Improvements to Syntax-based Machine Translation using Ensemble Dependency Parsers

Nathan David Green and Zdeněk Žabokrtský
Charles University in Prague
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics
Prague, Czech Republic
{green,zabokrtsky}@ufal.mff.cuni.cz

Abstract

Dependency parsers are almost ubiquitously evaluated on their accuracy scores, these scores say nothing of the complexity and usefulness of the resulting structures. The structures may have more complexity due to their coordination structure or attachment rules. As dependency parses are basic structures in which other systems are built upon, it would seem more reasonable to judge these parsers down the NLP pipeline.

We show results from 7 individual parsers, including dependency and constituent parsers, and 3 ensemble parsing techniques with their overall effect on a Machine Translation system, Treex, for English to Czech translation. We show that parsers' UAS scores are more correlated to the NIST evaluation metric than to the BLEU Metric, however we see increases in both metrics.

1 Introduction

Ensemble learning (Dietterich, 2000) has been used for a variety of machine learning tasks and recently has been applied to dependency parsing in various ways and with different levels of success. (Surdeanu and Manning, 2010; Haffari et al., 2011) showed a successful combination of parse trees through a linear combination of trees with various weighting formulations. To keep their tree constraint, they applied Eisner's algorithm for reparsing (Eisner, 1996).

Parser combination with dependency trees has been examined in terms of accuracy (Sagae and Lavie, 2006; Sagae and Tsujii, 2007; Zeman and Žabokrtský, 2005; Holan and Žabokrtský, 2006). Other methods of parser combinations have shown to be successful such as using one parser to generate features for another parser. This was shown in (Nivre and McDonald, 2008), in which Malt Parser was used as a feature to MST Parser. The result was a successful combination of a transition-based and graph-based parser, but did not address adding other types of parsers into the framework.

We will use three ensemble approaches. First a fixed weight ensemble approach in which edges are added together in a weighted graph. Second, we added the edges using weights learned through fuzzy clustering based on POS errors. Third, we will use a meta-classifier that uses an SVM to predict the correct model for edge using only model agreements without any linguistic information added. Parsing accuracy and machine translation has been examined in terms of BLEU score (Quirk and Corston-Oliver, 2006). However, we believe our work is the first to examine the NLP pipeline for ensemble parsing for both dependency and constituent parsers as well as examining both BLEU and NIST scores' relationship to their *U*nlabeled *A*ccuracy *S*core(UAS).

2 Methodology

2.1 Annotation

To find the maximum effect that dependency parsing can have on the NLP pipeline, we annotated English dependency trees to form a gold standard. Annotation was done with two annotators using a tree editor, Tred (Pajas and Fabian, 2011), on data that was preprocessed using MST parser. For the annotation of our gold data, we used the standard developed by the Prague Dependency Treebank (PDT) (Hajič, 1998). PDT is annotated on three levels, morphological, analytical, and tectogrammatical. For our gold data we do not touch the morphological layer, we only correct the analytical layer (i.e. labeled dependency trees). For machine translation experiments later in the paper

Proceedings of the Second Workshop on Hybrid Approaches to Translation, pages 19–24,
Sofia, Bulgaria, August 8, 2013. ©2013 Association for Computational Linguistics

we allow the system to automatically generate a new tectogrammatical layer based on our new analytical layer annotation. Because the Treex machine translation system uses a tectogrammatical layer, when in doubt, ambiguity was left to the tectogrammatical (t-layer in Figure 1) to handle.

2.1.1 Data Sets

For the annotation experiments we use data provided by the 2012 Workshop for Machine Translation (WMT2012). The data which consists of 3,003 sentences was automatically tokenized, tagged, and parsed. This data set was also chosen since it is disjoint from the usual dependency training data, allowing researchers to use it as a out-of-domain testing set. The parser used was an implementation of MST parser. We then hand corrected the analytical trees to have a "Gold" standard dependency structure. Analytical trees were annotated on the PDT standard. Most changes involved coordination construction along with prepositional phrase attachment. We plan to publicly release this data and corresponding annotations in the near future[1].

Having only two annotators has limited us to evaluating our annotation only through spot checking and through comparison with other baselines. Annotation happened sequentially one after another. Possible errors were additionally detected through automatic means. As a comparison we will evaluate our gold data set versus other parsers in respect to their performance on previous data sets, namely the Wall Street Journal (WSJ) section 23.

2.2 Translation

2.2.1 Data Sets

All the parsers were trained on sections 02-21 of the WSJ, except the Stanford parser which also uses section 01. We retrained MST and Malt parsers and used pre-trained models for the other parsers. Machine translation data was used from WMT 2010, 2011, and 2012. Using our gold standard we are able to evaluate the effectiveness of different parser types from graph-base, transition-based, constituent conversion to ensemble approaches on the 2012 data while finding data trends using previous years data.

2.2.2 Translation Components

To examine the effects of dependency parsing down the NLP pipeline, we now turn to syntax based machine translation. Our dependency models will be evaluated using the Treex translation system (Popel and Žabokrtský, 2010). This system, as opposed to other popular machine translation systems, makes direct use of the dependency structure during the conversion from source to target languages via a tectogrammatical tree translation approach.

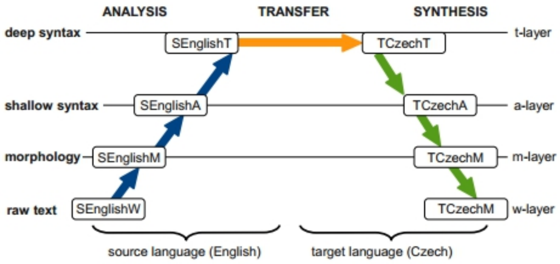

Figure 1: Treex syntax-based translation scenario (Popel and Žabokrtský, 2010)

We use the different parsers in separate translation runs each time in the same Treex parsing block. So each translation scenario only differs in the parser used and nothing else. As can be seen in Figure 1, we are directly manipulating the Analytical portion of Treex. The parsers used are as follows:

- **MST**: Implementation of Ryan McDonald's Minimum spanning tree parser (McDonald et al., 2005)
- **MST with chunking**: Same implementation as above but we parse the sentences based on chunks and not full sentences. For instance this could mean separating parentheticals or separating appositions (Popel et al., 2011)
- **Malt**: Implementation of Nivre's Malt Parser trained on the Penn Treebank (Nivre, 2003)
- **Malt with chunking**: Same implementation as above but with chunked parsing
- **ZPar**: Yue Zhang's statistical parser. We used the pretrained English model (english.tar.gz) available on the ZPar website for all tests (Zhang and Clark, 2011)
- **Charniak**: A constituent based parser (ec50spfinal model) in which we transform

[1] When available the data and description will be at www.nathangreen.com/wmtdata

the results using the Pennconverter (Johansson and Nugues, 2007)

- **Stanford**: Another constituent based parser (Klein and Manning, 2003) whose output is converted using Pennconverter as well (wsjPCFG.ser.gz model)

- **Fixed Weight Ensemble**: A stacked ensemble system combining five of the parsers above (MST, Malt, ZPar, Charniak, Stanford). The weights for each tree are assigned based on UAS score found in tuning data, section 22 of the WSJ (Green and Žabokrtský, 2012)

- **Fuzzy Cluster**: A stacked ensemble system as well but weights are determined by a cluster analysis of POS errors found in the same tuning data as above (Green and Žabokrtský, 2012)

- **SVM**: An ensemble system in which each individual edge is picked by a meta classifier from the same 5 parsers as the other ensemble systems. The SVM meta classifier is trained on results from the above tuning data (Green et al., 2012a; Green et al., 2012b).

2.2.3 Evaluation

For Machine Translation we report two automatic evaluation scores, BLEU and NIST. We examine parser accuracy using UAS. This paper compares a machine translation system integrating 10 different parsing systems against each other, using the below metrics.

The BLEU (*B*i*L*ingual *E*valuation *U*nderstudy) and NIST(from the *N*ational *I*nstitute of *S*tandards and *T*echnology), are automatic scoring mechanisms for machine translation that are quick and can be reused as benchmarks across machine translation tasks. BLEU and NIST are calculated as the geometric mean of n-grams multiplied by a brevity penalty, comparing a machine translation and a reference text (Papineni et al., 2002). NIST is based upon the BLEU n-gram approach however it is also weighted towards discovering more "informative" n-grams. The more rare an n-gram is, the higher the weight for a correct translation of it will be.

Made a standard in the CoNLL shared tasks competition, UAS studies the structure of a dependency tree and assesses how often the output has the correct head and dependency arcs (Buchholz and Marsi, 2006). We report UAS scores for each parser on section 23 of the WSJ.

3 Results and Discussion

3.1 Type of Changes in WMT Annotation

Since our gold annotated data was preprocessed with MST parser, our baseline system at the time, we started with a decent baseline and only had to change 9% of the dependency arcs in the data. These 9% of changes roughly increase the BLEU score by 7%.

3.2 Parser Accuracy

As seen in previous Ensemble papers (Farkas and Bohnet, 2012; Green et al., 2012a; Green et al., 2012b; Green and Žabokrtský, 2012; Zeman and Žabokrtský, 2005), parsing accuracy can be improved by combining parsers' outputs for a variety of languages. We apply a few of these systems, as described in Section 2.2.2, to English using models trained for both dependencies and constituents.

3.2.1 Parsers vs our Gold Standard

On average our gold data differed in head agreement from our base parser 14.77% of the time. When our base parsers were tested on the WSJ section 23 data they had an average error rate of 12.17% which is roughly comparable to the difference with our gold data set which indicates overall our annotations are close to the accepted standard from the community. The slight difference in percentage fits into what is expect in annotator error and in the errors in the conversion process of the WSJ by Pennconverter.

3.3 Parsing Errors Effect on MT

3.3.1 MT Results in WMT with Ensemble Parsers

WMT 2010

As seen in Table 1, the highest resulting BLEU score for the 2010 data set is from the fixed weight ensemble system. The other two ensemble systems are beaten by one component system, Charniak. However, this changes when comparing NIST scores. Two of the ensemble method have higher NIST scores than Charniak, similar to their UAS scores.

WMT 2011

The 2011 data corresponded the best with UAS scores. While the BLEU score increases for all

Parser	**UAS**	**NIST(10/11/12)**	**BLEU(10/11/12)**
MST	86.49	5.40/5.58/5.19	12.99/13.58/11.54
MST w chunking	86.57	5.43/5.63/5.23	13.43/14.00/11.96
Malt	84.51	5.37/5.57/5.14	12.90/13.48/11.27
Malt w chunking	87.01	5.41/5.60/5.19	13.39/13.80/11.73
ZPar	76.06	5.26/5.46/5.08	11.91/12.48/10.53
Charniak	92.08	5.47/5.65/5.28	13.49/13.95/**12.26**
Stanford	87.88	5.40/5.59/5.18	13.23/13.63/11.74
Fixed Weight	92.58	**5.49**/5.68/**5.29**	**13.53**/14.04/12.23
Fuzzy Cluster	92.54	5.47/5.68/5.26	13.47/14.06/12.06
SVM	92.60	5.48/**5.68**/5.28	13.45/**14.11**/12.22

Table 1: Scores for each machine translation run for each dataset (WMT 2010, 2011 and 2012)

the ensemble systems, the order of systems by UAS scores corresponds exactly to the systems ordered by NIST score and corelates strongly (Table 2). Unlike the 2010 data, the MST parser was the highest base parser in terms of the BLEU metric.

WMT 2012

The ensemble increases are statistically significant for both the SVM and the Fixed Weight system over the MST with chunking parser with 99% confidence, our previous baseline and best scoring base system from 2011 in terms of BLEU score. We examine our data versus MST with chunking instead of Charniak since we have preprocessed our gold data set with MST, allowing us a direct comparison in improvements. The fuzzy cluster system achieves a higher BLEU evaluation score than MST, but is not significant. In pairwise tests it wins approximately 78% of the time. This is the first dataset we have looked at where the BLEU score is higher for a component parser and not an ensemble system, although the NIST score is still higher for the ensemble systems.

	NIST	BLEU
2010	0.98	0.93
2011	0.98	0.94
2012	0.95	0.97

Table 2: Pearson correlation coefficients for each year and each metric when measured against UAS. Statistics are taken from the WMT results in Table 1. Overall NIST has the stronger correlation to UAS scores, however both NIST and BLEU show a strong relationship.

3.3.2 Human Manual Evaluation: SVM vs the Baseline System

We selected 200 sentences at random from our annotations and they were given to 7 native Czech speakers. 77 times the reviewers preferred the SVM system, 48 times they preferred the MST system, and 57 times they said there was no difference between the sentences. On average each reviewer looked at 26 sentences with a median of 30 sentences. Reviewers were allowed three options: sentence 1 is better, sentence 2 is better, both sentences are of equal quality. Sentences were displayed in a random order and the systems were randomly shuffled for each question and for each user.

	+	=	-
+	12	12	0
=		3	7
-			7

Table 3: Agreement for sentences with 2 or more annotators for our baseline and SVM systems. (-,-) all annotators agreed the baseline was better, (+,+) all annotators agreed the SVM system was better, (+,-) the annotators disagreed with each other

Table 3 indicates that the SVM system was preferred. When removing annotations marked as equal, we see that the SVM system was preferred 24 times to the Baseline's 14.

Although a small sample, this shows that using the ensemble parser will at worse give you equal results and at best a much improved result.

3.3.3 MT Results with Gold Data

In the perfect situation of having gold standard dependency trees, we obtained a NIST of 5.30 and a BLEU of 12.39. For our gold standard system run, the parsing component was removed and replaced with our hand annotated data. These are the highest NIST and BLEU scores we have obtained including using all base parsers or any combinations of parsers. This indicates that while an old problem which is a "solved" problem for some languages, Parsing is still worth researching and improving for its cascading effects down the NLP pipeline.

4 Conclusion

We have shown that ensemble parsing techniques have an influence on syntax-based machine translation both in manual and automatic evaluation. Furthermore we have shown a stronger correlation between parser accuracy and the NIST rather than the more commonly used BLEU metric. We have also introduced a gold set of English dependency trees based on the WMT 2012 machine translation task data, which shows a larger increase in both BLEU and NIST. While on some datasets it is inconclusive whether using an ensemble parser with better accuracy has a large enough effect, we do show that practically you will not do worse using one and in many cases do much better.

5 Acknowledgments

This research has received funding from the European Commission's 7th Framework Program (FP7) under grant agreement n° 238405 (CLARA). Additionally, this work has been using language resources developed and/or stored and/or distributed by the LINDAT-Clarin project of the Ministry of Education of the Czech Republic (project LM2010013).

References

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X Shared Task on Multilingual Dependency Parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, pages 149–164, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thomas G. Dietterich. 2000. Ensemble Methods in Machine Learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, pages 1–15, London, UK. Springer-Verlag.

Jason Eisner. 1996. Three New Probabilistic Models for Dependency Parsing: An Exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 340–345, Copenhagen, August.

Richárd Farkas and Bernd Bohnet. 2012. Stacking of Dependency and Phrase Structure Parsers. In *Proceedings of COLING 2012*, pages 849–866, Mumbai, India, December. The COLING 2012 Organizing Committee.

Nathan Green and Zdeněk Žabokrtský. 2012. Hybrid Combination of Constituency and Dependency Trees into an Ensemble Dependency Parser. In *Proceedings of the EACL 2012 Workshop on Innovative hybrid approaches to the processing of textual data*, Avignon, France.

Nathan Green and Zdeněk Žabokrtský. 2012. Ensemble Parsing and its Effect on Machine Translation. Technical Report 48.

Nathan Green, Septina Dian Larasati, and Zdeněk Žabokrtský. 2012a. Indonesian Dependency Treebank: Annotation and Parsing. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*, pages 137–145, Bali,Indonesia, November. Faculty of Computer Science, Universitas Indonesia.

Nathan Green, Loganathan Ramasamy, and Zdeněk Žabokrtský. 2012b. Using an SVM Ensemble System for Improved Tamil Dependency Parsing. In *Proceedings of the ACL 2012 Joint Workshop on Statistical Parsing and Semantic Processing of Morphologically Rich Languages*, pages 72–77, Jeju, Republic of Korea, July 12. Association for Computational Linguistics.

Gholamreza Haffari, Marzieh Razavi, and Anoop Sarkar. 2011. An Ensemble Model that Combines Syntactic and Semantic Clustering for Discriminative Dependency Parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 710–714, Portland, Oregon, USA, June. Association for Computational Linguistics.

Jan Hajič. 1998. Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. In Eva Hajičová, editor, *Issues of Valency and Meaning. Studies in Honor of Jarmila Panevová*, pages 12–19. Prague Karolinum, Charles University Press.

Tomáš Holan and Zdeněk Žabokrtský. 2006. Combining Czech Dependency Parsers. In *Proceedings of the 9th international conference on Text, Speech and Dialogue*, TSD'06, pages 95–102, Berlin, Heidelberg. Springer-Verlag.

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. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 423–430, Stroudsburg, PA, USA. Association for Computational Linguistics.

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, Morristown, NJ, USA. Association for Computational Linguistics.

Joakim Nivre and Ryan McDonald. 2008. Integrating Graph-Based and Transition-Based Dependency Parsers. In *Proceedings of ACL-08: HLT*, pages 950–958, Columbus, Ohio, June. Association for Computational Linguistics.

Joakim Nivre. 2003. An Efficient Algorithm for Projective Dependency Parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT*, pages 149–160.

Petr Pajas and Peter Fabian. 2011. TrEd 2.0 - newly refactored tree editor. http://ufal.mff.cuni.cz/tred/, Institute of Formal and Applied Linguistics, MFF UK.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Morristown, NJ, USA. Association for Computational Linguistics.

Martin Popel and Zdeněk Žabokrtský. 2010. TectoMT: modular NLP framework. In *Proceedings of the 7th international conference on Advances in natural language processing*, IceTAL'10, pages 293–304, Berlin, Heidelberg. Springer-Verlag.

Martin Popel, David Mareček, Nathan Green, and Zdenek Zabokrtsky. 2011. Influence of parser choice on dependency-based mt. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 433–439, Edinburgh, Scotland, July. Association for Computational Linguistics.

Chris Quirk and Simon Corston-Oliver. 2006. The impact of parse quality on syntactically-informed statistical machine translation. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 62–69, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kenji Sagae and Alon Lavie. 2006. Parser Combination by Reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 129–132, New York City, USA, June. Association for Computational Linguistics.

Kenji Sagae and Jun'ichi Tsujii. 2007. Dependency Parsing and Domain Adaptation with LR Models and Parser Ensembles. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 1044–1050, Prague, Czech Republic, June. Association for Computational Linguistics.

Mihai Surdeanu and Christopher D. Manning. 2010. Ensemble models for dependency parsing: cheap and good? In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 649–652, Stroudsburg, PA, USA. Association for Computational Linguistics.

Daniel Zeman and Zdeněk Žabokrtský. 2005. Improving Parsing Accuracy by Combining Diverse Dependency Parsers. In *In: Proceedings of the 9th International Workshop on Parsing Technologies.*

Yue Zhang and Stephen Clark. 2011. Syntactic Processing Using the Generalized Perceptron and Beam Search. *Computational Linguistics*, 37(1):105–151.