Do LSTMs really work so well for PoS tagging? –
A replication study

Tobias Horsmann and Torsten Zesch
Language Technology Lab
Department of Computer Science and Applied Cognitive Science
University of Duisburg-Essen, Germany
{tobias.horsmann,torsten.zesch}@uni-due.de

Abstract

A recent study by Plank et al. (2016) found that LSTM-based PoS taggers considerably improve over the current state-of-the-art when evaluated on the corpora of the Universal Dependencies project that use a coarse-grained tagset. We replicate this study using a fresh collection of 27 corpora of 21 languages that are annotated with fine-grained tagsets of varying size. Our replication confirms the result in general, and we additionally find that the advantage of LSTMs is even bigger for larger tagsets. However, we also find that for the very large tagsets of morphologically rich languages, hand-crafted morphological lexicons are still necessary to reach state-of-the-art performance.

1 Introduction

Part-of-Speech (PoS) tagging is an important processing step for many NLP applications. When researchers want to use a PoS tagger, they would ideally choose an off-the-shelf PoS tagger which is optimized for a specific language. If a suited tagger is not available two options remain: a) implementation of your own tagger, which requires technical knowledge and experience, or b) using an existing tagger and hope that the resulting model will be sufficiently accurate. One can assume that many taggers fit more languages than the one for which they have been constructed originally. Ideally, researchers should be able to fall back to a well-evaluated language-independent tagger if no reference implementation for a language is available.

A recent study by Plank et al. (2016) evaluated an LSTM PoS tagger and compared the results to Conditional Random Fields (CRF) (Lafferty et al., 2001) and Hidden-Markov (HMM) implementations on corpora of various languages. Their evaluation concludes that the LSTM tagger reaches better results than the CRF and HMM tagger. The evaluation corpora were all annotated with a coarse-grained tagset with 17 tags. Thus, this LSTM tagger seems to be a well-performing, language-independent choice for learning models on coarse-grained tagsets. While for many tasks a coarse-grained tagset might be sufficient some tasks require more fine-grained tagsets.

We, thus, consider it worthwhile to explore if the results are reproducible using corpora with fine-grained tagsets. We use the LSTM tagger provided by Plank et al. (2016) and compare the results likewise to CRF and an off-the-shelf HMM tagger implementation. We compile a fresh set of 27 corpora of 21 languages which uses the commonly used fine-grained tagset of the respective language. We suggest these corpora as evaluation set for tasks which require fine-grained PoS tags, as all corpora are freely available for research purposes. Our intention is to replicate the findings of Plank et al. (2016), which have been achieved on a coarse-grained tagset and investigate if they transfer to fine-grained tagsets.

2 PoS Tagger Paradigms

We distinguish two PoS tagger paradigms, which can be used to implement a tagger: The first one is Feature Engineering, in which a classifier learns a mapping from human-defined features to a PoS tag. Defining good features is often a non-trivial task, which furthermore requires a lot of experience. For instance a suffix feature which checks a word-ending for “ing” is highly discriminative for English gerunds, but might not provide any useful information for other languages. The details of the feature implementation might render a
tagger unsuited for learning models for other languages or tagsets. We will, thus, experiment with features and their configurations, and investigate how well they perform in combination for learning fine-grained tagsets of various languages. We implement those experiments using CRF which are frequently used for PoS tagging (Remus et al., 2016; Ljubešić et al., 2016).

The second paradigm is Architecture Engineering, which relies on methods to learn the input representation by themselves. The challenge lies in finding an architecture that supports this self-learning process. Most recent representatives of this paradigm are neural networks of which we use the LSTM tagger provided by Plank et al. (2016).

In our experiments, we will focus on how to provide word- and character-level information to the classifiers as these two types of information are most relevant and most frequently used for training PoS tagger models. Furthermore, we will evaluate the performance on Out-Of-Vocabulary (OOV) words to learn if the taggers generalize to unseen words.

To provide a reference value to a well-known PoS tagger, we will compare all results to the HMM-based HunPos (Halácsy et al., 2007) tagger, which is a freely available re-implementation of the TNT tagger (Brants, 2000). HunPos has been used before for training models of various languages and tagsets (Seraji, 2011; Attardi et al., 2010; Hládek et al., 2012) which is why we consider this tagger to be a suitable baseline.

### 3 Evaluation Corpora Dataset

Table 1 shows the fine-grained annotated corpora we collected by screening the literature. We do not claim that this list is complete, but the provided corpora are all reasonably easy to access and can be freely used for research purposes.

**Selection** To ensure reproducibility, we preferably selected corpora which are directly available via the Internet except German-3, Hungarian and Swedish-2. We intentionally exclude languages such as Chinese or Japanese, which do not provide whitespace delimiters to mark word boundaries. Tagging those languages requires a morpho-

---

**Table 1: Corpora used in our experiments**

<table>
<thead>
<tr>
<th>Group</th>
<th>Corpus Id</th>
<th>Source</th>
<th>Tokens (10^3)</th>
<th># Tags</th>
<th>Annotation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Germanic</td>
<td>Danish</td>
<td>Copenhagen DTB</td>
<td>255</td>
<td>36</td>
<td>manual</td>
<td>(Buch-Kromann and Korzen, 2010)</td>
</tr>
<tr>
<td></td>
<td>Dutch</td>
<td>Alpino</td>
<td>200</td>
<td>20</td>
<td>manual</td>
<td>(Bouma et al., 2000)</td>
</tr>
<tr>
<td></td>
<td>English</td>
<td>Brown</td>
<td>1,100</td>
<td>180</td>
<td>manual</td>
<td>(Nelson Francis and Kuçera, 1964)</td>
</tr>
<tr>
<td></td>
<td>German-1</td>
<td>Hamburg DTB</td>
<td>4,800</td>
<td>54</td>
<td>manual</td>
<td>(Brants et al., 2004)</td>
</tr>
<tr>
<td></td>
<td>German-2</td>
<td>Tiger</td>
<td>880</td>
<td>54</td>
<td>manual</td>
<td>(Telljohann et al., 2004)</td>
</tr>
<tr>
<td></td>
<td>German-3</td>
<td>Tüba-DZ</td>
<td>1,500</td>
<td>54</td>
<td>manual</td>
<td>(Foth et al., 2014)</td>
</tr>
<tr>
<td></td>
<td>Icelandic</td>
<td>Min</td>
<td>1,000</td>
<td>703</td>
<td>auto</td>
<td>(Helgadóttir et al., 2012)</td>
</tr>
<tr>
<td></td>
<td>Norwegian</td>
<td>Norwegian DTB</td>
<td>1,300</td>
<td>19</td>
<td>manual</td>
<td>(Solberg et al., 2014)</td>
</tr>
<tr>
<td></td>
<td>Swedish-1</td>
<td>Talbanken</td>
<td>96</td>
<td>25</td>
<td>manual</td>
<td>(Einarsdon, 1976)</td>
</tr>
<tr>
<td></td>
<td>Swedish-2</td>
<td>Stockholm-Umea</td>
<td>1,100</td>
<td>153</td>
<td>manual</td>
<td>(Ejerhed and Källgren, 1997)</td>
</tr>
<tr>
<td></td>
<td>Braz.Portuguese</td>
<td>MAC-Morpho</td>
<td>1,000</td>
<td>82</td>
<td>manual</td>
<td>(Aluísio et al., 2003)</td>
</tr>
<tr>
<td></td>
<td>French-1</td>
<td>Multitag</td>
<td>370</td>
<td>992</td>
<td>manual</td>
<td>(Paroubek, 2000)</td>
</tr>
<tr>
<td></td>
<td>French-2</td>
<td>Sequoia</td>
<td>200</td>
<td>29</td>
<td>manual</td>
<td>(Candito et al., 2014)</td>
</tr>
<tr>
<td></td>
<td>Italian</td>
<td>Turin Parallel</td>
<td>80</td>
<td>15</td>
<td>auto</td>
<td>(Bosco et al., 2012)</td>
</tr>
<tr>
<td></td>
<td>Spanish</td>
<td>IULA DTB</td>
<td>550</td>
<td>241</td>
<td>manual</td>
<td>(Marimon et al., 2014)</td>
</tr>
<tr>
<td></td>
<td>Croatian-1</td>
<td>Croatian DTB</td>
<td>200</td>
<td>692</td>
<td>manual</td>
<td>(Željko Agić and Ljubešić, 2014)</td>
</tr>
<tr>
<td></td>
<td>Croatian-2</td>
<td>Hr500k</td>
<td>500</td>
<td>769</td>
<td>manual</td>
<td>(Ljubešić et al., 2016)</td>
</tr>
<tr>
<td></td>
<td>Czech</td>
<td>Prague DTB</td>
<td>2,000</td>
<td>1,574</td>
<td>manual</td>
<td>(Bejček et al., 2013)</td>
</tr>
<tr>
<td></td>
<td>Polish</td>
<td>Polish National Corpus</td>
<td>1,000</td>
<td>27</td>
<td>manual</td>
<td>(Przepiórkowski et al., 2008)</td>
</tr>
<tr>
<td></td>
<td>Russian</td>
<td>Russian Open Corpus</td>
<td>1,700</td>
<td>22</td>
<td>manual</td>
<td>(Bocharov et al., 2013)</td>
</tr>
<tr>
<td></td>
<td>Slovak</td>
<td>MULTTEXT-East</td>
<td>84</td>
<td>956</td>
<td>manual</td>
<td>(Erjavec, 2010)</td>
</tr>
<tr>
<td></td>
<td>Slovene-1</td>
<td>IJS-ELAN</td>
<td>540</td>
<td>1,181</td>
<td>auto</td>
<td>(Erjavec, 2002)</td>
</tr>
<tr>
<td></td>
<td>Slovene-2</td>
<td>SSJ</td>
<td>590</td>
<td>1,304</td>
<td>manual</td>
<td>(Krek et al., 2013)</td>
</tr>
<tr>
<td></td>
<td>Afrikaans</td>
<td>AfriBooms</td>
<td>50</td>
<td>12</td>
<td>manual</td>
<td>(Augustinus et al., 2016)</td>
</tr>
<tr>
<td></td>
<td>Finnish</td>
<td>FinnTreebank</td>
<td>170</td>
<td>1573</td>
<td>manual</td>
<td>(Voutilainen, 2011)</td>
</tr>
<tr>
<td></td>
<td>Hebrew</td>
<td>HaAretz Corpus</td>
<td>11,000</td>
<td>22</td>
<td>auto</td>
<td>(Itai and Winter, 2008)</td>
</tr>
<tr>
<td></td>
<td>Hungarian</td>
<td>The Szeged Treebank</td>
<td>1,200</td>
<td>1,085</td>
<td>manual</td>
<td>(Csenedes et al., 2005)</td>
</tr>
</tbody>
</table>
We use Brown clustering (Brown et al., 1992) to create word clusters. The
language analysis which is a different task than the
tagging task on which we are focusing here. Most
corpora are manually annotated or were at least
human-verified. There are four exceptions which
we decided to add anyway to increase the number
of languages represented in our setup. The tagset
granularity of the corpora ranges from coarse (12
tags) to morphologically fine (1574 tags) to evalu-
ate all taggers on various stages of granularity.

Language & Corpora Diversity We analyzed
the distribution of PoS tags in the corpora by map-
ing all tags to the 17 coarse-grained PoS tags of
the Universal Dependencies (UD) project (Nivre
et al., 2015) in Figure 1. The mappings to the
UD tagset have been manually created. The partly
large differences between the syntactical classes
help to better understand the challenge in con-
struction a tagger that is suited for all those lan-
guages. For instance, Germanic and Romanic lan-
guages have a lot of determiners while they do not
occur at all in Slavic languages.

Corpus Size & Tagset The corpora have vary-
ing sizes which makes a direct comparison be-
tween corpora difficult. To run our experiments
under fully controlled conditions, we extract a ran-
domized sub-sample of sentences from each cor-
pus, which accounts for 50k tokens, and run all our
experiments with 10fold cross-validation (CV). Results reported use the fine-grained tagset of the respective corpus.

We deliberately do not use the corpora from the
UD Treebank project in order to provide results on a fresh dataset. Additionally, UD uses a coarse-
grained tagset for all its corpora. While this gran-
ularity is sufficient for many tasks, linguistic anal-
ysis often requires more fine-grained tagsets, and
it is not clear whether results achieved on coarse-
grained tagsets transfer well to more fine-grained
tagsets. The collected corpora, thus, also represent
an alternative dataset, which we suggest in case
the UD tagset is too coarse-grained.

4 CRF Experiments

We reviewed the recent literature to determine the
most commonly used features for training PoS tag-
gers. As re-occurring features, we found word
grams, fixed character sequences focusing on ei-
ther pre-, in-, or suffixes of words and word dis-
tributional knowledge for PoS taggers of various
languages (Brants, 2000; Horsmann and Zesch,
2016; Ljubešić et al., 2016). Word- and character-
grams have been used with various parametriza-
tions depending on the language and there is no
agreement which parameters are most advisable.
We will, hence, run a series of parameter-search
experiments over the word- and character-ngram
parametrization to determine a configuration ap-
plicable to all languages. For this, we evaluate all
permutations of the subsequently introduced fea-
ture configurations with 10fold cross-validation.
The objective is to find a configuration that works
well on all corpora, languages, and tagsets.

Word Features We experiment with adding the
1, 2, 3 words to the right and left of the current
word as lower-cased string features.

Character Features Which character-ngram is
discriminative for a PoS tag strongly depends on
the language. To avoid a language bias, we use a
frequency-based approach in which we select the
N most frequently occurring character-ngrams of
length 1, 2, 3, 4 from the training dataset. We ex-
periment with the following frequency cut-off val-
ues of $N \in \{250, 500, 750, 1000\}$ to select only
frequent and potentially informative character-
genres as features. These $N$ features are boolean
and are set to 1 if the respective character-ngram
occurs in the current word.

Semantic Features We use Brown clustering
(Brown et al., 1992) to create word clusters. The

Figure 1: Coarse-grained PoS tag distribution of
corpora by language group

![Graph showing coarse-grained PoS tag distribution by language group.](image-url)
<table>
<thead>
<tr>
<th>Lang. Group</th>
<th>Corpus Id</th>
<th>Word Ngrams ±1 Char Ngrams Clusters</th>
<th>Best CRF</th>
<th>HunPos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Germanic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Danish</td>
<td></td>
<td>90.9 53.3 90.3 69.3 89.5 67.6</td>
<td>96.1 82.4</td>
<td>94.9 74.2</td>
</tr>
<tr>
<td>Dutch</td>
<td></td>
<td>86.5 66.9 85.0 71.7 88.0 77.7</td>
<td>90.7 83.7</td>
<td>89.9 80.6</td>
</tr>
<tr>
<td>English</td>
<td></td>
<td>87.5 45.1 90.3 70.1 89.1 64.0</td>
<td>94.6 80.2</td>
<td>93.8 77.7</td>
</tr>
<tr>
<td>German-1</td>
<td></td>
<td>88.5 62.4 90.3 77.7 90.8 73.7</td>
<td>94.6 84.6</td>
<td>94.4 83.7</td>
</tr>
<tr>
<td>German-2</td>
<td></td>
<td>87.2 60.3 90.9 77.7 90.8 76.1</td>
<td>95.2 87.1</td>
<td>94.9 85.4</td>
</tr>
<tr>
<td>German-3</td>
<td></td>
<td>86.3 58.5 91.7 76.8 91.6 77.6</td>
<td>94.4 85.0</td>
<td>94.4 83.9</td>
</tr>
<tr>
<td>Icelandic</td>
<td></td>
<td>67.5 14.2 76.5 45.1 68.3 28.9</td>
<td>80.9 53.6</td>
<td>79.8 51.9</td>
</tr>
<tr>
<td>Norwegian</td>
<td></td>
<td>92.4 77.1 91.6 80.6 92.8 82.7</td>
<td>96.1 89.7</td>
<td>95.5 86.5</td>
</tr>
<tr>
<td>Swedish-1</td>
<td></td>
<td>91.1 70.6 92.9 82.2 92.3 79.9</td>
<td>96.3 90.3</td>
<td>95.6 85.9</td>
</tr>
<tr>
<td>Swedish-2</td>
<td></td>
<td>78.7 29.7 87.2 67.3 81.4 48.8</td>
<td>91.0 74.6</td>
<td>91.4 77.6</td>
</tr>
<tr>
<td>Romanc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B-Portug.</td>
<td></td>
<td>86.9 62.8 87.8 73.6 89.7 76.0</td>
<td>92.8 83.8</td>
<td>93.3 84.2</td>
</tr>
<tr>
<td>French-1</td>
<td></td>
<td>81.9 40.1 85.9 66.5 81.6 58.2</td>
<td>89.2 75.7</td>
<td>88.2 71.8</td>
</tr>
<tr>
<td>French-2</td>
<td></td>
<td>95.4 67.3 93.8 74.5 91.9 79.3</td>
<td>97.7 88.2</td>
<td>97.4 82.4</td>
</tr>
<tr>
<td>Italian</td>
<td></td>
<td>93.3 68.6 91.6 74.8 91.7 75.5</td>
<td>96.4 86.5</td>
<td>95.8 80.8</td>
</tr>
<tr>
<td>Spanish</td>
<td></td>
<td>88.5 45.5 94.5 78.2 88.1 58.8</td>
<td>96.4 83.5</td>
<td>96.6 83.6</td>
</tr>
<tr>
<td>Slavic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Croatian-1</td>
<td></td>
<td>69.0 18.6 80.6 56.3 75.2 47.2</td>
<td>84.9 65.4</td>
<td>84.7 66.7</td>
</tr>
<tr>
<td>Croatian-2</td>
<td></td>
<td>66.3 15.9 78.5 54.4 73.5 44.8</td>
<td>83.4 63.9</td>
<td>82.6 63.9</td>
</tr>
<tr>
<td>Czech</td>
<td></td>
<td>64.1 14.4 79.2 56.0 75.2 39.2</td>
<td>83.1 62.9</td>
<td>81.7 60.9</td>
</tr>
<tr>
<td>Polish</td>
<td></td>
<td>82.9 58.1 92.5 86.9 86.5 72.5</td>
<td>95.5 91.5</td>
<td>93.6 85.4</td>
</tr>
<tr>
<td>Russian</td>
<td></td>
<td>83.7 53.7 93.0 83.5 88.2 70.9</td>
<td>95.5 87.5</td>
<td>94.6 83.6</td>
</tr>
<tr>
<td>Slovak</td>
<td></td>
<td>67.7 14.9 80.5 57.8 65.6 31.9</td>
<td>83.5 63.8</td>
<td>82.9 61.6</td>
</tr>
<tr>
<td>Slovene-1</td>
<td></td>
<td>72.6 17.4 83.5 55.6 72.4 39.4</td>
<td>86.4 62.5</td>
<td>82.6 59.6</td>
</tr>
<tr>
<td>Slovene-2</td>
<td></td>
<td>65.4 12.1 78.2 50.5 73.0 39.0</td>
<td>83.0 59.4</td>
<td>86.2 59.5</td>
</tr>
<tr>
<td>Other</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Afrikaans</td>
<td></td>
<td>95.7 75.0 95.3 80.3 95.8 81.9</td>
<td>97.8 89.6</td>
<td>97.3 85.5</td>
</tr>
<tr>
<td>Finnish</td>
<td></td>
<td>62.6 10.0 77.1 48.5 67.8 33.8</td>
<td>82.3 56.7</td>
<td>81.3 55.8</td>
</tr>
<tr>
<td>Hebrew</td>
<td></td>
<td>82.3 41.7 81.3 60.9 76.3 53.3</td>
<td>90.5 68.5</td>
<td>90.3 60.1</td>
</tr>
<tr>
<td>Hungarian</td>
<td></td>
<td>72.7 13.9 86.7 63.3 72.0 31.7</td>
<td>89.9 69.6</td>
<td>89.4 69.5</td>
</tr>
</tbody>
</table>

Table 2: Accuracy of CRF taggers (10fold CV)

unlabelled text is obtained from the Leipzig Corpus Collection (Quasthoff et al., 2006), which provides large text quantities crawled from the web for many languages. We use 15 · 10^6 tokens to create the clusters from the same amount of text for all languages. We provide the cluster ids in substrings of varying length to the classifier (Owoputi et al., 2013).

Results In Figure 2, we show the results of our parameter search experiment. The triangles mark the results of the various feature configurations. The diamond symbol shows the configuration which works best over all corpora. We refer to this best working configuration as Best CRF subsequently, it uses a word-context window of 1 word to the left and right and the 750 most frequent character [1..4] grams with additionally adding word clusters. Especially for morphologically-rich languages, the spread is quite large which is caused by the lower number of character-ngrams in those configurations. For corpora such as Slovene-1, we see that more accurate configurations exist than Best CRF but more importantly, the selected configuration is always among the best working ones.

We show the results of Best CRF and the performance of the individual features for each language in Table 2, and compare the results to HunPos, the highest accuracies are highlighted in grey. When evaluating the features separately, the character-ngrams reach the highest accuracy on OOV words. Especially on the Slavic language family the character-ngrams perform much better than using only word-ngrams or clusters. Furthermore, using only character-ngrams is often competitive to using only word-ngrams. Hence, a rather naive strategy to achieving a decent performance on almost any language is to just use

741
all kinds of character-ngrams. The cluster feature also performs better than the word-ngrams. Considering that we had to limit the amount of data for creating the clusters for comparability, this feature assumedly has more potential when using larger data sizes (Derczynski et al., 2015). The combination of all features in the column Best CRF shows that the features address quite different information and add up well, so unsurprisingly, this configuration reaches the overall best accuracies. The difference to HunPos is, with often less than one percent point difference, only small. Off-the-shelf taggers do, hence, not necessarily have a disadvantage over constructing an own tagger. In the remainder of this work, we will use the Best CRF configuration when discussing CRF tagger results.

5 LSTM Experiments

When using neural networks, the details of how word and character information is provided greatly influences the learning success of the network. We will reproduce network setups which have also been used in Plank et al. (2016) to ensure comparability to the coarse-grained results to which we compare our results:

Word  In this setup, we train a network on the word embeddings only and provide them to a bidirectional LSTM. This setup will serve as baseline.

Char  The character embeddings of a word are provided to a bidirectional LSTM. The last state of the forward and the backward character LSTM are combined (Ling et al., 2015) and provided to another bidirectional LSTM layer.

Word-Char  This architecture is a combination of the previous two architectures. The last state of the character LSTMs is added to the word embedding information before it is provided to the next LSTM layer.

Word-Char+  The architecture by Plank et al. (2016) combines word and character level information and additionally considers the log-frequency of the next word during training. This tagger reported state-of-the-art results and we use the provided reference implementation of this tagger in our setup.

LSTMs have the reputation to require larger amounts of training data. With the 50k tokens we use this is barely fulfilled, however, Plank et al. (2016) find this sensitivity to be less severe and set a corpus size of 60k tokens as lower bound for their coarse-grained tagging experiments. We will come back to this data size issue in Section 7, where we evaluate using all tokens in a corpus (and arriving at the same conclusions as for our 50k token datasets). Furthermore, in many cases only smaller dataset sizes are available, sometimes even less than 50k tokens. It is, thus, important to know if considering neural network taggers makes sense at all (on fine-grained tagsets), thus we will train LSTM models on smaller dataset sizes.

We implement the LSTM taggers in DyNet (Neubig et al., 2017) and use the hyper-parameter settings by Plank et al. (2016), i.e. we train 20 epochs using Statistical-Gradient-Descent with a learning rate of 0.1 and adding Gaussian noise of 0.2 to the embedding layer. We train word embeddings on the data we already used for the semantic feature in the CRF experiments by using fastText (Bojanowski et al., 2016). The character-level embeddings are trained on-the-fly.

Results  In Figure 3, we show the results for the LSTM architectures. The Word-Char+ tagger performs best followed by Word-Char, which is not surprising as Word-Char+ is based on this

Figure 2: Variance of CRF taggers (10fold CV)
architecture. For the Germanic and Romance languages, the accuracy of the various architectures is similar but for Slavic languages, which use much more fine-grained tagsets, the differences are rather large. For instance, the Char architecture reaches only small improvements over the Word baseline on Croatian or Czech while on Spanish, or Hungarian the character architecture is clearly better than the baseline. Table 3 shows the detailed results and additionally reports the accuracy values on OOV with best results highlighted in grey. The Char architecture is in many cases competitive to the HunPos reference system. This shows that the performance of many off-the-shelf taggers is rather easy to approximate by relying only on character-level information.

The results by the Char architecture also explain why the Word-Char architecture performs so well although the amount of syntactical information is quite limited with 50k tokens. A large part of the necessary information is already obtained by the character model, which requires a lot less training data than a model on the word level. Thus, the results of Plank et al. (2016) on coarse-tagsets are reproducible for fine-grained tagsets with the Word-Char architecture being the essential property to achieving high accuracy.

6 Influence of Tagset Size

A researcher who works with morphologically rich languages will often be interested in additional morphologic details such as case or gender. This drastically complicates the task, as a few hundred instead of a few dozen PoS tag distinctions have to be learned. In this experiment, we will examine the impact of an increasing number of PoS tags on the accuracy of the taggers to provide reference values of how much performance a tagger seems to loose with an increasing tagset size.

Results In Figure 4, we show a comparison of the tagging accuracy in relation to the number of PoS tags. We show the best performing LSTM tagger Word-Char+, the CRF tagger and HunPos. Each data point represents the averaged CV result on one corpus with the respective tagger. We see a certain clustering of the data points for the small tagset sizes, which shows that the taggers tend to perform highly similarly for many languages. This means that the tagset size has a larger effect on the accuracy than the language of the corpus.

For each PoS tagger, a regression trendline is plotted which indicates the average loss in accuracy with an increasing tagset size. For one-hundred additional PoS tags, Word-Char+ loses 0.35 points in accuracy, while CRF and HunPos have a much steeper decay of 0.45 points. Hence, with growing tagset size the tagger choice becomes increasingly more important. Furthermore, the benefit of more sophisticated tagger architectures becomes only apparent on large PoS tagsets.

7 Comparison with Reference Taggers

In this experiment, we compare our results to reference taggers from the literature that are tailored towards certain languages. Our experiments until now were limited to the fixed dataset size that we set at the beginning for comparability. Especially for the morphologically fine-grained tagsets this might have been problematic, as it is doubtful if all PoS tags of a morphological tagset do even occur on 50k tokens. Thus, in order to evaluate the taggers using all available data, we will reproduce setups reported in the literature and compare the performance of the taggers to those results.

This experiment limits the number of comparisons we can make drastically, as we need to have...
the same corpora as used in the literature. We, thus, reproduce for Czech the setup by Spoustová et al. (2009) with training on $10^6$ and evaluation on $2 \cdot 10^5$ tokens, for German-2 the setup by Giesbrecht and Evert (2009) and for Swedish-2 the setup by Östling (2013), which both use 10fold cross-validation over the full corpus size.

Taggers for Slavic languages often make use of additional resources such as morphological dictionaries, which we intentionally do not include to avoid human-crafted resources that are not available for all languages. Thus, we do not expect to reach state-of-the-art performance, but we want to quantify the size of the gap.

### Results

In Table 4, we show a comparison of our results to the results reported in the literature. On German-2 and Swedish-2, the Word-Char+ tagger is able to reach better results than the reported reference values except for Czech which uses a morphologically fine-grained tagset. Thus, language-
fitted PoS taggers reach better results than neural networks when training models on corpora with extremely fine-grained PoS tagsets. However, for smaller tagset sizes the need for using language-fitting is negligible.

8 Conclusion

We replicated a study in which LSTM PoS taggers are compared to CRF and HMM taggers on corpora with a coarse-grained tagset. Our replication focused on whether results reported for coarse-grained tagsets do also hold when training models on fine-grained tagsets. Therefore, we collected a large set of 27 evaluation corpora that are annotated with the commonly used fine-grained tagset of 21 languages. The replication confirmed the superior performance of the LSTTagger reported by Plank et al. (2016) also on fine-grained tagsets. However, we also found that for smaller tagset sizes the differences between the LSTM, our self-implemented CRF and the HMM tagger are often only small. The advantages of the LSTM tagger over other taggers grow proportionally with the tagsets size of the corpus. On morphologically fine tagsets, even the LSTM tagger fails to reach results reported in the literature when reproducing those setups.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group “User-Centred Social Media”.

References


<table>
<thead>
<tr>
<th>Corpus Id</th>
<th># Tags</th>
<th>Acc (%)</th>
<th>HunPos CRF Word-Char+</th>
</tr>
</thead>
<tbody>
<tr>
<td>Czech</td>
<td>1,574</td>
<td>95.9</td>
<td>-4.7 -3.2 -1.5</td>
</tr>
<tr>
<td>German-2</td>
<td>54</td>
<td>97.6</td>
<td>-0.1 -0.2 0.9</td>
</tr>
<tr>
<td>Swedish-2</td>
<td>153</td>
<td>96.1</td>
<td>0.0 -0.6 0.1</td>
</tr>
</tbody>
</table>

Table 4: Results of reproducing setups in the literature using the full corpus size


Tobias Horsmann and Torsten Zesch. 2016. LTL-UDE @ EmpiriST 2015: Tokenization and PoS Tagging of Social Media Text. In Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task, pages 120–126, Berlin, Germany.


Simon Krek, Tomaž Erjavec, Kaja Dobrovolec, Sara Može, Nína Ledinek, and Nanika Holz. 2013. Training corpus sj500k 1.3. Slovenian language resource repository CLARIN-SI.


