Improved Transition-Based Parsing and Tagging with Neural Networks

Chris Alberti **David Weiss** **Greg Coppola** **Slav Petrov**
Google Inc
New York, NY
{chrisalberti,djweiss,gcoppola,slav}@google.com

Abstract

We extend and improve upon recent work in structured training for neural network transition-based dependency parsing. We do this by experimenting with novel features, additional transition systems and by testing on a wider array of languages. In particular, we introduce set-valued features to encode the predicted morphological properties and part-of-speech confusion sets of the words being parsed. We also investigate the use of joint parsing and part-of-speech tagging in the neural paradigm. Finally, we conduct a multi-lingual evaluation that demonstrates the robustness of the overall structured neural approach, as well as the benefits of the extensions proposed in this work. Our research further demonstrates the breadth of the applicability of neural network methods to dependency parsing, as well as the ease with which new features can be added to neural parsing models.

1 Introduction

Transition-based parsers (Nivre, 2008) are extremely popular because of their high accuracy and speed. Inspired by the greedy neural network transition-based parser of Chen and Manning (2014), Weiss et al. (2015) and Zhou et al. (2015) concurrently developed structured neural network parsers that use beam search and achieve state-of-the-art accuracies for English dependency parsing.[1] While very successful, these parsers have made use only of a small fraction of the rich options provided inside the transition-based framework: for example, all of these parsers use virtually identical atomic features and the *arc-standard* transition system.

In this paper we extend this line of work and introduce two new types of features that significantly improve parsing performance: (1) a set-valued (i.e., bag-of-words style) feature for each word's morphological attributes, and (2) a weighted set-valued feature for each word's k-best POS tags. These features can be integrated naturally as atomic inputs to the embedding layer of the network and the model can learn arbitrary conjunctions with all other features through the hidden layers. In contrast, integrating such features into a model with discrete features requires non-trivial manual tweaking. For example, Bohnet and Nivre (2012) had to carefully discretize the real-valued POS tag score in order to combine it with the other discrete binary features in their system. Additionally, we also experiment with different transition systems, most notably the *integrated* parsing and part-of-speech (POS) tagging system of Bohnet and Nivre (2012) and also the *swap* system of Nivre (2009).

We evaluate our parser on the CoNLL '09 shared task dependency treebanks, as well as on two English setups, achieving the best published numbers in many cases.

2 Model

In this section, we review the baseline model, and then introduce the features (which are novel) and the transition systems (taken from existing work) that we propose as extensions. We measure the impact of each proposed change on the development sets of the multi-lingual CoNLL '09 shared task treebanks (Hajič et al., 2009). For details on our experimental setup, see Section 3.

2.1 Baseline Model

Our baseline model is the structured neural network transition-based parser with beam search of Weiss et al. (2015). We use a feed-forward network with embedding, hidden and softmax layers. The input consists of a sequence of matrices

[1] There is of course a much longer tradition of neural network dependency parsing models, going back at least to Titov and Henderson (2007).

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1354–1359,
Lisbon, Portugal, 17-21 September 2015. ©2015 Association for Computational Linguistics.

	Ca	Ch	Cz	En	Ge	Ja	Sp
Pipelined							
baseline	87.67	79.10	81.26	88.34	86.79	93.26	87.31
+morph	88.77	"	**84.50**	"	87.26	93.31	88.86
+morph +ktags	88.75	79.42	84.45	**88.62**	87.13	93.35	89.40
Integrated Tagging & Parsing							
+morph	88.93	79.71	84.41	88.57	87.07	93.32	89.35
+morph +ktags	**89.23**	**80.03**	84.27	88.55	**87.88**	**93.50**	**89.76**

Table 1: Ablation study on CoNLL'09 dev set. All scores in this table are LAS with beam 32. The first three rows use a pipeline of tagging and then parsing, while the last two rows use integrated parsing and tagging. Chinese and English have no morphology features provided in the dataset, so we omit morphology for those languages.

extracted deterministically from a transition-based parse configuration (consisting of a stack and a buffer). Each matrix $\mathbf{X}^g$, corresponds to a feature *group* g (one of *words*, *tags*, or *labels*), and has dimension $F^g \times V^g$. Here, X^g_{fv} is 1 if the f'th feature takes on value v for group g, i.e. each row $\mathbf{X}^g$ is a one-hot vector. These features are embedded and then concatenated to form the embedding layer, which in turn is input to the first hidden layer. The concatenated embedding layer can then be written as follows:

$$\mathbf{h}_0 = [\mathbf{X}^g\mathbf{E}^g \mid g \in \{\text{word}, \text{tag}, \text{label}\}] \quad (1)$$

where $\mathbf{E}^g$ is a (learned) $V^g \times D^g$ embedding matrix for group g, and D^g is the embedding dimension for group g. Beyond the embedding layer, there are two non-linear hidden layers (with non-linearity introduced using a rectified linear activation function), and a softmax layer that outputs class probabilities for each possible decision.

Training proceeds in two stages: We first train the network as a classifier by extracting decisions from gold derivations of the training set, as in Chen and Manning (2014). We then train a structured perceptron using the output of all network activations as features, as in Weiss et al. (2015). We use structured training and beam search during inference in all experiments. We train our models only on the treebank training set and do not use tri-training or other semi-supervised learning approaches (aside from using pre-trained word embeddings).

2.2 New Features

Prior work using neural networks for dependency parsing has not ventured beyond the use of one-hot feature activations for each feature type-location pair. In this work, we experiment with set-valued features, in which a set (or *bag*) of features for a given location fire at once, and are embedded into the same embedding space. Note that for both of the features we introduce, we extract features from the same 20 tokens as used in the *tags* and *words* features from Weiss et al. (2015), i.e. various locations on the stack and input buffer.

	Ca	Ch	Cz	En	Ge	Ja	Sp
CRF ($k = 1$)	98.60	93.19	98.25	97.55	96.73	97.47	98.02
Linear ($k = 4$)	98.75	93.71	98.48	97.70	97.27	97.75	98.33
Neural ($k = 4$)	**99.09**	**94.62**	**99.37**	**97.85**	**97.77**	**98.01**	**98.97**
BN'12 ($k = 3$)	-	93.06	99.32	97.77	97.63	-	-

Table 2: POS tagging results on the CoNLL '09 test set for integrated POS tagging and parsing. We compare the accuracy of our baseline CRF tagger, 'Linear' (our re-implementation of Bohnet and Nivre (2012, BN'12)), 'Neural' (the neural parser presented in this work), and results reported by BN'12.

Morphology. It is well known that morphological information is very important for parsing morphologically rich languages (see for example Bohnet et al. (2013)). We incorporate morphological information into our model using a set-valued feature function. We define the feature group *morph* as the matrix $\mathbf{X}^{morph}$ such that, for $1 \le f \le F^{morph}$, $1 \le v \le F^{morph}$,

$$\mathbf{X}^{morph}_{f,v} = \begin{cases} 1/N_f, & \text{token } f \text{ has attribute } v \\ 0, & \text{otherwise} \end{cases}, \quad (2)$$

where N_f is the number of morphological features active on the token indexed by f. In other words, we embed a bag of features into a shared embedding space by *averaging* the individual feature embeddings.

***k*-best Tags.** The non-linear network models of Weiss et al. (2015) and Chen and Manning (2014) embed the 1-best tag, according to a first-stage tagger, for a select set of tokens for any configuration. Inspired by the work of Bohnet and Nivre (2012), we embed the *set* of top tags according to a first-stage tagger. Specifically, we define the feature group *ktags* as the matrix $\mathbf{X}^{ktags}$ such that, for $1 \le f \le F^{ktags}$, $1 \le v \le V^{ktags}$,

$$\mathbf{X}^{ktags}_{f,v} = \begin{cases} P(POS = v \mid f), & v \in \text{top } k \text{ tags for } f \\ 0, & \text{otherwise} \end{cases}, \quad (3)$$

where $P(POS = v \mid f)$ is the marginal probability that the token indexed by f has the tag indexed by v, according to the first-stage tagger.

Method	B	Catalan UAS	Catalan LAS	Chinese UAS	Chinese LAS	Czech UAS	Czech LAS	English UAS	English LAS	German UAS	German LAS	Japanese UAS	Japanese LAS	Spanish UAS	Spanish LAS
Best Shared Task Result	-	-	87.86	-	79.17	-	80.38	-	89.88	-	87.48	-	92.57	-	87.64
Pipelined															
Zhang and McDonald (2014)	-	91.41	87.91	82.87	78.57	86.62	80.59	92.69	90.01	89.88	87.38	92.82	91.87	90.82	87.34
Lei et al. (2014)	-	91.33	87.22	81.67	76.71	88.76	81.77	92.75	90.00	90.81	87.81	**94.04**	91.84	91.16	87.38
This work linear	32	90.81	87.74	81.62	77.62	85.61	76.50	91.86	89.42	89.28	86.79	92.56	91.90	90.02	86.92
This work neural	32	92.31	89.17	83.34	79.50	88.35	83.50	92.37	90.21	90.12	87.79	93.99	**93.10**	91.71	88.68
Integrated Tagging & Parsing															
Bohnet and Nivre (2012)	40	92.02	88.97	81.18	77.00	88.07	82.70	92.06	89.54	90.43	88.23	93.67	92.63	91.43	88.54
Bohnet and Nivre (2012)+G+C	80	**92.44**	**89.60**	82.52	78.51	**88.82**	**83.73**	**92.87**	**90.60**	**91.37**	**89.38**	93.52	92.63	92.24	**89.60**
This work linear	32	91.02	87.98	82.26	78.32	85.73	78.37	91.57	88.83	88.80	86.38	93.28	92.38	90.24	87.09
This work neural	32	92.21	89.15	**83.57**	**79.90**	88.45	83.57	92.70	90.56	90.58	88.20	93.85	92.97	**92.26**	89.33

Table 3: Final CoNLL '09 test set results. The results not from this work were solicited from the respective authors.

Results. The contributions of our new features for pipelined arc-standard parsing are shown in Table 1. Morphology features (+*morph*) contributed a labeled accuracy score (LAS) gain of 2.9% in Czech, 1.5% in Spanish, and 0.9% in Catalan. Adding the k-best tag feature (+*morph* +*ktags*) provides modest gains (and modest losses), peaking at 0.54% LAS for Spanish. This feature proves more beneficial in the integrated transition system, discussed in the next section. We note the ease with which we can obtain these gains in a multilayer embedding framework, without the need for any hand-tuning.

2.3 Integrating Parsing and Tagging

While past work on neural network transition-based parsing has focused exclusively on the *arc-standard* transition system, it is known that better results can often be obtained with more sophisticated transition systems that have a larger set of possible actions. The *integrated arc-standard* transition system of Bohnet and Nivre (2012) allows the parser to participate in tagging decisions, rather than being forced to treat the tagger's tags as given, as in the arc-standard system. It does this by replacing the *shift* action in the arc-standard system with an action $shift_p$, which, aside from shifting the top token on the buffer also assigns it one of the k best POS tags from a first-stage tagger. We also experiment with the *swap* action of Nivre (2009), which allows reordering of the tokens in the input sequence. This transition system is able to produce non-projective parse trees, which is important for some languages.

Results. The effect of using the integrated transition system is quantified in the bottom part of Table 1. The use of both 1) +*morph* +*kbest* features and 2) integrated parsing and tagging achieves the best score for 5 out of 7 languages tested. The use of integrated parsing and tagging provides, for example, a 0.8% LAS gain in German.

3 Experiments

In this section we provide final test set results for our baseline and full models on three standard setups from the literature: *CoNLL '09*, *English WSJ* and *English Treebank Union*.

3.1 General Setup

To train with predicted POS tags, we use a CRF-based POS tagger to generate 5-fold jack-knifed POS tags on the training set and predicted tags on the dev, test and tune sets; our tagger gets comparable accuracy to the Stanford POS tagger (Toutanova et al., 2003) with 97.44% on the WSJ test set. The candidate tags allowed by the integrated transition system on every $shift_p$ action are chosen by taking the top 4 tags for a token according to the CRF tagger, sorted by posterior probability, with no minimum posterior probability for a tag to be selected. We report unlabeled attachment score (UAS) and labeled attachment score (LAS). Whether punctuation is included in the evaluation is specified in each subsection.

We use 1024 units in all hidden layers, a choice made based on the development set. We found network sizes to be of critical importance for the accuracy of our models. For example, LAS improvements can be as high as 0.98% in *CoNLL'09* German when increasing the size of the two hidden layers from 200 to 1024. We use $B = 16$ or $B = 32$ based on the development set performance per language. For ease of experimentation, we deviate from Bohnet and Nivre (2012) and use a single unstructured beam, rather than separate beams for POS tag and parse differences.

We train our neural networks on the standard training sets only, except for initializing with word

Method	B	UAS	LAS
Graph-based pipelined			
Bohnet (2010)	-	92.88	90.71
Martins et al. (2013)	-	92.89	90.55
Zhang and McDonald (2014)	-	93.22	91.02
Transition-based pipelined			
Zhang and Nivre (2011)	32	93.00	90.95
Bohnet and Kuhn (2012)	80	93.27	91.19
Chen and Manning (2014)	1	91.80	89.60
Dyer et al. (2015)	1	93.20	90.90
Weiss et al. (2015), supervised	8	93.99	92.05
Weiss et al. (2015), semi-sup.	8	**94.26**	**92.41**
Transition-based integrated			
Bohnet and Nivre (2012)	80	93.33	91.22
This work, supervised	32	**94.23**	**92.36**

Table 4: WSJ test set results on Stanford dependencies. Both the best supervised and semi-supervised results are bolded.

embeddings generated by word2vec and using cluster features in our POS tagger. Unlike Weiss et al. (2015) we train our model only on the treebank training set and do not use tri-training, which can likely further improve the results.

3.2 CoNLL '09

Our multilingual evaluation follows the setup of the CoNLL '09 shared task[2] (Hajič et al., 2009). As standard, we use the supplied predicted morphological features from the shared task data; however, we predict k-best tags with our own POS tagger since k-best tags are not part of the given data. We follow standard practice and include all punctuation in the evaluation. We used the (integrated) arc-standard transition system for all languages except for Czech where we added a swap transition, obtaining a 0.4% absolute improvement in UAS and LAS over just using arc-standard.

Results. In Table 3, we compare our models to the winners of the CoNLL '09 shared task, Gesmundo et al. (2009), Bohnet (2009), Che et al. (2009), Ren et al. (2009), as well as to more recent results on the same datasets. It is worth pointing out that Gesmundo et al. (2009) is itself a neural net parser. Our models achieve higher labeled accuracy than the winning systems in the shared task in all languages. Additionally, our pipelined neural network parser always outperforms its linear counterpart, an in-house reimplementation of the system of Zhang and Nivre (2011), as well as the more recent and highly accurate parsers of Zhang and McDonald (2014) and Lei et al. (2014). For the integrated models our neural network parser again outperforms its linear counterpart (Bohnet and Nivre, 2012), however, in some cases the addition of graph-based and cluster features (Bohnet and Nivre, 2012)+G+C can lead to even better results. The improvements in POS tagging (Table 2) range from 0.3% for English to 1.4% absolute for Chinese and are always higher for the neural network models compared to the linear models.

[2]http://ufal.mff.cuni.cz/conll2009-st/results/results.php

	News		Web		QTB	
Method	UAS	LAS	UAS	LAS	UAS	LAS
Bohnet (2010)	93.29	91.38	88.22	85.22	94.01	91.49
Martins et al. (2013)	93.10	91.13	88.23	85.04	94.21	91.54
Zhang et al. (2014)	93.32	91.48	88.65	85.59	93.37	90.69
Weiss et al. (2015)	93.91	92.25	89.29	86.44	94.17	92.06
This work (B=16)	**94.10**	**92.55**	**89.55**	**86.85**	**94.74**	**93.04**

Table 5: Final English Treebank Union test set results.

3.3 English WSJ

We experiment on English using the Wall Street Journal (WSJ) part of the Penn Treebank (Marcus et al., 1993), with standard train/test splits. We convert the constituency trees to Stanford style dependencies (De Marneffe et al., 2006) using version 3.3.0 of the converter. We use predicted POS tags and exclude punctuation from the evaluation, as is standard for English.

Results. The results shown in Table 4, we find that our full model surpasses, to our knowledge, all previously reported supervised parsing models for the Stanford dependency conversions. It surpasses its linear analog, the work of Bohnet and Nivre (2012) on Stanford Dependencies UAS by 0.9% UAS and by 1.14% LAS. It also outperforms the pipeline neural net model of Weiss et al. (2015) by a considerable margin and matches the semi-supervised variant of Weiss et al. (2015).

3.4 English Treebank Union

Turning to cross-domain results, and the "Treebank Union" datasets, we use an identical setup to the one described in Weiss et al. (2015). This setup includes the WSJ with Stanford Dependencies, the OntoNotes corpus version 5 (Hovy et al., 2006), the English Web Treebank (Petrov and McDonald, 2012), and the updated and corrected Question Treebank (Judge et al., 2006). We train on the union of each corpora's training set and test on each domain separately.

Results. The results of this evaluation are shown in Table 5. As for the WSJ we find that the integrated transition system combined with our novel

features performs better than previous work and in particular the model of Weiss et al. (2015), which serves as the starting point for this work. The improvements on the out-of-domain Web and Question corpora are particularly promising.

4 Conclusions

Weiss et al. (2015) presented a parser that advanced the state of the art for English Stanford dependency parsing. In this paper we showed that this parser can be significantly improved by introducing novel set features for morphology and POS tag ambiguities, which are added with almost no feature engineering effort. The resulting parser is already competitive in the multi-lingual setting of the CoNLL'09 shared task, but can be further improved by utilizing an integrated POS tagging and parsing transition system. We find that for all settings the dense neural network model produces higher POS tagging and parsing accuracy gains than its sparse linear counterpart.

Acknowledgements. We thank Bernd Bohnet for running his parser on additional data sets, and Emily Pitler for helpful comments.

References

Bernd Bohnet and Jonas Kuhn. 2012. The best of both worlds: a graph-based completion model for transition-based parsers. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 77–87.

Bernd Bohnet and Joakim Nivre. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465.

Bernd Bohnet, Joakim Nivre, Igor Boguslavsky, Richárd Farkas, Filip Ginter, and Jan Hajič. 2013. Joint morphological and syntactic analysis for richly inflected languages. *Transactions of the Association for Computational Linguistics*, 1:415–428.

Bernd Bohnet. 2009. Efficient parsing of syntactic and semantic dependency structures. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 67–72.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97.

Wanxiang Che, Zhenghua Li, Yongqiang Li, Yuhang Guo, Bing Qin, and Ting Liu. 2009. Multilingual dependency-based syntactic and semantic parsing. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 49–54.

Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 740–750.

Marie-Catherine De Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of Fifth International Conference on Language Resources and Evaluation*, pages 449–454.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 334–343.

Andrea Gesmundo, James Henderson, Paola Merlo, and Ivan Titov. 2009. A latent variable model of synchronous syntactic-semantic parsing for multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 37–42.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–18.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Short Papers*, pages 57–60.

John Judge, Aoife Cahill, and Josef van Genabith. 2006. Questionbank: Creating a corpus of parse-annotated questions. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 497–504.

Tao Lei, Yu Xin, Yuan Zhang, Regina Barzilay, and Tommi Jaakkola. 2014. Low-rank tensors for scoring dependency structures. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1381–1391.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Andre Martins, Miguel Almeida, and Noah A. Smith. 2013. Turning on the turbo: Fast third-order non-projective turbo parsers. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 617–622.

Joakim Nivre. 2008. Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, 34(4):513–553.

Joakim Nivre. 2009. Non-projective dependency parsing in expected linear time. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 351–359.

Slav Petrov and Ryan McDonald. 2012. Overview of the 2012 shared task on parsing the web. Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL).

Han Ren, Donghong Ji, Jing Wan, and Mingyao Zhang. 2009. Parsing syntactic and semantic dependencies for multiple languages with a pipeline approach. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 97–102.

Ivan Titov and James Henderson. 2007. Constituent parsing with incremental sigmoid belief networks. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 632–639.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 173–180.

David Weiss, Chris Alberti, Michael Collins, and Slav Petrov. 2015. Structured training for neural network transition-based parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 323–333.

Hao Zhang and Ryan McDonald. 2014. Enforcing structural diversity in cube-pruned dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 656–661.

Yue Zhang and Joakim Nivre. 2011. Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 188–193.

Hao Zhou, Yue Zhang, and Jiajun Chen. 2015. A neural probabilistic structured-prediction model for transition-based dependency parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 1213–1222.