Explicitly modeling case improves neural dependency parsing

Citation for published version:

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Publisher's PDF, also known as Version of record

Published In:
Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
Explicitly modeling case improves neural dependency parsing

Clara Vania and Adam Lopez
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
c.vania@ed.ac.uk, alopez@inf.ed.ac.uk

Abstract
Neural dependency parsing models that compose word representations from characters can presumably exploit morphosyntax when making attachment decisions. How much do they know about morphology? We investigate how well they handle morphological case, which is important for parsing. Our experiments on Czech, German and Russian suggest that adding explicit morphological case—either oracle or predicted—improves neural dependency parsing, indicating that the learned representations in these models do not fully encode the morphological knowledge that they need, and can still benefit from targeted forms of explicit linguistic modeling.

1 Introduction
Parsing morphologically rich languages (MRLs) is difficult due to the complex relationship of syntax to morphology. But the success of neural networks offer an appealing solution to this problem by computing word representation from characters. Character-level models (Ling et al., 2015; Kim et al., 2016) learn relationship between similar word forms and have shown to be effective for parsing MRLs (Ballesteros et al., 2015; Dozat et al., 2017; Shi et al., 2017; Björkelund et al., 2017). Does that mean that we can do away with explicit modeling of morphology altogether? Consider two challenges in parsing MRLs raised by Tsarfaty et al. (2010, 2013):

• Can we represent words abstractly so as to reflect shared morphological aspects between them?
• Which types of morphological information should we include in the parsing model?

It is tempting to hypothesize that character-level models effectively solve the first problem. For the second, Tsarfaty et al. (2010) and Seeker and Kuhn (2013) reported that morphological case is beneficial across morphologically rich languages with extensive case systems, where case syncretism is pervasive and often hurts parsing performance. But these studies focus on vintage parsers; do neural parsers with character-level representations also solve this second problem?

We attempt to answer this question by asking whether an explicit model of morphological case helps dependency parsing, and our results show that it does. Furthermore, a pipeline model in which we feed predicted case to the parser outperforms multi-task learning in which case prediction is an auxiliary task. These results suggest that neural dependency parsers do not adequately infer this crucial linguistic feature directly from the input text.

2 Dependency Parsing Model
We use a neural graph-based dependency parser similar to that of Kiperwasser and Goldberg (2016) and Zhang et al. (2017) for all our experiments. We treat our parser as a black box and experiment only with the input representations of the parser. Let \( w = w_1, \ldots, w_{|w|} \) be an input sentence of length \(|w|\) and let \( w_0 \) denote an artificial ROOT token. For each input token \( w_i \), we compute the context-independent representation, \( e(w_i) \), with a bidirectional LSTM (bi-LSTM) over characters. We concatenate the result with its part-of-speech (POS) representation, \( t_i \): \( x_i = [e(w_i); t_i] \). We then feed \( x_i \) to a word-level bi-LSTM encoder to learn a contextual word representation \( w_i \). The model uses these representations to compute the probability \( p(h_i, \ell_i \mid w, i) \) of head \( h_i \in \{0, ..., |w|\} \) and label \( \ell_i \) of word \( w_i \).

3 Experiments
We experiment with three fusional languages with extensive case systems: Czech, German, and Rus-
Language | Input              | Dev | Test  
---|-----------------|-----|-------
Czech | word           | 89.9 | 89.3  
(68.5K) | char            | 91.2 | 90.6  
| char (multi-task) | 91.6 | 91.0  
| char + predicted case | **92.2** | **91.8**  
| char + gold case | 92.3 | 91.9  
| char + full analysis | 92.5  | 92.0  
German | word           | 86.7 | 84.5  
(14.1K) | char            | 87.5 | 84.5  
| char (multi-task) | **87.9** | 84.4  
| char + predicted case | 87.8  | **86.4**  
| char + gold case | 90.2 | 86.9  
| char + full analysis | 89.7  | 86.5  
Russian | word           | 89.5 | 90.1  
(48.8K) | char            | 91.6 | 92.4  
| char (multi-task) | 92.2 | 92.6  
| char + predicted case | **92.5** | **93.3**  
| char + gold case | 92.8 | 93.5  
| char + full analysis | 92.6  | 93.3  

Table 1: Label Attachment Score (LAS) results. For each language, we show the number of training sentences.

<table>
<thead>
<tr>
<th>Language</th>
<th>%case</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Czech</td>
<td>66.5</td>
<td>95.4</td>
<td>96.7</td>
</tr>
<tr>
<td>German</td>
<td>36.2</td>
<td>92.6</td>
<td>92.0</td>
</tr>
<tr>
<td>Russian</td>
<td>55.8</td>
<td>95.8</td>
<td><strong>96.5</strong></td>
</tr>
</tbody>
</table>

Table 2: Case accuracy for case-annotated tokens, for pipeline (PL) vs. multitask (MT) setup. % case shows percentage of training tokens annotated with case.

We then look at the performance when we replace gold case with predicted case. We train a morphological tagger to predict case information. The tagger has the same structure as the parser’s encoder, with an additional feedforward neural network with one hidden layer followed by a softmax layer. We found that predicted case improves accuracy, although the effect is different across languages. These results are interesting, since in vintage parsers, predicted case usually harmed accuracy (Tsarfaty et al., 2010). However, we note that our taggers use gold POS, which might help.

**Pipeline model vs. Multi-task learning** In general, MTL models achieve similar or slightly better performance than the character-only models, suggesting that supplying case in this way is beneficial. However, we found that using predicted case in a pipeline model gives more improvements than MTL. We also observe an interesting pattern in which MTL achieves better tagging accuracy than the pipeline model but lower performance in parsing (Table 2). This is surprising since it suggests that the MTL model must learn to effectively encode case in the model’s representation, but must not effectively use it for parsing.

**4 Conclusion**

Vintage dependency parsers rely on hand-crafted feature engineering to encode morphology. The recent success of character-level models for many NLP tasks motivates us to ask whether their learned representations are powerful enough to completely replace this feature engineering. By empirically testing this using a single feature known to be important—morphological case—we have shown that they are not. Experiments with multi-task learning suggest that although MTL gives better performance, it is still underperformed by a traditional pipeline model.
References


