0.961	0.966	0.963	NA
0.944	0.953	0.948	DT
0.93	0.962	0.946	IK
0.943	0.946	0.944	VG
0.975	0.901	0.937	PL
0.895	0.954	0.924	AT
0.91	0.934	0.922	PA
0.932	0.902	0.917	IC
0.927	0.895	0.911	JC
0.908	0.898	0.903	SS
0.898	0.899	0.899	ROOT
0.935	0.847	0.889	IT
0.916	0.841	0.877	+A
0.861	0.868	0.864	UA
0.814	0.846	0.83	OO
0.788	0.81	0.799	SP
1	0.647	0.786	IG
0.803	0.767	0.785	CA
0.788	0.781	0.784	CJ
0.913	0.656	0.764	VA
0.806	0.684	0.74	FS
0.713	0.753	0.733	ET
0.77	0.646	0.703	HD
0.706	0.664	0.684	MS
0.707	0.644	0.674	AA
0.588	0.769	0.667	DB
0.611	0.689	0.648	OA
0.75	0.553	0.636	IO
0.588	0.667	0.625	AG
0.6	0.6	0.6	IS
0.639	0.531	0.58	TA
0.533	0.608	0.568	ES
0.697	0.479	0.568	MA
0.497	0.604	0.545	RA
0.562	0.527	0.544	+F
0.533	0.533	0.533	JT
0.391	0.818	0.529	VS
0.571	0.444	0.5	OP
0.483	0.452	0.467	KA
0.455	0.455	0.455	FP
0.481	0.382	0.426	PT
0.434	0.407	0.42	AN
0.259	0.7	0.378	VO
0.5	0.222	0.308	XX
0.5	0.214	0.3	EF

Stanford NN Parser: UD Treebank

precision	recall	fscore	Deprel
0.999	1	0.999	punct
0.981	0.988	0.984	nmod:poss
0.986	0.98	0.983	cc
0.97	0.974	0.972	case
1	0.946	0.972	neg
0.965	0.961	0.963	det
0.962	0.962	0.962	name
0.936	0.965	0.95	amod
0.955	0.94	0.948	mark
0.95	0.939	0.944	aux
0.926	0.936	0.931	nummod
0.926	0.926	0.926	cop
0.962	0.889	0.924	compound:prt
0.909	0.909	0.909	advmod
0.902	0.91	0.906	nmod
0.896	0.868	0.882	nsubj
0.881	0.881	0.881	nsubjpass
0.85	0.875	0.862	dobj
0.842	0.85	0.846	acl:relcl
0.823	0.823	0.823	root
0.747	0.81	0.778	conj
0.833	0.714	0.769	nmod:agent
0.747	0.763	0.755	expl
0.805	0.684	0.74	mwe
0.66	0.701	0.68	advcl
0.645	0.709	0.675	ccomp
0.665	0.628	0.646	xcomp
0.8	0.5	0.615	csubjpass
0.75	0.5	0.6	csubj
0.667	0.4	0.5	auxpass
0.5	0.5	0.5	compound
0.737	0.378	0.5	iobj
0.476	0.472	0.474	appos
0.438	0.51	0.471	acl
0.407	0.412	0.41	parataxis
0.194	0.226	0.209	dislocated

Stanford NN Parser: Talbanken

precision	recall	fscore	Deprel
1	1	1	I?
0.999	1	1	IP
0.983	0.96	0.971	NA
0.933	0.977	0.955	JR
0.975	0.924	0.949	PL
0.952	0.935	0.943	IF
0.936	0.95	0.943	DT
0.889	1	0.941	IU
0.911	0.953	0.932	IR
0.91	0.948	0.929	AT
0.921	0.921	0.921	IK
0.918	0.918	0.918	IC
0.9	0.934	0.917	VG
0.9	0.928	0.914	PA
0.961	0.86	0.907	JC
0.853	0.967	0.906	IQ
0.888	0.884	0.886	SS
0.878	0.878	0.878	ROOT
0.859	0.859	0.859	IT
0.872	0.824	0.847	+A
0.714	1	0.833	XT
0.821	0.837	0.829	OO
0.798	0.839	0.818	UA
0.793	0.798	0.795	SP
0.762	0.799	0.78	CJ
0.699	0.768	0.731	ET
0.75	0.692	0.72	DB
0.797	0.642	0.711	CA
0.684	0.711	0.697	AA
0.814	0.608	0.696	FS
0.857	0.562	0.679	VA
0.757	0.609	0.675	HD
0.8	0.557	0.657	ES
0.688	0.624	0.654	MS
0.664	0.636	0.65	OA
0.576	0.658	0.615	RA
0.642	0.577	0.608	TA
0.875	0.412	0.56	IG
0.759	0.407	0.53	OP
0.789	0.395	0.526	IO
0.488	0.488	0.488	+F
0.409	0.6	0.486	AG
0.6	0.4	0.48	JT
0.455	0.5	0.476	VO
0.767	0.344	0.475	MA
0.483	0.452	0.467	KA
0.417	0.455	0.435	FP
0.5	0.364	0.421	VS
1	0.2	0.333	IS
0.667	0.222	0.333	XX
0.391	0.265	0.316	PT
0.305	0.319	0.312	AN
0.5	0.143	0.222	EF

Table 3. The chart of the dependency relation tags of the four models.

References

Ballesteros M., Nivre J. (2012) *MaltOptimizer: A System for MaltParser Optimization.* Proceedings of EACL'12 – Conference of the European Chapter of the Association for Computational Linguistics.

Chen D., Manning C. (2014) *A fast and accurate dependency parser using neural networks.* Proceedings of EMNLP'14.

Cirik V., Şensoy H. (2013) *The AI-KU System at the SPMRL 2013 Shared Task: Unsupervised Features for Dependency Parsing.* Proceedings of the 4th Workshop on Statistical Parsing of Morphologically Rich Languages.

McDonald R., Petrov S., Hall K. (2011) *Multi-Source Transfer of Delexicalized Dependency Parsers.* Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing.

McDonald R., Nivre J., Quirmbach-Brundage Y., Goldberg Y., Das D., Ganchev K., Hall K., Petrov S., Zhang H., Täckström O., Bedini C., Castello N., Lee J. (2013) *Universal Dependency Annotation for Multilingual Parsing.* Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics.

Nilsson J., Nivre J. (2008) *MaltEval: An Evaluation and Visualization Tool for Dependency Parsing.* LREC'08.

Nivre J., Megyesi B. (2007) *Bootstrapping a Swedish Treebank using cross-corpus harmonization and annotation projection.* Proceedings of TLT'07.

Nivre J., Hall J., Nilsson J., Chanev A., Eryiğit G., Kübler S., Marinov S., Marsi E. (2007) *MaltParser: A language-independent system for data-driven dependency parsing.* Natural Language Engineering.

Nivre J. (2015) *Towards a Universal Grammar of Natural Language Processing.* Computational

Linguistics and Intelligent Text Processing.

Nivre J., de Marneffe M.-C., Ginter F., Goldberg Y., Hajič J., Manning C., McDonald R., Petrov S., Pyysalo S., Silveira N., Tsarfaty R., Zeman D. (2016) *Universal Dependencies v1: A Multilingual Treebank Collection.* Proceedings of LREC'16.

Östling R. (2013) *Stagger: an Open-Source Part of Speech Tagger for Swedish.* Northern European Journal of Language Technology.

Tsarfaty R., Nivre J., Andersson E. (2011) *Evaluating Dependency Parsing: Robust and Heuristics-Free Cross-Annotation Evaluation.* Proceedings of the 8th International Conference on Empirical Methods in Natural Language Processing.

Zeman D., Mareček D., Popel M., Ramasamy L., Štěpanek J., Žabokrtsky Z., Hajič J. (2012) *HamleDT: To Parse or Not to Parse?* Proceedings of the 8th International Conference on Language Resources and Evaluation.

Zeman D. (2015) *Slavic Languages in Universal Dependencies.* Slovko'15 – Natural Language Processing, Corpus Linguistics, E-learning.