Sources of Transfer in Multilingual Named Entity Recognition

David Mueller[1,2] **Nicholas Andrews**[2] **Mark Dredze**[1,2]
[1]Center for Language and Speech Processing, Johns Hopkins University
[2]Human Language Technology Center of Excellence, Johns Hopkins University
dam,noa@jhu.edu mdredze@cs.jhu.edu

Abstract

Named-entities are inherently multilingual, and annotations in any given language may be limited. This motivates us to consider *polyglot* named-entity recognition (NER), where one model is trained using annotated data drawn from more than one language. However, a straightforward implementation of this simple idea does not always work in practice: naive training of NER models using annotated data drawn from multiple languages consistently underperforms models trained on monolingual data alone, despite having access to more training data. The starting point of this paper is a simple solution to this problem, in which polyglot models are *fine-tuned* on monolingual data to consistently and significantly outperform their monolingual counterparts. To explain this phenomena, we explore the sources of multilingual transfer in polyglot NER models and examine the weight structure of polyglot models compared to their monolingual counterparts. We find that polyglot models efficiently share many parameters across languages and that fine-tuning may utilize a large number of those parameters.

1 Introduction

Multilingual learning—using data from multiple languages to train a single model—can take many forms, such as adapting a model from a high-resource to low-resource language (Xie et al., 2018; Ni et al., 2017; Mayhew et al., 2017; Cotterell and Duh, 2017; Wu and Dredze, 2019; Màrquez et al., 2003), taking advantage of beneficial multilingual features or datasets (Kim et al., 2012; Ehrmann et al., 2011; Täckström, 2012), and unsupervised representation learning (Devlin et al., 2018a). We adopt the term "Polyglot" from Tsvetkov et al. (2016) to refer to models that are trained on and applied to multiple languages. There are several advantages to training a single polyglot model across languages. Single models ease production requirements; only one model need be maintained. They can be more efficient, using fewer parameters than multiple monolingual models. Additionally, they can enable multilingual transfer (Devlin, 2018; Wu and Dredze, 2019; Pires et al., 2019).

However, a key goal of polyglot learning concerns producing a single model that does better on each language than a monolingual model. In the context of named entity recognition, we may expect aspects of the task to transfer across languages. For example, since entity names tend to be transliterated or directly used across languages, even distant languages may see benefit from training a single model, e.g. "Apple" (company) is rendered as such in French rather than as "Pomme." Intuitively, the more similar and the larger the set of languages, the more we should expect to see a benefit from considering them jointly. These polyglot models can take advantage of different sets of labeled corpora in different languages (Gillick et al., 2016; Mulcaire et al., 2019).

Nevertheless, progress towards this goal remains mixed; polyglot models often do not improve results in each language (Mulcaire et al., 2019; Kondratyuk and Straka, 2019; Upadhyay et al., 2018; Conneau et al., 2019). Models trained across all languages come close but typically fail to outperform monolingual models. Thus, while multilingual learning can benefit low resource languages through transfer and simplify models by sharing one across all languages, it fails to realize a key goal: improving results in each language. Our experiments in §4 confirm this negative result in two different multilingual settings for 4 different neural NER models.

Our first contribution is a technique in which a polyglot NER model can be adapted to a target language by fine-tuning on monolingual data.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8093–8104
July 5 - 10, 2020.

A similar *continued training* approach to transfer has been explored for domain adaptation in neural machine translation (Luong and Manning, 2015; Khayrallah et al., 2018); we show that it works with polyglot models for NER, improving performance by up to 3 F_1 over monolingual baselines.

Our second contribution is an explanation of the surprising effectiveness of this technique through an extensive empirical study of polyglot models for NER. We compare several types of neural NER models, including three character (or byte) level architectures, and evaluate transfer across a small (4) and large (10) set of languages. In particular, we find that:

- §4 Other than Byte-to-Span (BTS; Gillick et al., 2016), most NER architectures do not benefit from polyglot training. Still, simpler models than BTS, with more inductive bias, can outperform BTS in both monolingual and polyglot settings.
- §5.2 Polyglot models are more efficient than monolingual models in that for a given level of performance, they require vastly fewer parameters. This suggests that many parameters are shared cross-lingually.
- §4.2 Polyglot weights transfer to unseen languages with mixed results. In particular, transfer can occur when there is high lexical overlap or closely related languages in the polyglot training set.
- §5.3 Languages share a large number of important parameters between each other in polyglot models, and fine-tuning may utilize those parameters to strengthen it's performance.

To our knowledge, ours is the first systematic study of polyglot NER models.

2 Related Work

There is a long history of multilingual learning for NER (Kim et al., 2012; Ehrmann et al., 2011; Täckström, 2012). This work has is driven by an interest in learning NER models for many languages (Cucerzan and Yarowsky, 1999; Pan et al., 2017a) and the relative lack of data for many languages of interest (Das et al., 2017).

Polyglot Models Johnson et al. (2017) and Lee et al. (2017) showed that a single neural MT model could benefit from being trained in a multilingual setting. Gillick et al. (2016) showed similar results for NER, presenting a model that benefited from learning to perform NER on 4 languages at once. We find that other polyglot NER models are rarely better than monolingual models in terms of absolute performance.

Mulcaire et al. (2019) showed that polyglot language model pretraining can help improve performance on NER tasks, although polyglot NER training hurts. However, multilingual BERT (Devlin et al., 2018b), when compared to monolingual BERT performance on NER, shows that polyglot pretraining is not always beneficial for downstream tasks.

Finally, most recently, Kondratyuk and Straka (2019) showed how to train a single model on 75 languages for dependency parsing while retaining competitive performance or improving performance, mostly on low-resource languages. This work is closely related to ours, although we are predominantly interested in how we can leverage polyglot learning to improve performance across *all* languages.

Cross-lingual Models Cross-lingual transfer leverages labeled data from different source languages to augment data for a target language. Rahimi et al. (2019) do this on a massive scale for NER, leveraging over 40 languages for cross-lingual transfer. Xie et al. (2018) employed self-attention to combat word-order differences when transferring parameters from high-resource languages to low-resource.

Much work in this space has looked at how to leverage a mixture of shared features and language-specific features (Kim et al., 2017), similar to domain adaptation techniques Daumé III (2007). Recently, a lot of this work has focused on using adversarial models to force models to learn language-agnostic feature spaces (Chen et al., 2019; Huang et al., 2019). These works show, similar to our work, that it is possible to leverage multilingual data to increase performance across languages.

3 Models

We evaluate three polyglot NER neural models.[1]

3.1 Word Level CRF

The Neural (BiLSTM) CRF is a standard model for sequence labeling tasks (Ma and Hovy, 2016; Durrett and Klein, 2015). Our implementation

[1]We release the code for these models at https://github.com/davidandym/multilingual-NER

Model	Eng	Deu	Nld	Spa	**Avg**	Amh	Ara	Fas	Hin	Hun	Ind	Som	Swa	Tgl	Vie	**Avg**
Character CRF																
Monolingual	84.91	71.39	78.96	82.60	79.45	60.62	43.22	45.11	62.12	60.47	62.14	61.75	68.04	84.13	47.31	59.49
Polyglot	83.38	70.86	79.38	81.64	77.85	59.39	43.25	43.20	62.88	60.86	64.59	65.45	68.32	84.80	49.71	59.87
Finetuned	86.49	72.95	80.91	82.72	**80.82**	59.86	44.69	46.85	68.30	65.21	67.15	66.11	70.07	87.03	51.80	**62.71**
Byte CRF																
Monolingual	85.75	71.42	78.36	81.19	79.18	59.13	44.95	44.76	65.89	57.91	61.46	61.05	67.09	84.46	48.73	59.54
Polyglot	83.79	71.54	79.43	80.25	78.75	57.03	42.88	41.88	65.10	60.46	61.07	62.22	68.40	82.75	47.27	58.90
Finetuned	86.68	73.02	80.09	82.95	**80.69**	59.37	42.69	45.25	67.68	63.91	64.38	64.92	70.78	86.25	51.14	**61.64**
CharNER																
Monolingual	83.83	69.30	79.60	79.46	78.05	54.33	36.31	40.68	62.03	53.04	58.05	56.88	63.70	81.04	39.64	54.53
Polyglot	84.14	69.19	78.94	79.39	77.92	49.64	36.98	37.41	60.02	49.37	55.51	58.56	63.49	79.36	44.50	53.48
Finetuned	85.23	70.60	81.00	82.00	**79.70**	53.46	40.15	39.20	65.57	59.84	60.70	59.09	68.85	84.61	45.47	**57.70**
Byte To Span																
Monolingual	87.91	63.92	71.34	73.07	74.06	48.23	39.41	26.76	19.01	44.51	54.32	58.81	54.27	71.76	26.90	44.50
Polyglot	86.43	71.10	76.11	74.26	**76.98**	46.41	41.59	40.09	55.69	60.53	57.58	62.30	54.78	74.52	43.95	**53.64**
Multilingual BERT																
Monolingual	90.94	81.50	88.62	88.16	**87.31**	-	48.36	56.42	72.52	66.99	78.32	62.69	72.18	86.13	54.18	66.75
Polyglot	90.67	80.96	87.48	87.04	86.53	-	48.33	56.92	74.81	68.16	77.56	59.29	71.92	87.59	57.06	66.84
Finetuned	91.08	81.27	88.74	86.87	86.99	-	49.94	54.67	76.83	69.52	80.14	62.70	73.16	88.05	56.74	**69.97**

Table 1: Performance for monolingual, multilingual, and finetuned models trained on either CoNLL (left) or LORELEI (right) data sets. The results are taken from the best model out of 5 random seeds, as measured by dev performance. Almost every model achieves the best performance in the finetuned setting, indicating that multilingual pretraining is learning transferable parameters, but multilingual models are not able to use them effectively across all languages simultaneously. Note that we do not evaluate Amharic with mBERT, because the Amharic script is not a part of mBERT's vocabulary.

broadly follows the description in Lample et al. (2016), and we consider three different variants of this model.

The first two are character- and byte-level models.[2] We consider these since Gillick et al. (2016) showed that multilingual transfer could occur across byte-level representations and we were interested in whether characters produced similar results when more diverse languages were involved. Each word passes through a multi-layer BiLSTM as a sequence of characters or bytes to produce word-level representations. Word-level representations feed into a sentence-level BiLSTM, which outputs, for each time step, logits for all possible labels. The logits are then fed into a CRF model (Lafferty et al., 2001) trained to maximize the log-likelihood of the gold label sequences.

The third variant of this model uses contextualized representations from multilingual BERT (mBERT) (Devlin et al., 2018b). This model is similar to the one described above, with the key difference being that word-level representation are obtained using a pretrained subword-level BERT model, as opposed to being built from raw characters/bytes. As is done in the original BERT paper, we treat the representation of the first subword of each word as a representation for that word, and take the concatenation of the outputs of the last 4 layers at that subword position as our final word representation.

3.2 CharNER

CharNER (Kuru et al., 2016) is a deep neural sequence labeling architecture which operates strictly at the character level during training, but uses word-level boundaries during inference. The model runs a 5-layer BiLSTM over sequences of characters, and is trained to predict the NER tag for each character of the sequence (without BIO labels). During inference a Viterbi decoder with untrained transition parameters enforces consistent character level tags across each word; no heuristics and little post-processing is necessary to obtain word-level BIO labels.

To compare with the other architectures, we apply this model to bytes and evaluate its polyglot performance. Intuitively, we expect this model to do better than a word-level CRF at seeing beneficial transfer across languages, as it is closer to the model of Gillick et al. (2016): a deep, byte-level model that performs inference at the level of individual bytes.

[2]Early experiments found these models suffered much less from multilingual training than subword/word models.

3.3 Byte to Span (BTS)

BTS is a sequence-to-sequence model operating over byte sequences (Gillick et al., 2016). The input consists of a window of UTF-8 bytes, and the output is sequences with sufficient statistics of labeled entity spans occurring in the input sequence.[3] Because byte sequences are long BTS operates over a sliding window of 60 bytes, treating each segment independently; the model's entire context is always limited to 60 bytes. By consuming bytes and producing byte annotations, it has the attractive quality of being truly language-agnostic, without any language specific preprocessing.

Despite obviating the need for language-specific preprocessing, BTS achieves comparable results to more standard model architectures with no pretraining information. Additionally, it showed significant improvement in monolingual CoNLL performance after being trained on all 4 CoNLL languages. In this paper, we find that this trend holds in our multilingual settings, although our results show lower overall numbers to those reported in Gillick et al. (2016).[4]

3.4 Hyperparameters

All experiments are run on GeForce RTX 2080 Ti GPUs, using Tensorflow (Abadi et al., 2016).

CRF The character- and byte-level neural CRF use a sub-token BiLSTM encoder with 2-layers and 256 hidden units. The sentence-level BiLSTM has 1-layer with 256 hidden units. All characters and bytes have randomly initialized embeddings of size 256. We optimized these parameters with grid-search over 1-3 layers at each level and hidden sizes of {128, 256, 512}. We train using Adam with a learning rate of 0.001 and tune the early stop parameter for each model based on development set F1 performance.

CharNER Our CharNER model operates over bytes rather than characters. It uses the same hyperparameters reported in Kuru et al. (2016), (5 layers with hidden size 128, Adam Optimizer) with a byte dropout of 0.2, and dropout rates of 0.8 on the final layer, and 0.5 on the other layers. We also train our models using a learning rate of 0.001 and early stop based on development set F1 performance.

BTS For BTS we use the same training scheme and hyperparameters reported in Gillick et al. (2016).[5] Since we do not have document-level information in LORELEI, we treat each separate language dataset as its a whole document and slide a window across the entire dataset at once. We train using SGD (Adam performed much worse), with a learning rate of 0.3, and similarly, early stop based on development set F1 performance.

Language	Code	Family	Genus	Script	# Train Sent.
CoNLL					
English	eng	Indo-European	Germanic	Latin	11,663
Spanish	spa	Indo-European	Romance	Latin	8,323
German	deu	Indo-European	Germanic	Latin	12,152
Dutch	nld	Indo-European	Germanic	Latin	15,806
LORELEI					
Amharic	amh	Afroasiatic	Semitic	Ge'ez	4,923
Arabic	ara	Afroasiatic	Semitic	Arabic	4,990
Farsi	fas	Indo-Iranian	-	Arabic	3,849
Hindi	hin	Indo-European	Indo-Aryan	Devanagari	4,197
Hungarian	hun	Uralic	Ugric	Latin	4,846
Indonesian	ind	Austronesian	Malayo-Polynesian	Latin	4,605
Somali	som	Afroasiatic	Cushitic	Latin	3,253
Swahili	swa	Niger-Congo	Bantu	Latin	3,318
Tagalog	tgl	Austronesian	-	Latin	4,780
Vietnamese	vie	Austroasiatic	Vietic	Latin (Viet.)	4,042
LORELEI - held out for zeroshot					
Russian	rus	Indo-European	Slavic	Cyrillic	6,480
Bengali	ben	Indo-European	Indo-Aryan	Bengali	7,538
Uzbek	uzb	Turkic	-	Arabic	11,323
Yoruba	yor	Niger-Congo	-	Latin	1,753

Table 2: Different sets of languages we used, their sources, family and genus, script, and training set size.

4 Experiments

Each LORELEI language has less than half the data of a CoNLL language, but in total, the two datasets are roughly equal in size. The CoNLL setting consists of European languages in the same alphabet, and prior work has shown beneficial transfer in this setting (Gillick et al., 2016). LORELEI is more challenging because it contains more distantly related languages.

We train a monolingual NER model for each language (14 models) and two polyglot models: CoNLL and LORELEI. For polyglot training we concatenate each annotated language-specific dataset into one combined corpus. Because our language-specific datasets are comparable in size

[3] For a `PER` span at bytes 5-10, the correct output sequence is y = `S:5, L:5, PER, STOP`

[4] We reimplemented BTS based on correspondence with the model authors. We matched the published results on CoNLL English, and the same overall trends, but could not match the other three CoNLL languages. Despite significant effort, two differences remained: the authors could not share their proprietary implementation or deep learning library, and reported using more byte segments than is available in our CoNLL dataset.

[5] 4 layers with 320 hidden units, byte dropout of 3.0 and layer dropout of 5.0.

we do not correct for minor size differences.[6] All models were trained over 5 random seeds, with the best model selected by development performance. For polyglot models, we select the best model using the average development performance across all languages.

Results Table 1 reports test performance. With few exceptions, polyglot training does worse than monolingual. In some cases, the two settings do nearly the same (such as Character and mBERT CRFs on LORELEI) but we do not see improved results from a polyglot model.

Murthy et al. (2018) found that languages with different label distributions do worse for transfer. We find large label distribution changes in CoNLL, but not LORELEI. To determine if this could explain polyglot NER failures in CoNLL, we allow our CRF models to learn language-specific label distributions via language-specific CRF transition parameters. However, we saw little difference in the results for either CoNLL or LORELEI (no more than 0.5 F1 on any language). This suggests that other factors are preventing more language transfer.

The exception to these observations is the BTS model, which showed significant improvements in the polyglot settings, matching the conclusion of Gillick et al. (2016). However, our implementation failed to match the higher numbers of the original paper, and so the model is significantly worse overall compared to the other NER models. Perhaps the unique architecture of BTS enables it to improve in the polyglot setting. However, if BTS requires more training data to achieve results similar to the other models, the polyglot improvements may not hold up.

Conclusion Polyglot NER models fail to improve over their monolingual counterparts, despite using 4 (CoNLL) or 10 (LORELEI) times more labeled data. Discrepancies of label priors between languages do not, by themselves, account for this.

4.1 Target Language Polyglot Adaptation

While polyglot models perform worse than monolingual models, they are competitive. This suggests that polyglot models may be successfully learning multilingual representations, but that the optimization procedure is unable to find a global minimum for all languages. To test this theory, we fine-tune the polyglot model separately for each language. We treat the parameters of the polyglot NER models as *initializations* for monolingual models of each language, and we train these models in the same fashion as the monolingual models, with the exception of using a different initial step size.[7] With few exceptions, fine-tuned polyglot models surpass their monolingual counterparts (Table 1), improving up to 3 F_1 over monolingual baselines.

Language	Monoling.	Poly. (Zero-shot)	Poly. (Fine-tuned)
Russian	**43.97**	1.61	41.55
Bengali	76.10	2.08	**76.63**
Uzbek	**65.39**	14.54	61.10
Yoruba	62.66	29.02	**64.95**

Table 3: F1 of a Byte-level CRF on 4 different lorelei language datasets, compared to the performance of the multilingual model which was not trained on any of these 4 languages, as well as the multilingual model after finetuning. The results are mixed - moreover, zero-shot performance does not seem to be a good indicator of transferability.

Conclusion This demonstrates that the polyglot models are in fact learning more from observing multiple languages, and that this information can transfer to each language. Additionally, this indicates that the ideal optima for a monolingual model may not be achievable using standard training objectives without observing other languages; we found more regularization did not help the monolingual models. However, jointly optimizing all languages naively may provide too challenging an optimization landscape to obtain that optima for each language simultaneously.

4.2 Novel language transfer

Finally, since the polyglot models demonstrate the ability to transfer information between languages, we ask: can these models generalize to unseen languages? We consider a similar approach to the previous section, except we now fine-tune the polyglot model on a novel language for which we have supervised NER data. In this setting, we only consider byte-level models, since byte vocabularies mean we can use the same parameters on unseen languages with different character sets. We select 4 additional LORELEI languages: Rus-

[6] A uniform sampling strategy is recommended for language combinations with significant size discrepancies.

[7] We use the Adam optimizer settings saved from multilingual training.

sian, Yoruba, Bengali, and Uzbek. For comparison, we train monolingual Byte CRF models (from scratch), following the same optimization protocols, as described above.

Table 3 shows results for the monolingual model, polyglot fine-tuned, and the polyglot model evaluated without any fine-tuning (zero-shot). Unsurprisingly, the polyglot model does poorly in the zero-shot setting as it has **never** seen the target language. However, sharing a script with some languages in the polyglot training set can lead to significantly better than random performance (as in the case of Yoruba and Uzbek). In the fine-tuning setting, the results are mixed. Yoruba, which enjoys high script overlap with the polyglot training set, sees a large boost in performance from utilizing the polyglot parameters, whereas Uzbek, which has moderate script overlap but no family overlap, is hurt by it. Russian and Bengali have no script overlap with the polyglot training set, but Bengali, which is closely related to Hindi (sharing family and genus) sees a moderate amount of transfer, while Russian, which is not closely related to any language in the training set, is negatively impacted from using the polyglot weights.

Conclusion The transferability of the polyglot parameters to unseen languages depends on a variety of factors. We conjecture that these factors are partially connected to relatedness to languages in the original polyglot training set.

5 How do Polyglot Models Learn?

We now turn our attention towards understanding how polyglot models are transferring information across languages. We examine the types of errors made in each setting, as well as how polyglot models efficiently use parameters and how parameter weights are shared across languages.

5.1 Error Analysis

We broadly examine the types of errors made across each of our regimes, focusing on results from the Byte-CRF model. To explore what kinds of errors polyglot fine-tuning targets we plot, in Figure 1, the counts of recall errors (including O-tags) on validation data made by the monolingual and polyglot models, compared to the fine-tuned model. We find that polyglot models tend to make more errors on O-tags, indicating a tendency towards making *precision* errors, but that

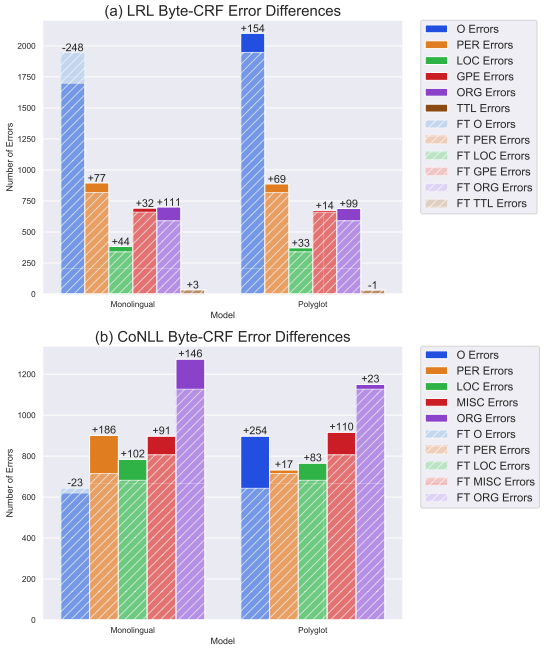

Figure 1: (a) The count of errors made by the LORELEI Byte-CRF monolingual and polyglot models, compared to the fine-tuned (FT) models (across all languages), (b) shows the CoNLL setting. Deltas (Errors minus FT Errors) are displayed on top. Polyglot models tend to make more errors on O-tagged tokens (precision errors) than monolingual models. However, fine-tuning tends to recover these errors to nearly monolingual performance. In the CoNLL regime, polyglot models make fewer errors on PER and ORG tags, and fine-tuned models generally maintain that error rate.

fine-tuning tends to correct this trend back towards monolingual performance. We additionally find that, compared to monolingual models, fine-tuned models do much better PER and ORG tags (in both LORELEI and CoNLL settings). However, the same is not true for polyglot LORELEI models, indicating that some of this transfer comes from the combination of polyglot and fine-tune training.

One reason that polyglot fine-tuned models may perform better than monolingual models is the larger number of entities they see during training. Many languages contain entities in their validation set, which appear in the training sets of *other languages*. We identify such "common entities" as entities in the validation set of a language l which share some level of surface form overlap (either n-gram or exact match)[8] and type with an entity appearing in the training set of lan-

[8]We explore n-gram overlap with $n = 4, 5, 6, 7, 8$ and exact name overlap. We report the average rate across each granularity.

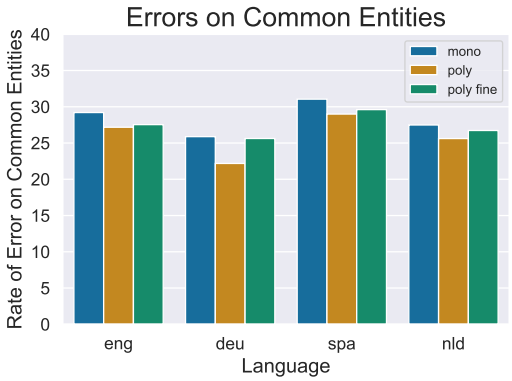

Figure 2: The rate of errors containing surface forms that overlap with an entity of the same type in other languages' training set. We report the harmonic mean between the rate in precision and recall errors, for the monolingual, polyglot, and fine-tuned byte-CRF models. We find that polyglot models have a lower rate of errors on entities which appear in other languages' training sets, indicating that they are benefiting from the higher quantity of entities seen.

guage $l' \neq l$. We plot the average error rate (defined as the harmonic mean between the rate of precision errors and the rate of recall errors) of the CoNLL Byte-CRF model in Figure 2. We find that polyglot models have a lower error rate on "common entities" than monolingual models, indicating that such entities are a source of transfer in polyglot NER. We also see that language-specific fine-tuning tends to increase the error rate, either due to forgetting or simply to decreasing errors on "non-common entities" during fine-tuning.

5.2 Polyglot Parameter Efficiency

Many studies have demonstrated that modern neural models have enormous capacity, and that not all parameters are needed to model the target function (LeCun et al., 1990; Hinton et al., 2015; Frankle and Carbin, 2019; Sanh et al., 2019). Let us assume that it takes M^l parameters to learn a monolingual NER model for language l. If we sought to train monolingual models for each language in L, we would need $\hat{M} = \sum_{l \in L} M^l$ parameters. Does a polyglot model trained on these languages need $\hat{M}$ parameters? Perhaps the polyglot NER model is partitioning its parameters by language, and little sharing occurs across languages, so the full $\hat{M}$ parameters are needed. In this case, the negative results for polyglot learning could be explained by the under-parameterization of the model. Conversely, the model could be sharing parameters across many languages, effectively learning cross-lingual representations. In this case, we would expect the model to need much fewer than $\hat{M}$ parameters, and the over-sharing of parameters across languages could explain the poor polyglot performance.

Model Compression To explore polyglot model behavior, we utilize model compression techniques, which have the goal of compressing a large number of parameters into a smaller amount with minimal loss in overall model accuracy. We use magnitude weight pruning (Han et al., 2015) to answer two questions: (1) How many more parameters do polyglot models require than monolingual models? (2) Does fine-tuning learn an equally compact solution to that of monolingual training?

We analyze the byte-level CRF because they are stronger than, or comparable to, all other models with no pretraining, and have the same number of parameters across all languages and settings (monolingual, polyglot, and fine-tuned). We perform our analysis on models without pretraining, as we wish to isolate the effects of polyglot learning on our models from external polyglot resources. We prune the lowest magnitude weights of each model in 10% increments and plot the average[9] performance over time in Figure 3. Additionally, we define "over-pruning" to occur for language l and model m when pruning causes the performance of model m on language l to decrease by more than 1 F1 from model m 's original performance. We plot the pruning threshold for each language and model[10] before "over-pruning" occurs in Figure 3 as well.

We find that polyglot models require more parameters than monolingual models to maintain their performance, but are significantly more efficient, i.e. they need much fewer than $\hat{M}$ parameters. For example, the CoNLL polyglot model needs 60% of its parameters to maintain performance on all languages; English, Spanish, and Dutch require fewer parameters still. Compared to the total number of parameters needed by the four individual monolingual models combined ($\hat{M}$), the polyglot model needs only 30% of that, although this is paid for by an average decrease of 0.33 F1. This suggests that polyglot performance suffers due to over-sharing parameters, rather than

[9] Averaged across all CoNLL or LORELEI languages.

[10] For polyglot models we report the percentage required to maintain performance on each individual language using the *same* model.

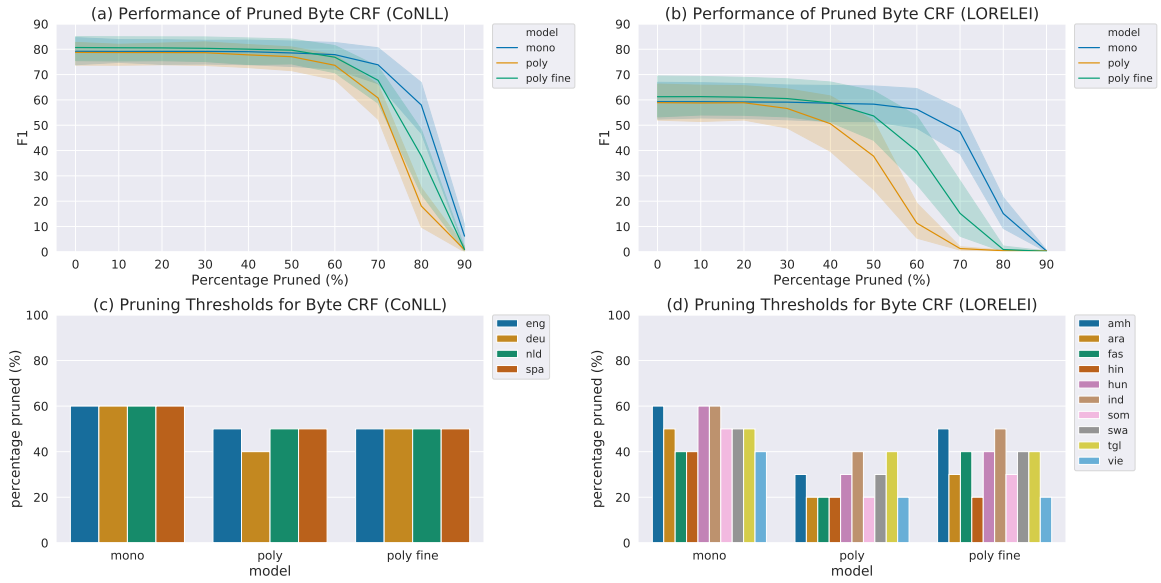

Figure 3: (a & b) Average F1 of Byte-CRF models as the pruning threshold increases. We find that monolingual models learn much more sparse solutions than polyglot models. Interestingly, fine-tuning does not recover the sparsity of the monolingual models. (c & d) The pruning thresholds before language performance drops by more than 1 F1 for each model. In the CoNLL setting, languages share nearly equally sparse solutions. However, in the LORELEI setting, the sparsity across all languages exhibits high variance, even in the fully shared polyglot model.

under-sharing, during joint optimization.

Additionally, we find that fine-tuning the polyglot models does not recover as sparse a solution as monolingual training. This finding suggests that either fine-tuning utilizes polyglot parameters to learn a denser solution than monolingual models, or that fine-tuning retains several high-magnitude polyglot weights not crucial to the target language. In the latter case, more sophisticated pruning criteria may be better suited to determining the sparsity of fine-tuned models, despite recent evidence indicating the strength of simple magnitude pruning (Gale et al., 2019).

5.3 Important Weights Across Languages

In addition to measuring the parameter efficiency of the polyglot models, we are interested in knowing *how much* overlap exists between the parameters which are most important for different languages, and how those parameters change during fine-tuning. This answers two important questions: 1) How do languages utilize shared polyglot parameters? 2) Does fine-tuning benefit from many or few polyglot weights?

To measure overlap between important weights for each language in a polyglot model, we compare the language-specific Fisher information matrix diagonals of the polyglot model. The Fisher information matrix has been used in this way to measure individual parameter importance on a specific task, and has been shown to be effective for retaining important information across tasks during sequential learning (Kirkpatrick et al., 2016; Thompson et al., 2019).

For a given language l with N training examples we estimate the Fisher information matrix F^l with the *empirical* Fisher information matrix $\bar{F}^l$. F^l is computed via[11]

$$\frac{1}{N}\sum_{i=1}^{N} \mathop{\mathbb{E}}_{y\sim p_\theta} \left[\nabla_\theta \log p_\theta(y|x_i) \nabla_\theta \log p_\theta(y|x_i)^T\right]$$

We take the diagonal values $\bar{F}_{i,i}$ as an assignment of importance to θ_i.

To compute the overlap of important weights shared between two tasks, we take the top 5%, 25%, and 50% of weights from each layer for each task (given by the tasks' Fishers) and calculate the percentage overlap between them. We do this for two settings: First, we consider the percentage of weights shared between a specific language and all other languages in a polyglot model. Second, we examine the percentage of weights that remain important to a particular language after fine-tuning. We plot the average overlap across all lan-

[11] The expectation over $y \sim p_\theta$ is approximated by sampling exactly from the posterior of each x_i. We take 1,000 samples for each example.

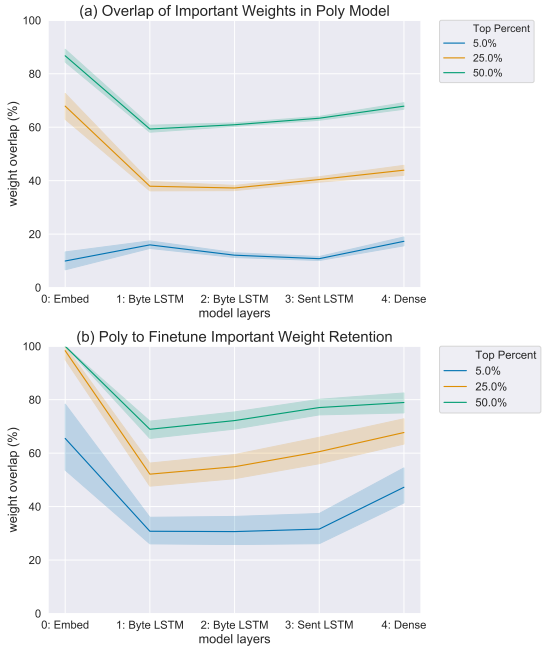

Figure 4: (a) Percentage of important weight overlap between a single language and *all other* languages in the polyglot Byte-CRF LORELEI model (averaged over all languages). The top 5% of parameters for each language share little overlap with other languages, implying that the most important weights for each language are uniquely important to that language. (b) Overlap of important weights between the polyglot and fine-tuned Byte-CRF LORELEI model, for a given language (averaged over all languages). Only 30% of the top 5% of weights important to a given language are retained after fine-tuning, suggesting that fine-tuning targets the *most important* parameters for a language.

guages for each setting with our LORELEI Byte-CRF models in Figure 4.

We find that languages share a high number of important weights between each other in the polyglot model (40% overlap in the top 25% of weights of the LSTM layers), which helps explain how polyglot models are competitive, with fewer parameters, than multiple monolingual models. Interestingly, however, we find that the *most* important weights (top 5%) for each language share little overlap, implying that in polyglot learning, each language acquires parameters that are uniquely important to that language.

We additionally find that fine-tuning does not shift the importance of a significant number of weights (more than half of the top 25% important weights for a language in the polyglot model remain similarly important after fine-tuning). Surprisingly, the parameters that were most important to a language in the polyglot model are the parameters that are the most affected during fine-tuning for that language. Thus, we see that language-specific fine-tuning retains the importance of many shared parameters, but the *most* important weights to that language are significantly affected.[12]

6 Conclusions

We explore the benefits of polyglot training for NER across a range of models. We find that, while not all models can benefit in performance from polyglot training, the parameters learned by those models can be leveraged in a language-specific way to consistently outperform monolingual models. We probe properties of polyglot NER models, and find that they are *much* more efficient than monolingual models in terms of the parameters they require, while generally maintaining a competitive performance across all languages. We show that the high amount of parameter sharing in polyglot models partially explains this, and additionally find that language-specific fine-tuning may use a large portion of those shared parameters. In future work, we will explore whether the observed trends hold in much larger polyglot settings, e.g. the `Wikiann` NER corpus (Pan et al., 2017b).

Finally, regarding the sharing of weights between languages in polyglot models, our key conclusion is that standard training objectives are unable to find an optimum which simultaneously achieves high task performance across all languages. With this in mind, exploring different training strategies, such as multi-objective optimization, may prove beneficial (Sener and Koltun, 2018). On the other hand, when the objective is to maximize performance on a single target language it may be possible to improve the proposed fine-tuning approach further using methods such as elastic weight consolidation (Kirkpatrick et al., 2016).

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments.

[12]Note that typically it is not reasonable to compare the weights of two different neural networks, as they are unidentifiable (Goodfellow and Vinyals, 2015). However, since one model is initialized from the other, we believe it is reasonable to characterize how weights shift during language-specific fine-tuning.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Xilun Chen, Ahmed Hassan Awadallah, Hany Hassan, Wei Wang, and Claire Cardie. 2019. Multi-source cross-lingual model transfer: Learning what to share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3098–3112, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale.

Ryan Cotterell and Kevin Duh. 2017. Low-resource named entity recognition with cross-lingual, character-level neural conditional random fields. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 91–96. Asian Federation of Natural Language Processing.

Silviu Cucerzan and David Yarowsky. 1999. Language independent named entity recognition combining morphological and contextual evidence. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

Arjun Das, Debasis Ganguly, and Utpal Garain. 2017. Named entity recognition with word embeddings and wikipedia categories for a low-resource language. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 16(3):18.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

Jacob Devlin. 2018. Multilingual bert readme document.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018a. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Greg Durrett and Dan Klein. 2015. Neural crf parsing. In *Proceedings of the Association for Computational Linguistics*, Beijing, China. Association for Computational Linguistics.

Maud Ehrmann, Marco Turchi, and Ralf Steinberger. 2011. Building a multilingual named entity-annotated corpus using annotation projection. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 118–124, Hissar, Bulgaria. Association for Computational Linguistics.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*.

Trevor Gale, Erich Elsen, and Sara Hooker. 2019. The state of sparsity in deep neural networks. *ArXiv*, abs/1902.09574.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2016. Multilingual language processing from bytes. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1296–1306. Association for Computational Linguistics.

Ian J. Goodfellow and Oriol Vinyals. 2015. Qualitatively characterizing neural network optimization problems. *CoRR*, abs/1412.6544.

Song Han, Jeff Pool, John Tran, and William J. Dally. 2015. Learning both weights and connections for efficient neural networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1135–1143, Cambridge, MA, USA. MIT Press.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Lifu Huang, Heng Ji, and Jonathan May. 2019. Cross-lingual multi-level adversarial transfer to enhance low-resource name tagging. 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 3823–3833, Minneapolis, Minnesota. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Huda Khayrallah, Brian Thompson, Kevin Duh, and Philipp Koehn. 2018. Regularized training objective for continued training for domain adaptation in neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 36–44.

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, Copenhagen, Denmark. Association for Computational Linguistics.

Sungchul Kim, Kristina Toutanova, and Hwanjo Yu. 2012. Multilingual named entity recognition using parallel data and metadata from wikipedia. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 694–702, Stroudsburg, PA, USA. Association for Computational Linguistics.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2016. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 114 13:3521–3526.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. 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 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Onur Kuru, Ozan Arkan Can, and Deniz Yuret. 2016. Charner: Character-level named entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 911–921, Osaka, Japan. The COLING 2016 Organizing Committee.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

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. Association for Computational Linguistics.

Yann LeCun, John S. Denker, and Sara A. Solla. 1990. Optimal brain damage. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 598–605. Morgan-Kaufmann.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 76–79.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074. Association for Computational Linguistics.

Lluís Màrquez, Adrià de Gispert, Xavier Carreras, and Lluís Padró. 2003. Low-cost named entity classification for Catalan: Exploiting multilingual resources and unlabeled data. In *Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition*, pages 25–32, Sapporo, Japan. Association for Computational Linguistics.

Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. 2017. Cheap Translation for Cross-Lingual Named Entity Recognition. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Phoebe Mulcaire, Jungo Kasai, and Noah A. Smith. 2019. Polyglot contextual representations improve crosslingual transfer. 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 3912–3918, Minneapolis, Minnesota. Association for Computational Linguistics.

Rudra Murthy, Anoop Kunchukuttan, and Pushpak Bhattacharyya. 2018. Judicious selection of training data in assisting language for multilingual neural ner. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 401–406. Association for Computational Linguistics.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1470–1480, Vancouver, Canada. Association for Computational Linguistics.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017a. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1946–1958.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017b. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, pages 527–538.

Oscar Täckström. 2012. Nudging the envelope of direct transfer methods for multilingual named entity recognition. In *Proceedings of the NAACL-HLT Workshop on the Induction of Linguistic Structure*, pages 55–63, Montréal, Canada. Association for Computational Linguistics.

Brian Thompson, Jeremy Gwinnup, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 2019. Overcoming catastrophic forgetting during domain adaptation of neural machine translation. 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 2062–2068, Minneapolis, Minnesota. Association for Computational Linguistics.

Yulia Tsvetkov, Sunayana Sitaram, Manaal Faruqui, Guillaume Lample, Patrick Littell, David Mortensen, Alan W Black, Lori Levin, and Chris Dyer. 2016. Polyglot neural language models: A case study in cross-lingual phonetic representation learning. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Shyam Upadhyay, Nitish Gupta, and Dan Roth. 2018. Joint multilingual supervision for cross-lingual entity linking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2495, Brussels, Belgium. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of bert. *arXiv preprint arXiv:1904.09077*.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 369–379, Brussels, Belgium. Association for Computational Linguistics.