Cross-lingual Transfer Learning for Japanese Named Entity Recognition

Andrew Johnson
Saarland University
Saarbrücken, Germany
ajohnson@coli.uni-saarland.de

Penny Karanasou
Amazon
Cambridge, UK
pkarana@amazon.co.uk

Judith Gaspers
Amazon
Aachen, Germany
gaspers@amazon.de

Dietrich Klakow
Spoken Language Systems
Saarland University
Saarbrücken, Germany
dietrich.klakow@lsv.uni-saarland.de

Abstract

This work explores cross-lingual transfer learning (TL) for named entity recognition, focusing on bootstrapping Japanese from English. A deep neural network model is adopted and the best combination of weights to transfer is extensively investigated. Moreover, a novel approach is presented that overcomes linguistic differences between this language pair by romanizing a portion of the Japanese input. Experiments are conducted on external datasets, as well as internal large-scale real-world ones. Gains with TL are achieved for all evaluated cases. Finally, the influence on TL of the target dataset size and of the target tagset distribution is further investigated.

1 Introduction

Due to the growing interest in voice-controlled devices, such as Amazon Alexa-enabled devices or Google Home, porting these devices to new languages quickly and cheaply has become an important goal. One of the main components of such a device is a model for Named Entity Recognition (NER). Typically, NER models are trained on large amounts of annotated training data. However, collecting and annotating the required data to bootstrap a large-scale NER model for an industry application with reasonable performance is time-consuming, costly, and it doesn't scale to a growing number of new languages.

Aiming to reduce the time and costs needed for bootstrapping an NER model for a new language, we leverage existing resources. In particular, we explore *cross-lingual transfer learning*, in which weights from a trained model in the source language are transferred to a model in the target language. Transfer learning (TL) has been shown previously to improve performance for target models (Yang et al., 2017; Lee et al., 2017; Riedl and Padó, 2018). However, work related to cross-lingual transfer learning for NER has mainly focused on rather similar languages, e.g. transferring from English to German or Spanish. In contrast, we focus on transferring between dissimilar languages, i.e. from English to Japanese.

We present experimental results on external, i.e. publicly available, corpora, as well as on internally gathered large-scale real-world datasets. First, a deep neural network model is developed for NER, and we extensively explore which combinations of weights are most useful for transferring information from English to Japanese. Furthermore, aiming to overcome the linguistic and orthographic dissimilarity between English and Japanese, we propose to romanize the Japanese input, i.e. convert the Japanese text into the Latin alphabet. This results in a common character embedding space between the two languages, and intuitively should allow for more efficient transfer learning at the character level.

Gains with TL are achieved on all evaluated target datasets, even large-scale industrial ones. Moreover, the effect of TL on the target dataset size and of the target tagset distribution is investigated. Finally, we show that similar gains are achieved when applying the proposed approach from English to German, indicating the possibility to generalize it both to European and non-European target languages.

The author Andrew Johnson conducted the work for this paper during an internship at Amazon, Aachen, Germany.

Proceedings of NAACL-HLT 2019, pages 182–189
Minneapolis, Minnesota, June 2 - June 7, 2019.

2 NER model

The growth in neural approaches spurred a push towards "NLP from scratch", that is, without engineering task- or language-specific features by hand (Collobert et al., 2011). Currently, mainly recurrent and/or convolutional neural networks are applied. In Chiu and Nichols (2015), the authors combined a Bi-LSTM to learn long-distance relationships with a CNN to generate character-level representations. A Bi-LSTM-CNN-CRF showed state-of-the-art performance on NER (Ma and Hovy, 2016). CNNs have been shown to be less useful for languages like Japanese, in which average NEs are quite short at around two characters on average (Misawa et al., 2017). Bi-LSTM-CRF models without any CNN layer have also performed well on NER (Huang et al., 2015; Lample et al., 2016). Using this architecture with a novel type of embeddings termed "contextual string embeddings" has recently led to state-of-the-art results (Akbik et al., 2018).

For our baseline NER system we use a Bi-LSTM architecture that takes word and character embeddings as input. The same architecture is used both for the source and the target languages to allow for transfer of weights when the cross-lingual TL is applied. This architecture largely resembles the model in Lample et al. (2016), except for the final CRF layer. For every token, word and character embeddings are generated. The latter are passed through a character Bi-LSTM, the output of which is concatenated with the word embeddings. This combined representation is then passed into the word Bi-LSTM, followed by a dense layer and a final softmax layer. An example for English is presented in Figure 1. Note that the character level inputs in this figure are unigrams, but in practice we use bigrams, i.e. "Ye" and "es" for "Yes".

Although including a CRF as the final layer tends to raise scores overall (Reimers and Gurevych, 2017; Huang et al., 2015), others have demonstrated that transferring CRF weights does not contribute to meaningful gain in the context of TL (Lee et al., 2017). In this work, a CRF layer is not included in the baseline. In another recent work, monolingual Japanese models have used "character-based models", with labels assigned to each individual character (Misawa et al., 2017). We do not employ this approach since our source model in English is not character-based.

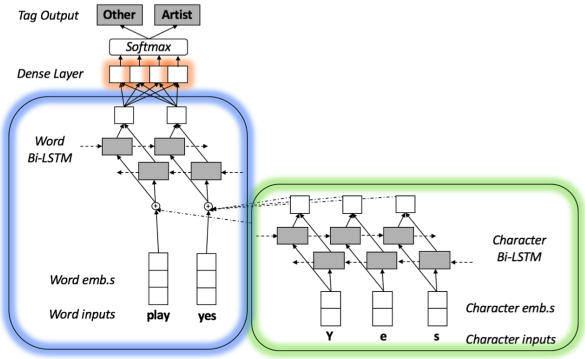

Figure 1: NER model: an English example

3 Transfer Learning

Cross-lingual TL is applied to transfer knowledge from the source to the target language. Working with neural network-based models, this is achieved by initializing some layers of the target network using the weights of the source network, which is assumed to be already trained using a (large) available annotated training corpus.

3.1 Related work

One of the first works on cross-lingual TL for NER that did not rely on parallel corpora used a CRF and included hand-crafted features (Zirikly and Hagiwara, 2015). Currently, most work on TL is done with neural models. Because neural models often consist of multiple layers, one important design decision is which layers to transfer from source to target. Much related work involves only transferring a single layer or specific combination of layers. In Lee et al. (2017) the authors present more thorough results combining lower and higher layers, without transferring intermediate layers though. In Yang et al. (2017) it is suggested to transfer only the character embeddings and the character RNN weights between languages. The reason for this is likely that many languages written in the Latin alphabet have a large charset overlap, but far less vocabulary overlap.

Another question of interest concerns the pair of languages between which TL can be achieved. Past work has shown transferring to a related language to help more than to an unrelated one for NER, POS tagging, and NMT (Zirikly and Hagiwara, 2015; Kim et al., 2017; Dabre et al., 2017). In Yang et al. (2017) it is mentioned that without additional resources, it is "very difficult for transfer learning between languages with disparate alphabets". This background suggests TL from En-

glish to Japanese to be non-trivial.

Finally, another consideration with TL is the size of the target dataset. For one NER task, TL gains were shown to decrease to nearly zero as the size of the target training data increased to around 50k tokens (Lee et al., 2017). Similarly, for domain adaptation, a "phase transition" was observed in the amount of used target data, such that using source data was not effective when the target model was trained on 3.13k or more target instances (Ben-David et al., 2010).

3.2 Specificities of Japanese language

Transferring between English and Japanese is more challenging and less explored than transferring between languages with the same alphabet. Japanese is written using an unsegmented mixture of two syllabaries as well as thousands of Chinese characters, which encode semantic information.

A process that we explore in this work to overcome the orthographic dissimilarity is the "romanization" of Japanese text, i.e. the process of transcribing Japanese text into the Latin alphabet. However, when applying romanization we lose the disambiguating effect that characters have. In fact, due to its small phonemic inventory, Japanese contains many homophones. Consider the homophone pairs in Table 1, actual examples taken from our external Japanese dataset. In their original written forms, there is no ambiguity, as there are different characters representing each meaning. This information, which is crucial here to determining which is the NE, is lost after romanization. Empirical results for sentiment classification have confirmed that romanizing Japanese text hurts performance for a monolingual model (Zhang and LeCun, 2017).

押収	欧州	加盟	亀井
oushuu	*oushuu*	*kamei*	*kamei*
to seize	Europe	to join	Kamei [surname]

Table 1: Japanese Homophones

3.3 Proposed model

Since we explore transferring weights from a source network, an important design decision is which layers to transfer. Addressing this question, we evaluate different combinations of layers to find the best one for our task. We group our weights together as shown in Figure 1 (grouped layers in boxes): character embeddings and character Bi-LSTM weights form the "character weights", word embeddings and word Bi-LSTM weights form the "word weights", and dense layer weights form the "dense weights". All possible combinations of these three groups are explored. To account for the incomplete overlap when transferring embeddings, we only update the vectors that correspond to char n-grams or words observed in both the source and target training data. This is a limitation that could be overcome if multi-lingual embeddings were used which we leave for future work.

For transferring to a target language with a different writing system than the source one we propose the *Mixed Orthographic Model* (MOM). Specifically, the character layer inputs are romanized while the word layer inputs are kept in their original Japanese text. This allows for transfer of character information from a source to a target language with originally different writing systems by creating a common and overlapping character embedding space. At the same time, keeping the original Japanese text in the word level allows us to keep the capacity to disambiguate homophones, which is lost via the romanizing process as explained in the previous section (Section 3.2).

Here is an example of the MOM for the utterance "play jazz":

Raw utterance	"ジャズを流して"
Word input	["ジャズ", "を", "流して"]
Character input	["jazu", "wo", "nagashite"] [1]

4 Experimental setup

In this section, the datasets as well as the details of the developed NER model are presented.

4.1 Datasets

For our experiments we make use of datasets in three languages. First, an English dataset is used to train the source NER model. Then, a target language dataset, which is smaller in size than the source dataset, is used to build a target NER model. This serves as the target baseline. The weights transferred from the source model are used to initialize this target model, which is then trained with the available target data, resulting in a new target model. As mentioned before, the focus of this paper is TL between dissimilar languages, and thus the main experiments use a Japanese dataset as the target corpus. However, for the sake

[1] Shown prior to conversion into character n-grams

Language	Dataset	Train	Test	Dev
EN	ConLL 2003	14,987	3,684	3,466
JP	BCCWJ	3,600	325	324
	CRF-KNBC	2,940	980	979

Table 2: Number of utterances per external dataset

of comparison, we also conducted some experiments using a German target dataset, thus transferring between more similar languages, i.e. both belonging to the indo-European family, and evaluating the generalization power of the adopted approach.

We evaluate our approach both on external and internal datasets. External datasets are composed of company data and are mainly used for comparing our monolingual models to the state-of-the-art, while internal datasets are composed of publicly available data and are used to explore potential data reductions in a real-world large-scale industry setting.

Segmentation and romanization of Japanese text are performed with the open source Japanese text analyzer MeCab[2].

4.1.1 External data

As external English data, we use the English CoNLL 2003 NER dataset (Tjong Kim Sang and De Meulder, 2003) which contains four named entity (NE) categories.

We make use of three external datasets for Japanese NER. The first one is the Balanced Corpus of Contemporary Written Japanese (BCCWJ) (Iwakura et al., 2016), containing a variety of writing types, such as blogs and magazine articles. In addition, we created a dataset by combining two small Japanese datasets annotated with NEs: i) a small dataset included in the CRF++ tool, and ii) the Kyoto University and NTT Blog Corpus (KNBC) with data from blogs on topics such as tourism, sports, and technology. We are referring to this dataset as "CRF-KNBC".

Most Japanese NER datasets use IREX tags. Similar to CoNLL 2003, IREX 1999 was a shared task for NER and contains eight tags, three of which are the same as in CoNLL. The remaining tags can be viewed as an expansion of ConLL's fourth category, and hence can be grouped together to have the same tagset as CoNLL. We do this to facilitate TL from English.

See Table 2 for details on the external datasets.

[2]http://taku910.github.io/mecab/

Language	Dataset	Train	Test	Dev
EN	Large	5M	200k	200k
JP	Large	1M	255k	255k
	Medium	381k	47k	47k
	Small	49.3k	6.2k	6.2k
DE	Large	1M	143.6k	143.6k
	Medium	99.4k	12.4k	12.4k

Table 3: Number of utterances per internal dataset

4.1.2 Internal data

We are mainly interested in exploring TL and the resulting potential data reduction in a large-scale industry setting with different amounts of target data being available, as target data amounts typically increase over time due to continuous data collection. Internal datasets comprise utterances which are representative of user requests to voice-controlled devices and are annotated with NEs. To explore the benefit of TL during different stages of system development, i.e. with availability of different data sources, we include different datasets in our experiments which we distinguish by their size. In particular, we shall refer to any dataset containing over one million utterances as "Large", anything with fewer than one million but more than one hundred thousand as "Medium", and anything with fewer than one hundred thousand utterances as "Small". Note the difference in scale from the external data, the largest of which would still be well below the threshold defined here as small. Following this convention, we have the internal datasets presented in Table 3. None of the smaller datasets are subsets of the larger ones; each is an entirely separate dataset. However, each dataset includes the same kind of data and largely shares the same tagset.

Another major difference from the external datasets is the size of the tagset. Internal data, including both source and target, use over two hundred distinct tags, which are not evenly distributed. In fact, Figure 2 (in log-log scale) shows a very long tail, with the most frequently observed tags belonging to a very small subset of all possible tags. This characteristic makes the internal data a challenging case.

4.2 Model setup

For optimizing our NER models we used Adam (Kingma and Ba, 2014) over cross entropy loss. To avoid overfitting, a dropout layer was used before the Word Bi-LSTM. We lowercase all word-level input. However, since capitalization is a

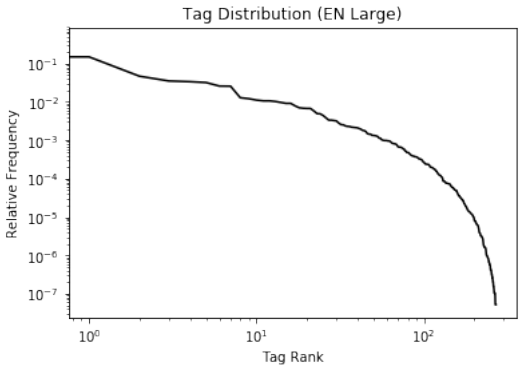

Figure 2: Tag distribution-EN internal training data in log-log scale

feature that strongly predisposes a word to be an NE, we did not lowercase the character-level input. No pre-trained word embeddings were used with internal datasets, while external datasets used Polyglot word embeddings (Al-Rfou et al., 2013). The word embedding dimensionality was 50, except where Polyglot pre-trained embeddings were used, in which case it was 64. The word LSTM size was set to 300. Character embeddings were 50-dimensional and character bigrams were used. The character LSTM was of size 100 for external datasets and 30 for internal ones. Dropout was set to 0.5. We used the evaluation script from the CoNLL shared task to compute F1 score.

During the parameter tuning phase, development set performance stabilized after 10 epochs for external models and 20 epochs for internal models. Therefore, we conduct experiments on the test set by training for these respective number of epochs. The scores reported for each model reflect the highest F1 value among all epochs.

5 Results

5.1 Layer combinations for TL

We first investigate which layer combination yields best results when being transferred. The layer groups defined in Section 3.3 are combined and experiments are conducted on the two external JP datasets as well as on a subset of the JP "Medium" internal one. The results are presented in Table 4 as absolute gains against the baseline without TL. Approximate randomization is used for each experiment (Noreen, 1989; Yeh, 2000), and all TL gains were found to be significant to $p<0.001$. The results reported are the average of running experiments five times with different random seeds. In all experiments, the system configuration detailed in Section 4.2 is followed and the MOM (see Section 3.3) is applied.

Layers Transferred	Corpora BCCWJ	CRF-KNBC	Med.-10k
No TL	65.50	50.48	81.64
Char	+0.34	-2.63	-1.62
Word	+1.50	+0.86	+1.14
Dense	-1.54	+2.86	**+3.77**
Char+Word	-0.09	+1.33	-0.39
Word+Dense	+0.63	**+4.69**	+1.95
Char+Dense	**+3.86**	+3.74	+3.72
Char+Word +Dense	+2.35	+3.92	-0.02

Table 4: Absolute gains on JP datasets by transferring different layer combinations

The best performing combination (in bold) varies between datasets. However, the "Char+Dense" combination seems to be the most reliable one, providing consistent and significant TL gains over all three evaluated corpora. This combination is different than what was previously reported in litterature (Yang et al., 2017), where it was suggested that transferring character level weights suffices. This might be because of the specific nature of our task on transferring weights between languages with dissimilar alphabets. In our case transferring word weights actually performs better than transferring character weights (compare rows "Word" and "Char"). In addition, combining the weights at word or character levels with the next dense layer weights improves further the results (rows "Word+Dense" and "Char+Dense") indicating that this dense layer still captures some language-independent information.

Due to these results, we use the "Char+Dense" combination in the following experiments.

5.2 Effect of romanization of Japanese on TL

The effect of romanization of Japanese is evaluated on one external ("BCCWJ") and a subset of an internal ("Med.-10k") JP dataset. Results are presented in Table 5 with and without romanization before and after TL, and consistent gains are shown when MOM is used with TL. In addition, there are significant gains when used without TL in the case of the internal dataset ("Med.-10k"). To the best of our knowledge, this is the first work introducing the MOM and comparing these approaches for Japanese in the context of TL. Since this model gives consistently improved results with TL, all the remaining results on Japanese data will employ this approach.

Dataset		No roman.	MOM
BCCWJ	No TL	**67.31**	65.50
	With TL	69.08	**69.36**
Med.-10k	No TL	80.65	**81.64**
	With TL	84.12	**85.36**

Table 5: Romanization of Japanese - Effect on TL

Dataset	No TL	With TL	Rel. gain
BCCWJ	65.50	**69.36**	+5.89
CRF-KNBC	50.48	**54.22**	+7.41
Small	83.15	**84.85**	+2.04
Medium	91.64	**92.20**	+0.61
Large	91.66	**92.21**	+0.60
DE Medium	87.82	**88.86**	+1.18
DE Large	89.24	**89.63**	+0.44

Table 6: Results with TL over full JP and DE datasets

5.3 TL on external and internal datasets

Applying the best configuration established previously, i.e. transfer "Char+Dense" layers and use of MOM, the results before and after TL on the full JP datasets are presented in Table 6. Gains with TL are achieved in all evaluated datasets. Moreover, with MOM and TL, we achieve state-of-the-art NER results on BCCWJ, outperforming Ichihara et al. (2015) (reported F1 score 67.68% vs. ours 69.36%). In addition, important relative gains are achieved by TL in the small external datasets, making our method particularly suited for bootstrapping a new language with very limited available annotated data. Another interesting outcome is that we still see gains in the large internal datasets (i.e. up to 1M training utterances in the internal "Large" set). This will be investigated further in the next section (Section 5.4).

Results on DE internal datasets are presented for sake of comparison and show the same trends as JP internal datasets, thus revealing the generalization of our approach for cross-lingual TL both to European and non-European target languages.

5.4 Effect of target dataset size on TL

To further investigate how the size of the target datasets influences the performance of TL, we conducted experiments on different sizes of the internal data. This was done by training on subsets of the original "Large" JP internal training set, with sizes varying from 10k to 1M utterances. Note that the source English training data is still used in full each time. The results are presented in Figure 3. As expected, larger gains are observed for smaller splits. However, TL still produces statistically significant gains for all split sizes. Note

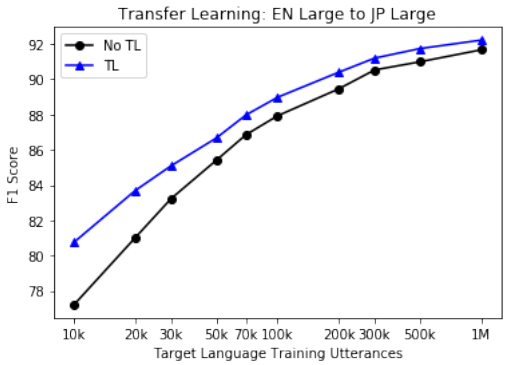

Figure 3: Applying TL on varying target training size

also that training, for example, on 500k utterances with TL is better than training on 1M utterances without TL, indicating the possibility of reducing data requirements with TL even in large-scale industrial systems.

A further analysis of the results on the internal datasets showed that the frequency of a tag class in the target training data correlated the most with TL gain. This is visualized in Figure 4 for a subset of the JP "Small" dataset. An arrow is used for each tag class with the tail of the arrow indicating the F1 score without TL and the point indicating the F1 score of that same class with TL. Thus, classes with gains point upward (blue arrows), while those that performed worse point downwards (red arrows). Classes that showed no change are indicated as circles. These mostly cluster along the bottom as classes that have an F1 score of zero before and after TL. The tags are arranged along the x axis based on their frequency in the target training dataset.

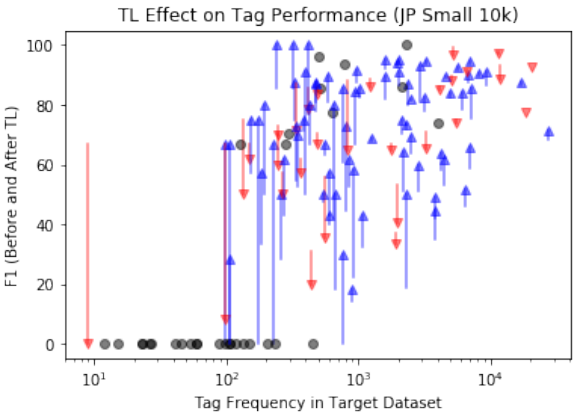

Figure 4: Gains by tag

This figure indicates that frequent tags (rightmost part of the figure) gain less by TL, probably because they already perform well. Tag classes

generally begin to show gains from TL only after they pass a certain *minimum frequency threshold* in the target dataset, which appears to be around 100. This may be the reason why we have TL gains even with large target datasets. As infrequent tag classes are observed more and more in larger splits, they begin to cross this threshold and gain from TL. Real-world data generally have long-tailed distributions, thus even very large target datasets are likely to have tag classes with few data which can benefit from TL.

6 Conclusions

A cross-lingual transfer learning approach for NER was proposed, focusing on dissimilar languages, i.e. English and Japanese. A deep neural network model was adopted and the best layer combination to transfer was extensively investigated. To overcome the orthographic dissimilarity between source and target languages a novel method, the MOM, was proposed that romanizes part of the Japanese input. Gains with TL were consistently achieved on external and large-scale real-world datasets showing that it is possible to transfer knowledge between dissimilar languages, even for large target corpora.

In the future, the proposed approach could be applied to other dissimilar language pairs, e.g. English and Chinese. Other possible extensions include using multi-lingual embeddings that could complement the currently transferred weights.

References

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.

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192. Association for Computational Linguistics.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175.

Jason PC Chiu and Eric Nichols. 2015. Named entity recognition with bidirectional lstm-cnns. *arXiv preprint arXiv:1511.08308*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Raj Dabre, Tetsuji Nakagawa, and Hideto Kazawa. 2017. An empirical study of language relatedness for transfer learning in neural machine translation. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, pages 282–286.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Masaaki Ichihara, Kanako Komiya, Tomoya Iwakura, and Maiko Yamazaki. 2015. Error analysis of named entity recognition in bccwj. *Recall*, 61:2641.

Tomoya Iwakura, Kanako Komiya, and Ryuichi Tachibana. 2016. Constructing a japanese basic named entity corpus of various genres. In *Proceedings of the Sixth Named Entity Workshop*, pages 41–46.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2832–2838.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2017. Transfer learning for named-entity recognition with neural networks. *arXiv preprint arXiv:1705.06273*.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Shotaro Misawa, Motoki Taniguchi, Yasuhide Miura, and Tomoko Ohkuma. 2017. Character-based bidirectional lstm-crf with words and characters for japanese named entity recognition. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 97–102.

Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.

Nils Reimers and Iryna Gurevych. 2017. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv: 1707.06799*.

Martin Riedl and Sebastian Padó. 2018. A named entity recognition shootout for german. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 120–125.

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-Volume 4*, pages 142–147. Association for Computational Linguistics.

Zhilin Yang, Ruslan Salakhutdinov, and William W Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. *arXiv preprint arXiv:1703.06345*.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*, pages 947–953. Association for Computational Linguistics.

Xiang Zhang and Yann LeCun. 2017. Which encoding is the best for text classification in chinese, english, japanese and korean? *arXiv preprint arXiv:1708.02657*.

Ayah Zirikly and Masato Hagiwara. 2015. Cross-lingual transfer of named entity recognizers without parallel corpora. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 390–396.