On the Choice of Auxiliary Languages for Improved Sequence Tagging

Lukas Lange[1,2,3] **Heike Adel**[1] **Jannik Strötgen**[1]
[1] Bosch Center for Artificial Intelligence, Renningen, Germany
[2] Spoken Language Systems (LSV), Saarland University, Saarbrücken, Germany
[3] Saarbrücken Graduate School of Computer Science, Saarbrücken, Germany
{Lukas.Lange,Heike.Adel,Jannik.Stroetgen}@de.bosch.com

Abstract

Recent work showed that embeddings from related languages can improve the performance of sequence tagging, even for monolingual models. In this analysis paper, we investigate whether the best auxiliary language can be predicted based on language distances and show that the most related language is not always the best auxiliary language. Further, we show that attention-based meta-embeddings can effectively combine pre-trained embeddings from different languages for sequence tagging and set new state-of-the-art results for part-of-speech tagging in five languages.

1 Introduction

State-of-the-art methods for sequence tagging tasks, such as named entity recognition (NER) and part-of-speech (POS) tagging, exploit embeddings as input representation. Recent work suggested to include embeddings trained on related languages as auxiliary embeddings to improve model performance: Catalan and Portuguese embeddings, for instance, help NER models on Spanish-English code-switching data (Winata et al., 2019a). In this paper, we analyze whether auxiliary embeddings should be chosen from related languages, or if embeddings from more distant languages could also help.

For this, we revisit current language distance measures (Gamallo et al., 2017) and adapt them to the embeddings and training data used in our experiments. We investigate the question, whether we can predict the best auxiliary language based on those language distance measures. Our results suggest that no strong correlation exists between language distance and performance and that even less related languages can be a good choice as auxiliary languages.

In our experiments, we explore both available monolingual and multilingual pre-trained byte-pair encoding (Heinzerling and Strube, 2018) and FLAIR embeddings (Akbik et al., 2018). For combining monolingual subword embeddings from different languages, we investigate two different methods: the concatenation of embeddings and the use of attention-based meta-embeddings (Kiela et al., 2018; Winata et al., 2019a).

We perform experiments on CoNLL and universal dependency datasets for NER and POS tagging in five languages and show that meta-embeddings are a promising alternative to the concatenation of additional auxiliary embeddings as they learn to decide on the auxiliary languages in an unsupervised way. Moreover, the inclusion of more languages is often beneficial and meta-embeddings can be effectively used to leverage a larger number of embeddings and achieve new state-of-the-art performance on all five POS tagging tasks. Finally, we propose guidelines to decide which auxiliary languages one should use in which setting.

2 Related Work

Combination of Embeddings. Previous work has seen performance gains by combining, e.g., various types of word embeddings (Tsuboi, 2014) or variants of the same type of embeddings trained on different corpora (Luo et al., 2014). For the combination, alternative solutions have been proposed, such as different input channels (Zhang et al., 2016), concatenation (Yin and Schütze, 2016), averaging of embeddings (Coates and Bollegala, 2018), and attention (Kiela et al., 2018). In this paper, we compare the inclusion of auxiliary languages via concatenation to the dynamic combination with attention.

Auxiliary Languages. Winata et al. (2019a) proposed to include embeddings from closely-related languages to improve NER performance
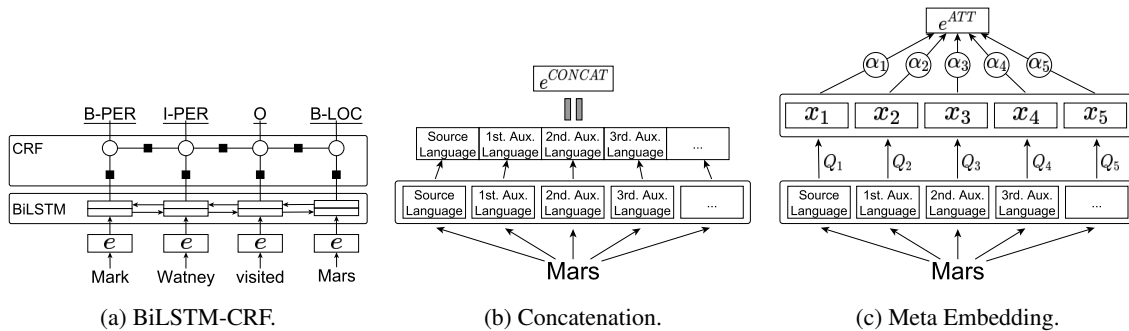
Figure 1: Overview of our model architecture (left). The embedding combination e can be either computed using the concatenation e^{CONCAT} (middle) or the meta embedding method e^{ATT} (right).

in code-switching settings, i.e., it was shown that Catalan and Portuguese embeddings help for Spanish-English NER. In a later work, it was shown that also more distant languages can be beneficial (Winata et al., 2019b), but only tests in the special setting of code-switching NER were performed and no connection between the relatedness of languages and the performance increase was analyzed. In contrast, our work shows that the inclusion of auxiliary languages increases performance in monolingual settings as well and we analyze whether language distance measures can be used to select the best auxiliary language in advance.

Language Distance Measures. Gamallo et al. (2017) proposed to measure distances between languages by using the perplexity of language models trained on one language and applied to another language. Campos et al. (2019) used a similar method to retrace changes in multilingual diachronic corpora over time. Another popular measure of similarity is based on vocabulary overlap, assuming that similar languages share a large portion of their vocabulary (Brown et al., 2008).

3 Sequence Tagging Model

Following Lample et al. (2016), we use a bidirectional long short-term memory network (BiLSTM) (Hochreiter and Schmidhuber, 1997) followed by a conditional random field (CRF) classifier (Lafferty et al., 2001) (see Figure 1a).

3.1 Embeddings

Each input word is represented with a pretrained word vector. We experiment with byte-pair encoding (BPEmb) (Heinzerling and Strube, 2018) and FLAIR embeddings (Akbik et al., 2018), as

for both of them pretrained embeddings are publicly available for all the languages we consider.[1]

3.2 Combination of Embeddings

As we experiment with multiple word embeddings, we compare two combination methods: a simple **concatenation** e^{CONCAT} and attention-based **meta-embeddings** e^{ATT} as shown in Figure 1b and 1c, respectively, and described next.

In both cases, the input are n embeddings $e_i, 1 \leq i \leq n$ that should be combined. In our experiments, we use embeddings from n different languages.

For concatenation, we simply stack the individual embeddings into a single vector: $e^{CONCAT} = [e_1, e_2, .., e_n]$.

In the case of meta-embeddings, we follow Kiela et al. (2018) and compute the combination as a weighted sum of embeddings. For this, all n embeddings e_i need to be mapped to the same size first. In contrast to previous work, we use a nonlinear mapping as this yielded better performance in our experiments. Thus, we compute $x_i = tanh(Q_i \cdot e_i + b_i)$ with weight matrix Q_i, bias b_i and $x_i \in \mathbb{R}^E$ being the i-th embedding e_i mapped to the size E of the largest embedding. The attention weight α_i for each embedding x_i is then computed with a fully-connected hidden layer of size H with parameters $W \in \mathbb{R}^{H \times E}$ and $V \in \mathbb{R}^{1 \times H}$, followed by a softmax layer.

$$\alpha_i = \frac{\exp(V \cdot \tanh(W x_i))}{\sum_{l=1}^{n} \exp(V \cdot \tanh(W x_l))}$$

The parameters of the meta-embedding layer $(Q_1, ..., Q_n, b_1, ..., b_n, W, V)$ are randomly initialized and learnt during training.

---

[1] https://github.com/flairNLP/flair
https://nlp.h-its.org/bpemb/

NER	De	En	Es	Fi	Nl
Monolingual	79.78 ± .49	86.78 ± .15	78.99 ± .91	78.00 ± .87	78.91 ± .42
Multilingual	75.37 ± .87	86.52 ± .34	78.33 ± .47	77.41 ± .86	77.49 ± .45
Mono + Multi	81.13 ± .46	**88.01 ± .27**	80.32 ± .50	81.44 ± .36	81.15 ± .43
Mono + DE	-	87.46 ± .19	79.79 ± .74	80.31 ± .21	81.31 ± .15
Mono + EN	80.92 ± .29	-	80.48 ± .56	81.22 ± .26	80.84 ± .30
Mono + ES	80.29 ± .20	87.37 ± .30	-	80.80 ± .83	80.62 ± .39
Mono + FI	81.10 ± .36	87.94 ± .17	79.91 ± .82	-	80.65 ± .48
Mono + NL	81.25 ± .14	87.38 ± .22	80.93 ± .25	80.67 ± .49	-
Mono + All	81.52 ± .33	87.70 ± .06	80.63 ± .34	82.07 ± .33 [†]	81.73 ± .26 [†]
Meta-Embeddings	**81.75 ± .50** [†]	87.87 ± .23	80.84 ± .52	**83.12 ± .12** [†]	**82.13 ± .50** [†]

POS	De	En	Es	Fi	Nl
Monolingual	93.02 ± .11	94.17 ± .09	96.23 ± .04	92.84 ± .13	94.01 ± .21
Multilingual	92.19 ± .20	94.10 ± .06	96.01 ± .07	91.95 ± .11	93.35 ± .22
Mono + Multi	93.40 ± .08	95.11 ± .07	**96.54 ± .03**	94.70 ± .12	94.94 ± .13
Mono + DE	-	95.11 ± .09	96.43 ± .13	94.43 ± .18	94.70 ± .09
Mono + EN	93.26 ± .11	-	96.52 ± .06	94.45 ± .14	94.80 ± .12
Mono + ES	93.31 ± .13	95.03 ± .09	-	94.48 ± .14	94.79 ± .17
Mono + FI	93.41 ± .12	94.97 ± .04	96.34 ± .08	-	94.92 ± .13
Mono + NL	93.52 ± .10	94.99 ± .08	96.41 ± .07	94.42 ± .08	-
Mono + All	**93.60 ± .14** [†]	**95.40 ± .04** [†]	96.46 ± .09	**95.61 ± .08** [†]	95.31 ± .08
Meta-Embeddings	93.51 ± .08	95.36 ± .10 [†]	96.48 ± .06	**95.61 ± .11** [†]	**95.34 ± .14** [†]

Table 1: Results of NER (F_1, above) and POS (Accuracy, below) experiments with BPE embeddings. [†] marks models that are statistically significant to the best Mono + X model. box highlights the closest auxiliary language according to language distance measure d_{MV}, and box the best auxiliary language according to performance.

		De	En	Es	Fi	Nl
NER	Straková et al. (2019)	**85.1**	**93.3**	**88.8**	-	**92.7**
	Meta-Emb. (BPEmb)	81.8	87.9	80.8	83.1	82.1
	Meta-Emb. (FLAIR)	83.9	90.7	86.2	**85.1**	86.6
POS	Yasunaga et al. (2018)	94.4	95.8	96.8	95.4	93.1
	Meta-Emb. (BPEmb)	93.5	95.4	96.5	95.6	95.3
	Meta-Emb. (FLAIR)	**94.8**	**96.5**	**97.2**	**97.8**	**96.8**

Table 2: Comparison with state of the art.

Finally, the embeddings x_i are weighted using the attention weights α_i resulting in the word representation $e^{ATT} = \sum_i \alpha_i \cdot x_i$

4 Experiments and Results

We perform NER and POS experiments on five languages: German (De), English (En), Spanish (Es), Finnish (Fi), and Dutch (Nl). Note that we assume at least a character overlap to use auxiliary embeddings from another language. Thus, languages with a different character set, e.g., Asian languages, cannot be used, in this setting. Future work could investigate the inclusion of languages with different character sets, e.g., by using bilingual dictionaries or machine translation.

For NER, we use the CoNLL 2002/03 datasets (Tjong Kim Sang, 2002; Tjong Kim Sang and De Meulder, 2003) and the FiNER corpus (Ruokolainen et al., 2019). For POS tagging, we experiment with the universal dependencies treebanks.[2] For each language, we report results for the following methods:

Monolingual (Mono). Only embeddings from the source language were taken for the experiments. This is the baseline setting. We also compare our results to **multilingual embeddings (Multi)** which have been successfully used in monolingual settings as well (Heinzerling and Strube, 2019). To ensure comparability, we use the multilingual versions of BPEmb and FLAIR, which were trained simultaneously on 275 and 300 languages, respectively.

Mono + X. A second set of embeddings from a different language X is concatenated with the original monolingual embeddings. While for this typically embeddings from a related language are chosen, we report results for all language combinations and investigate in particular whether relatedness is necessary for improvement.

---

[2] We predict the UPOS tag from the following UD v2.0 treebanks: de_gsd, en_ewt, es_gsd, fi_tdt, nl_alpino.

Mono + All & Meta-Embeddings. We also experiment with the combination of all embeddings from all languages from our experiments. In this setting, we use all six embeddings (five monolingual embeddings and the multilingual embeddings) and combine them either using concatenation (Mono + All) or meta-embeddings.

We have chosen these settings that are mainly based on monolingual embeddings, as the current state-of-the-art for named entity recognition is based on monolingual FLAIR embeddings (Akbik et al., 2019). In addition, multilingual embeddings, such as multilingual BERT (Devlin et al., 2019) tend to perform worse than their monolingual counterparts[3] in monolingual experiments. For completeness, we include experiments with multilingual embeddings as mentioned before.

4.1 Results

Following Reimers and Gurevych (2017), we report all experimental results as the mean of five runs and their standard deviation in Table 1 for experiments with byte-pair encoding embeddings. The results with FLAIR embeddings can be found in the appendix. We performed statistical significance testing to check if the concatenation (Mono + All) and meta-embeddings models are better than the best Mono + X model. We used paired permutation testing with 2^{20} permutations and a significance level of 0.05 and performed the Fischer correction following Dror et al. (2017).[4]

For meta-embeddings, we found statistically significant differences in 12 out of 20 settings (6 with BPEmb, 6 with FLAIR) against the best monolingual + X model, while we found statistically significant differences for Mono + All in only 7 out of 20 cases. This suggests that meta-embeddings are superior to monolingual settings with one auxiliary language as well as to the concatenation of all embeddings. Further, we found that the combination of monolingual and multilingual byte-pair encoding embeddings is always superior to either monolingual or multilingual embeddings alone for both tasks. Even though the multilingual embeddings have seen many languages during pre-training, they can still benefit from the high performance of monolingual embeddings and vice versa. As the meta-embeddings

---

Rank	d_{MV}				
	De	En	Es	Fi	Nl
1	Nl	Nl	En	En	De*
2	En	Fi	Nl	De	En*
3	Fi	De	Fi	Nl	Fi
4	Es	Es	Es	Es	Es

Table 3: Language ranking according to the majority voting distance d_{MV}. * highlights equal ranks.

assign attention weights for each embedding, we can inspect the importance the models give to the different embeddings. An analysis for an example sentence can be found in Section D in the appendix. Table 2 provides the results of BPEmb and FLAIR meta-embeddings in comparison to state of the art, showing that we set the new state of the art for POS tagging.

4.2 Analysis of Language Distances

To evaluate how useful language distances are for predicting the best auxiliary language, we compare rankings based on language distances and the observed performance rankings based on Table 1. For this, we take the language distance from Gamallo et al. (2017), which is based on language modeling perplexity PP of unigram language models LM applied to texts of foreign languages CH. Lower language model perplexities on a foreign dataset indicate higher language relatedness.

$$d_P(L1, L2) = \text{PP}(\text{CH}_{L2}, \text{LM}_{L1})$$

We also test language similarities based on vocabulary overlap with $W(L1|L2)$ being the number of words of $L1$ which are shared with $L2$ and $N(L1)$ the number of words of $L1$ shared with other languages.

$$d_V(L1, L2) = \frac{W(L1|L2) + W(L2|L1)}{2 \cdot \min(N(L1), N(L2))}$$

For our experiments, we further adapt d_P to use the perplexity of the FLAIR forward language models on the test data provided by Gamallo et al. (2017) and call it d_P^*. Similarly, we adapt d_V^* to compute the overlap of words in our training data. Note that both variants, d_P^* and d_V^*, are based on properties from either our model or training data and are, therefore, specific to our setting. Finally, we create a ranking d_{MV} which combines the rankings from d_P, d_P^*, d_V, d_V^* with majority voting. The ranking of d_{MV} is provided in Table 3, the rankings of the individual distance measures are given in the appendix.
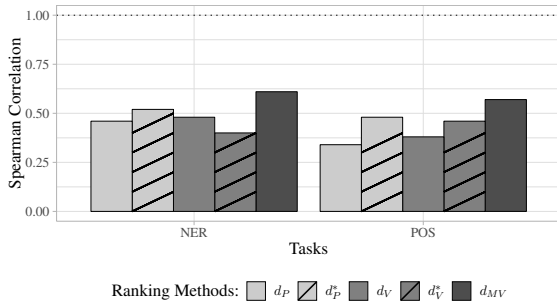
Figure 2: Spearman's rank correlation between language distance and model performance rankings for NER and POS tasks for different language distances.

To analyze the correlation between language distance measures and the performance of our model, we compute the Spearman's rank correlation coefficient between the real rankings based on performance and predicted rankings from our language distances. The results are shown in Figure 2. We conclude that predicting the auxiliary language ranking is a hard task and see that the most related language is not always the best auxiliary language in practice (cf., Table 1). This holds in particular for POS tagging, where the performance differences of models are quite small.

In general, d_P^* shows a higher correlation with model performance than d_P and d_V, indicating that not only word overlap plays a role but also context information. The majority voting d_{MV} achieves the highest correlation and often predicts the best auxiliary language for NER models using byte-pair encoding. However, the actual ranking of all languages does not match the performance ranking, which results in a relatively low correlation with only a little above 0.5.
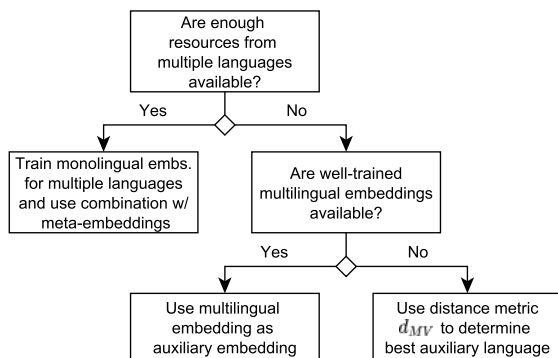
4.3 Practical Guide

Figure 3: Proposal for auxiliary embedding selection.

Finally, we propose a small guide in Figure 3 to decide which auxiliary languages one can use to improve performance over monolingual embeddings. Depending on the available amount of data, it is recommended to train multiple monolingual embeddings and combine them using meta-embeddings, which was observed to be the best method in our experiments. If not enough data is available to train monolingual embeddings, the best solution would be the inclusion of multilingual embeddings, assuming the existence of high-quality embeddings, such as multilingual byte-pair encoding. If none of the above applies, language distance measures, in particular the combination of multiple distances, can help to identify the most promising auxiliary embeddings. Despite not always predicting the best model, the predicted auxiliary language often led to improvements over the monolingual baseline in our experiments.

5 Conclusion

In this paper, we investigated the benefits of auxiliary languages for sequence tagging. We showed that it is hard to predict the best auxiliary language based on language distances. We further showed that meta-embeddings can leverage multiple embeddings effectively for those tasks and set the new state of the art on part-of-speech tagging across different languages. Finally, we proposed a guide on how to decide which method of including auxiliary languages one should use.

Acknowledgments

References

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 724–728, Minneapolis, Minnesota. Association for Computational Linguistics.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International*

Conference on Computational Linguistics, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Cecil H Brown, Eric W Holman, Søren Wichmann, and Viveka Velupillai. 2008. Automated classification of the world s languages: a description of the method and preliminary results. *STUF-Language Typology and Universals Sprachtypologie und Universalienforschung*, 61(4):285–308.

Jos Ramom Pichel Campos, Pablo Gamallo Otero, and Iaki Alegria Loinaz. 2019. Measuring diachronic language distance using perplexity: Application to english, portuguese, and spanish. *Natural Language Engineering*, page 122.

Joshua Coates and Danushka Bollegala. 2018. Frustratingly easy meta-embedding – computing meta-embeddings by averaging source word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 194–198, New Orleans, Louisiana. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rotem Dror, Gili Baumer, Marina Bogomolov, and Roi Reichart. 2017. Replicability analysis for natural language processing: Testing significance with multiple datasets. *Transactions of the Association for Computational Linguistics*, 5:471–486.

Pablo Gamallo, José Ramom Pichel, and Iñaki Alegria. 2017. From language identification to language distance. *Physica A: Statistical Mechanics and its Applications*, 484:152–162.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Benjamin Heinzerling and Michael Strube. 2019. Sequence tagging with contextual and non-contextual subword representations: A multilingual evaluation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 273–291, Florence, Italy. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computing*, 9(8):1735–1780.

Douwe Kiela, Changhan Wang, and Kyunghyun Cho. 2018. Dynamic meta-embeddings for improved sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1477, Brussels, Belgium. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Yong Luo, Jian Tang, Jun Yan, Chao Xu, and Zheng Chen. 2014. Pre-trained multi-view word embedding using two-side neural network. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 1982–1988. AAAI Press.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark. Association for Computational Linguistics.

Teemu Ruokolainen, Pekka Kauppinen, Miikka Silfverberg, and Krister Lindén. 2019. A finnish news corpus for named entity recognition. *Language Resources and Evaluation*, pages 1–26.

Jana Straková, Milan Straka, and Jan Hajič. 2019. Neural Architectures for Nested NER through Linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331, Florence, Italy. Association for Computational Linguistics.

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In

Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pages 142–147.

Yuta Tsuboi. 2014. Neural networks leverage corpus-wide information for part-of-speech tagging. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 938–950, Doha, Qatar. Association for Computational Linguistics.

Genta Indra Winata, Zhaojiang Lin, and Pascale Fung. 2019a. Learning multilingual meta-embeddings for code-switching named entity recognition. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 181–186, Florence, Italy. Association for Computational Linguistics.

Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. 2019b. Hierarchical meta-embeddings for code-switching named entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3541–3547, Hong Kong, China. Association for Computational Linguistics.

Michihiro Yasunaga, Jungo Kasai, and Dragomir Radev. 2018. Robust multilingual part-of-speech tagging via adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 976–986, New Orleans, Louisiana. Association for Computational Linguistics.

Wenpeng Yin and Hinrich Schütze. 2016. Learning word meta-embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1351–1360, Berlin, Germany. Association for Computational Linguistics.

Ye Zhang, Stephen Roller, and Byron C. Wallace. 2016. MGNC-CNN: A simple approach to exploiting multiple word embeddings for sentence classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1522–1527, San Diego, California. Association for Computational Linguistics.

A Hyperparameters and Training

We use the Byte-Pair-Encoding embeddings (Heinzerling and Strube, 2018) with 300 dimensions and a vocabulary size of 200k tokens for all languages. For FLAIR, we use the embeddings provided by the FLAIR framework (Akbik et al., 2018) with 2048 dimensions for each language model resulting in a total embedding size of 4096 dimensions. The bidirectional LSTM has a hidden size of 256 units. For training, we use stochastic gradient descent with a learning rate of 0.1 and a batch size of 64 sentences. The learning rate is halved after 3 consecutive epochs without improvement on the development set. We apply dropout with probability 0.1 after the input layer.

B Language Distances

We report the language rankings of the single metrics d_P, d_P^*, d_V and d_V^* in Table 4.

C Results on NER and POS tagging with FLAIR embeddings

We performed the same experiments as in Section 4.1 with FLAIR embeddings as well and report the results in Table 5 for NER and for POS tagging.

In difference to the BPE experiments reported in the paper, we do not include multilingual embeddings in the Mono + All and meta-embedding versions of FLAIR. The reason is prior experiments in which multilingual embeddings led to reduced performance. This is also reflected in the poor performance of the multilingual FLAIR embeddings alone. It seems that the multilingual BPE embeddings are more effective in downstream tasks than the multilingual FLAIR embeddings.

D Analysis of Attention Weights

As the meta-embeddings assign attention weights for each embedding, we can inspect the importance the models give to the different embeddings. Figure 4 shows the assigned weights for an English sentence. In general, the model assigns most weight to the English embeddings. However, we observe an increased weight for German and the multilingual embedding for the German word *Bayerische*. Even though *Vereinsbank* is also a German word, the model focuses more on English for this word, probably because the subword *bank* has the same meaning in English.

E Study: Increased Number of Parameters vs. Auxiliary Language

To investigate whether the performance increase comes from the increased number of parameters rather than the inclusion of more embeddings, we also investigated including the same embedding type twice (Mono + Mono). However, we found

Rank	de				en				es				fi				nl			
	d_P	d_P^*	d_V	d_V^*	d_P	d_P^*	d_V	d_V^*	d_P	d_P^*	d_V	d_V^*	d_P	d_P^*	d_V	d_V^*	d_P	d_P^*	d_V	d_V^*
1	nl	nl	en	nl	nl	nl	de	fi	en	nl	en	en	de	nl	en	en	de	de	en	en
2	en	en	nl	en	es	fi	nl	nl	nl	en	de	nl	nl	de	de	nl	en	en	de	de
3	fi	fi	es*	fi	de	de	fi	es	fi	de	fi	fi	en	en	es*	de	fi	fi	es*	es
4	es	es	fi*	es	fi	es	es	de	de	fi	nl	de	es	es	nl*	es	es	es	fi*	fi

Table 4: Language distances. Languages marked with * are ranked the same.

NER	De	En	Es	Fi	Nl
Straková et al. (2019)	85.10	93.28	88.81	-	92.69
Monolingual	82.66 ± .11	89.98 ± .11	85.08 ± .68	83.38 ± .31	85.68 ± .27
Multilingual	66.21 ± .79	82.87 ± .24	77.87 ± .93	73.95 ± .74	77.44 ± .52
Mono + Mono	82.45 ± .45	89.95 ± 0.21	85.26 ± .06	83.37 ± .48	85.67 ± .06
Mono + Multi	82.95 ± .21	90.04 ± .11	84.70 ± .50	83.46 ± .37	86.04 ± .28
Mono + DE	-	90.24 ± .19	85.16 ± .23	84.23 ± .22	85.82 ± .22
Mono + EN	83.27 ± .36	-	85.53 ± .20	84.10 ± .26	86.73 ± .09
Mono + ES	82.85 ± .34	90.14 ± .13	-	83.88 ± .31	86.16 ± .09
Mono + FI	83.10 ± .45	90.14 ± .09	85.06 ± .64	-	86.14 ± .31
Mono + NL	82.79 ± .24	90.18 ± .15	85.77 ± .27	83.65 ± .31	-
Mono + All	83.43 ± .29	90.29 ± .18	85.48 ± .78	84.32 ± .32	86.43 ± .33
Meta-Embeddings	83.90 ± .14 [†]	90.70 ± .29 [†]	86.18 ± .35	85.09 ± .30 [†]	86.58 ± .58
POS	**De**	**En**	**Es**	**Fi**	**Nl**
Yasunaga et al. (2018)	94.35	95.82	96.84	95.40	93.09
Monolingual	94.72 ± .07	96.28 ± .05	97.08 ± .03	97.52 ± .03	96.48 ± .11
Multilingual	92.82 ± .20	93.69 ± .07	96.06 ± .13	92.98 ± .10	94.85 ± .11
Mono + Mono	94.74 ± .15	96.24 ± .02	97.04 ± .08	97.55 ± .05	96.45 ± .13
Mono + Multi	94.72 ± .13	96.29 ± .04	97.04 ± .05	97.52 ± .05	96.77 ± .02
Mono + DE	-	96.41 ± .07	97.11 ± .08	97.64 ± .04	96.62 ± .06
Mono + EN	94.71 ± .04	-	97.13 ± .12	97.52 ± .06	96.49 ± .09
Mono + ES	94.67 ± .06	96.36 ± .03	-	97.48 ± .03	96.61 ± .13
Mono + FI	94.65 ± .05	96.38 ± .03	97.14 ± .05	-	96.68 ± .05
Mono + NL	94.64 ± .03	96.31 ± .07	97.06 ± .04	97.51 ± .04	-
Mono + All	94.64 ± .10	96.48 ± .05	97.11 ± .04	97.52 ± .06	96.54 ± .20
Meta-Embeddings	94.78 ± .09	96.49 ± .03 [†]	97.18 ± .07	97.82 ± .03 [†]	96.83 ± .13 [†]

Table 5: Results of NER (F_1, above) and POS (Accuracy, below) experiments with FLAIR embeddings. [†] marks models that are statistically significant to the best Mono + X model. box highlights the closest auxiliary language according to language distance measure d_{MV}, and box the best auxiliary language according to performance.
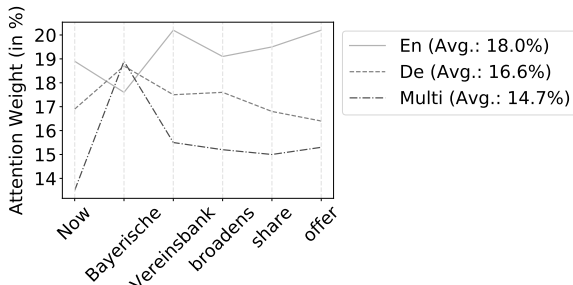

Figure 4: Learned attention weights of the meta-embeddings model with byte-pair encoding embeddings for English NER.

that this does not help: The performance is comparable to the monolingual baseline. Thus, the performance increase for Mono + X really comes from additional information provided by the embeddings of the auxiliary language.