Toward the Limitation of Code-Switching in Cross-Lingual Transfer

Yukun Feng[1] **Feng Li**[2] **Philipp Koehn**[1]
[1]Johns Hopkins University
[2]University of Illinois Urbana-Champaign
{yfeng55, phi}@jhu.edu, fengl3@illinois.edu

Abstract

Multilingual pretrained models have shown strong cross-lingual transfer ability. Some works used code-switching sentences, which consist of tokens from multiple languages, to enhance the cross-lingual representation further, and have shown success in many zero-shot cross-lingual tasks. However, code-switched tokens are likely to cause grammatical incoherence in newly substituted sentences, and negatively affect the performance on token-sensitive tasks, such as Part-of-Speech (POS) tagging and Named-Entity-Recognition (NER). This paper mitigates the limitation of the code-switching method by not only making the token replacement but considering the similarity between the context and the switched tokens so that the newly substituted sentences are grammatically consistent during both training and inference. We conduct experiments on cross-lingual POS and NER over 30+ languages, and demonstrate the effectiveness of our method by outperforming the mBERT by 0.95 and original code-switching method by 1.67 on F1 scores.

1 Introduction

Recent studies(Devlin et al., 2019; Lample and Conneau, 2019; Conneau et al., 2020a) have shown the success of multilingual corpus pre-training for cross-lingual knowledge transfer. Some works(Qin et al., 2020; Yang et al., 2021) further addressed the effectiveness of code-switching on improving the performance of multilingual models on zero-shot cross-lingual tasks(Conneau et al., 2018; Hu et al., 2020). Though code-switching has shown great potential and strong generalization ability on the semantic representation, the newly switched sequence fails to consider the token-level coherence. Specifically, a code-switched sentence consists of tokens from various languages, and such words are likely to cause grammatical incoherence, resulting in an inconsistent context space for the newly substituted sentence. Also, code-switched tokens are conditioned on the original context, for which, during the optimization phase, the embedding of substituted tokens is greatly affected. However, such updates do not consider the token-level dependency in their own language and will likely cause inconsistent embedding space during inference. Therefore, even though code-switching benefits most of the sentence-level cross-lingual tasks, it still limits the performance of token-level prediction.

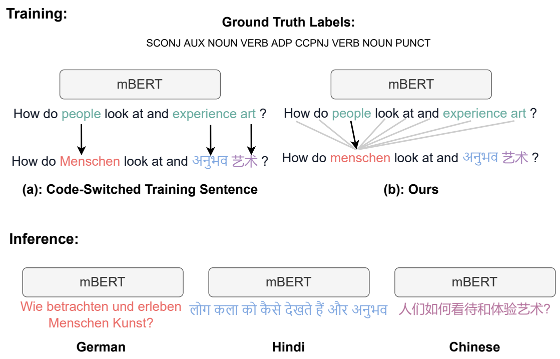
Training:
Ground Truth Labels:
SCONJ AUX NOUN VERB ADP CCPNJ VERB NOUN PUNCT
mBERT
How do people look at and experience art ?
How do Menschen look at and अनुभव 艺术 ?
(a): Code-Switched Training Sentence
mBERT
How do people look at and experience art ?
How do menschen look at and अनुभव 艺术 ?
(b): Ours
Inference:
mBERT
Wie betrachten und erleben Menschen Kunst?
German
mBERT
लोग कला को कैसे देखते हैं और अनुभव
Hindi
mBERT
人们如何看待和体验艺术?
Chinese

Figure 1: Overview of the training and inference for POS in code-switching and our settings.

To address such issues, we use an alignment strategy to reduce the gap between original context and code-switched tokens during training. Specifically, code-switched tokens are mapped to the original context so that the newly switched sentences are expected to be grammatically coherent and maintain the same contextual space in the same language. Meanwhile, our approach enables the substituted tokens not to significantly affect the token-level coherence in their own language and keep a consistent embedding space during inference. We conduct experiments on zero-shot cross-lingual POS and NER and outperforms both the mBERT and the original code-switching method by 0.77 and 2.08 on NER and 1.13 and 1.27 on POS, respectively. We also make further analysis regarding the performance on low-resource languages, code-switch contribution, substitution strategy, and token-level coherence.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 5966–5971
December 7-11, 2022 ©2022 Association for Computational Linguistics

af	ar	bg	bn	de	el	es	et	eu	fa	fi
fr	he	hi	hu	id	it	ja	jv	ko	ml	mr
ms	nl	pt	ru	sw	ta	tl	tr	ur	vi	zh

Table 1: List of 33 languages for building the dictionary.

2 Approach

We conduct cross-lingual tasks in the zero-shot setting, in which only English labeled sentences with code-switching augmentation are used for training, and evaluation is performed in all other languages.

2.1 Code-Switching Augmentation

Multilingual Dictionary Construction To build the dictionary that maps English tokens to other languages in Table 1, we adopt the parallel sentences from CCMatrix(Schwenk et al., 2021), and use fast_align(Dyer et al., 2013) to align tokens from parallel sentences. To keep languages being equally considered, we sample no more than 1 million sentence pairs for each language. For each language lg, a bilingual dictionary D^{lg} is built as a one-to-many structure based on extracted alignment, where each English token $\mathrm{t_{en}}$ has several candidates as $\mathrm{C}^{\mathrm{lg}}_{\mathrm{t_{en}}} = \{\mathrm{c}^{\mathrm{lg}}_1, ...\mathrm{c}^{\mathrm{lg}}_{\mathrm{m}}\}$, and the corresponding sampling probability are defined as $\beta^{\mathrm{lg}}_{\mathrm{C}} = \{\beta^{\mathrm{lg}}_{\mathrm{c_1}}, ...\beta^{\mathrm{lg}}_{\mathrm{c_m}}\}$. We merge such bilingual dictionaries into one unified dictionary D with respect to their keys of English tokens. Then for each English token $\mathrm{t_{en}}$, we have its corresponding candidates as $\mathrm{C_{t_{en}}} = \{\mathrm{c_1}, ...\mathrm{c_n}\}$, and sampling probability as $\beta_{\mathrm{C}} = \{\beta_{\mathrm{c_1}}, ...\beta_{\mathrm{c_n}}\}$, where n refers to the number of candidates among all languages, and the sampling rates are normalized after the merge.

Token Substitution Given the English training sentences, we decide whether to substitute an English token based on a substitution ratio α. For an English token $\mathrm{t_{en}}$ that is selected for substitution, the candidate token $\mathrm{t_X}$ is sampled following the probability distribution in dictionary D. We adopt the dynamic substitution to sample different substitution sequences for each epoch. We present analysis of the substitution strategy in Section 3.3.

2.2 Cross-Lingual Transfer

Our approach is built on both the code-switched sentence $\mathrm{S_X} = \{\mathrm{S^1_X}, ...\mathrm{S^{L_X}_X}\}$, and original English sentence $\mathrm{S_{en}} = \{\mathrm{S^1_{en}}, ...\mathrm{S^{L_{en}}_{en}}\}$. To avoid the influence of the grammatical incoherence,we use an alignment network to align substituted tokens with original English context.

Token-Level Alignment We map the code-switched sentence to the English context by computing the similarity between substituted tokens and English tokens. The similarity scores are further used as the weights to aggregate the embedding of English tokens, as the calculated potential for substituted tokens in the switched sequence.

We use the multilingual BERT (mBERT) to encode English and code-switched sentences into contextualized embedding as $\mathrm{H_{en}} = \mathrm{mBERT}(\mathrm{S_{en}})$, $\mathrm{H_X} = \mathrm{mBERT}(\mathrm{S_X})$ respectively. For any token $\mathrm{S^{(k)}_X}$, the similarity score $\mathrm{score^{(k)}}$ is computed as:

$$\mathrm{score^{(k)}} = \mathrm{H^{(k)}_X} * \mathrm{H_{en}}^{\mathrm{T}} \tag{1}$$

The final potential of the code-switched token $\widetilde{\mathrm{H}}^{\mathrm{(k)}}_{\mathrm{X}}$ is the weighted sum of the contextualized embedding $\mathrm{H_{en}}$, calculated along the time axis.

$$\alpha^{\mathrm{(k)}}_{\mathrm{t}} = \frac{\exp\left(\mathrm{score^{(k)}_t}\right)}{\sum_{\mathrm{t'}} \exp\left(\mathrm{score^{(k)}_{t'}}\right)} \tag{2}$$

$$\widetilde{\mathrm{H}}^{\mathrm{(k)}}_{\mathrm{X}} = \sum_{\mathrm{t}} \alpha^{\mathrm{(k)}}_{\mathrm{t}} \mathrm{H^{(t)}_{en}} \tag{3}$$

Eventually, the newly aligned code-switched sequence of tokens is represented as $\widetilde{\mathrm{H}}_{\mathrm{X}}$.

Training and Inference During training, the alignment is applied before the layer input in the mBERT. We compute $\widetilde{\mathrm{H}}^{\mathrm{i}}_{\mathrm{X}}$ from the original layer input pairs ($\mathrm{H^i_X}$, $\mathrm{H^i_{en}}$) as in Eq.3, where i refers to the layer index in mBERT. $\widetilde{\mathrm{H}}^{\mathrm{i}}_{\mathrm{X}}$ will be the new layer input for code-switched sentence. As lower layers in BERT learns about the syntax information, we only apply the alignment in lower six layers to better align the representations for switched tokens. To optimize the effect of the ground-truth labels on both the English and code-switched sequences during training, the final loss is described as: $\mathrm{Loss} = \frac{\mathrm{L(H_{en},Label)+L(H_X,Label)}}{2}$, where L refers to the loss function. In the inference, we do not use any code-switch to sentence tokens.

3 Experiment

3.1 Settings

We conduct experiments on two widely-used cross-lingual datasets for token-level prediction, Universal Dependencies for POS and WikiANN for NER (Pan et al., 2017). We adopt mBERT [1] with the

[1] We adopt the training pipeline from Transformers and Datasets: https://github.com/huggingface/transformers, https://github.com/huggingface/datasets

Model	af	ar	bg	nl	en	et	fi	fr	de	el	he	hi	hu	id	it	ja	kk
mBERT	87.3	66.4	87.5	87.6	95.3	84.6	82.9	79.5	85.6	53.2	76.8	75.1	80.9	74.3	85.4	58.5	72.5
M$_{switch}$	88.7	70.3	88.8	87.4	95.0	83.7	82.5	79.4	85.8	51.0	78.6	75.7	81.7	73.6	84.6	58.2	72.9
M$_{ours}$	87.6	69.0	87.7	88.1	95.0	84.5	83.2	79.7	85.9	57.7	80.2	77.0	80.0	74.4	85.6	58.9	74.6
	ko	zh	mr	fa	pt	ru	es	tl	ta	te	th	tr	ur	vi	yo	Avg	Diff
mBERT	57.4	68.6	70.5	67.4	88.9	84.1	87.8	83.0	63.9	73.5	53.5	70.3	65.2	62.8	61.2	74.73	-
M$_{switch}$	57.8	68.1	65.4	66.6	88.9	83.2	87.9	82.8	63.8	72.5	52.0	69.2	69.5	60.4	60.8	74.59	-0.14
M$_{ours}$	57.1	68.9	75.0	69.5	89.4	84.4	89.0	82.1	66.1	74.7	57.2	70.0	68.2	62.7	64.6	75.86	**+1.13**

Table 2: Experiments results for POS

Model	af	ar	bg	bn	de	el	en	es	et	eu	fa	fi	fr	he
mBERT	81.4	74.6	84.0	78.7	83.8	77.1	89.1	84.1	83.4	77.8	80.8	81.3	86.0	76.0
M$_{switch}$	80.6	74.4	81.8	82.0	83.1	74.3	89.2	77.1	81.9	76.3	77.2	81.3	86.2	75.1
M$_{ours}$	82.2	74.5	84.4	79.5	84.9	79.5	89.2	82.8	84.5	79.6	79.2	81.7	86.0	76.1
	hi	hu	id	it	ja	jv	ka	kk	ko	ml	mr	ms	my	nl
mBERT	80.2	80.6	85.0	87.4	43.5	73.7	77.9	69.4	72.4	71.9	75.0	72.7	67.0	85.6
M$_{switch}$	78.0	77.6	85.2	87.4	48.9	74.9	74.2	67.9	71.2	65.1	70.8	75.6	64.8	84.5
M$_{ours}$	79.3	81.7	85.3	87.7	46.5	75.9	76.9	74.3	72.8	71.2	76.2	76.7	68.0	85.7
	pt	ru	sw	ta	te	th	tl	tr	ur	vi	yo	zh	Avg	Diff
mBERT	86.0	76.9	79.3	75.1	71.1	20.6	74.4	78.3	73.6	85.6	59.6	64.2	75.62	-
M$_{switch}$	83.1	74.6	78.7	73.0	67.4	20.3	80.6	75.1	66.0	85.6	58.3	63.6	74.31	-1.31
M$_{ours}$	85.6	78.7	80.3	73.8	72.0	21.1	77.4	78.8	70.4	86.6	62.8	65.8	76.39	**+0.77**

Table 3: Experiments results for NER

Training Size (GB)	# Languages	Avg Gain
$[0.006, 0.354]$	13	+1.91
$[0.354, 1.414]$	10	+1.03
$[1.414, 5.657]$	8	+0.62

Table 4: Metric Breakdown on POS results

base configuration. The learning rate is set as $1e^{-5}$ for pretrained parameters and $9e^{-5}$ for newly initialized parameters. We use a batch-size of 32, warmup of 200 steps and patience of 3 on all experiments. We follow the zero-shot setting, in which we have only the English set for training and all languages for evaluation. We compare our method with the original mBERT and code-switching. The substitution ratio is set to 15% for main experiments. Seed numbers in all experiments keep the same so that the substituted tokens on both settings are the same for each run. To reduce the influence of the randomness caused by the token substitution, results are averaged by three runs.

3.2 Results

In Table 2 and 3, the code-switching negatively affects the mBERT by 0.14 on POS and 1.31 on NER, while our method makes improvement of 1.13 on POS and 0.77 on NER. Our method successfully eliminates the code-switching noise and demonstrates the effectiveness over 30+ languages.

3.3 Analysis

In this section, we perform the analysis on the POS task, with the default substitution ratio of 15%.

Metric Breakdown To determine how well our model performs on each language, we breakdown the evaluation scores based on how much pretraining data the mBERT used for each language. We follow the range of training size for each language as described in Wu and Dredze (2020), and average the performance gain on the POS task of our model compared to the original mBERT. As shown in Table 4, we notice our model has better performance gain especially on languages that are less trained in mBERT. For example, we have achieved 3.8 and 9.6 performance gain on yo and mr, respectively, for which the languages have less than 0.1 GB data in the mBERT training. It indicates that our alignment strategy enriches the representation of low-resource tokens, leading to a better performance on the low-resource languages.

Model	Switched	Original
Code-Switch	5.69%	4.05%
Ours	13.84%	2.07%

Table 5: Averaged percentage of the received gradient for switched and original English tokens.

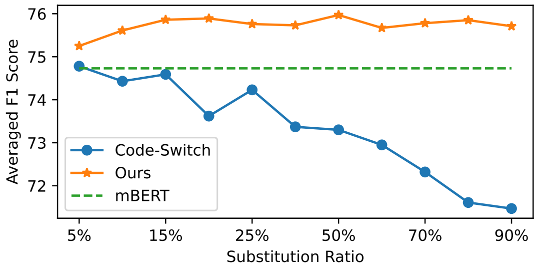

Figure 2: Evaluation of different substitution ratios. As the ratio increases, our method is not affected but the original code-switching sentences gets worse.

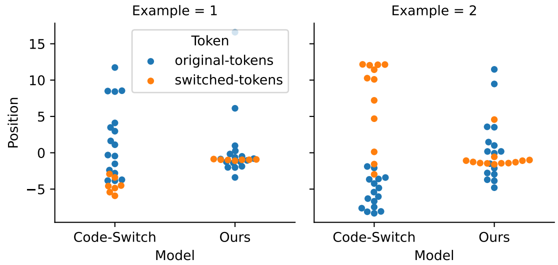

Figure 3: Token-Level density in training.

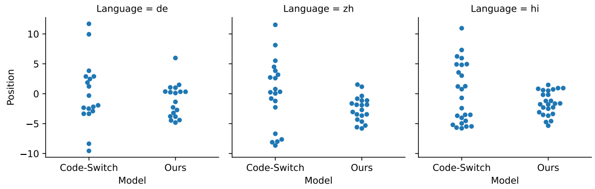

Figure 4: Token-Level density in inference.

Switching Effectiveness To evaluate whether our model actually benefits from the switched tokens, we adopt the gradient attribution test (Ancona et al., 2018). Specifically, we evaluate the importance of each token to the model's prediction by calculating the gradient for each test input. As in Table 5, we see the ratio of the received gradient for switched tokens in our method is much greater than original English tokens. Also, our model has shown greater relative importance of the switched tokens than the vanilla code-switching method. It indicates that such substituted tokens significantly contribute to our model's prediction and benefit the performance.

Substitution Strategy We compare the effect of different token substitution ratios for original code-switching and our method. As shown in Figure 2, our method has consistent greater scores on different substitution ratios from 5% to 90%. However, the performance of original code-switching method decreases as the substitution percentage increases. Such results show the stable performance of our method, in which the code-switched tokens keep benefiting the cross-lingual knowledge transfer.

Token-Level Coherence We plot the degree of dispersion between tokens in a sentence to further analyze whether our method keeps the consistent context space in both training and inference. Specifically, for both the switched and original tokens, we retrieve the token embedding from intermediate (6-th) layers and use the top feature calculated from the Principle Component Analysis (PCA). In Figure 3, our model has shown a more compact space for the sequence of tokens and the switched tokens are also inside the space of the context. However, the original code-switching method entirely separates the substituted tokens apart from the original context. Also, Figure 4 shows the effectiveness of our model on keeping a consistent context space during inference. Tokens in all three languages are very close in our method but separated apart in the original code-switched approach. Our model has demonstrated the effectiveness on keeping a consistent embedding space in training and inference.

4 Related Works

Previous studies(Huang et al., 2019; Liu et al., 2020a; Gritta and Iacobacci, 2021; Luo et al., 2021) trained language models on either monolingual or cross-lingual corpus to learn the multilingual representation. Recent works(Wu and Dredze, 2019; Hu et al., 2020; Conneau et al., 2020b) have proved the effective zero-shot transferable ability of multilingual models. Researchers(Zhang et al., 2019; Yang et al., 2020b,a; Qin et al., 2020; Liu et al., 2020b; Yang et al., 2021) tried to use code-switched sentences to enhance the representation among various languages and have been proved the success on many cross-lingual tasks. We believe our approach further addresses the limitation the code-switching on token-level classification.

5 Conclusion

This paper introduces an alignment strategy to map the code-switched tokens to original context and solves the grammatically incoherence in the embedding space of code-switching. Experimental results on POS and NER along with comprehensive analysis have demonstrated the effectiveness of our approach on the token-level classification. We think this work could further address other cross-lingual tasks and multilingual pretraining.

Acknowledgements

We thank all reviewers' valuable suggestions. We also appreciate the helpful feedback from Boyuan Zheng, and Shuai Zhang. The computing resources used by this work are supported by the Center for Language and Speech Processing (CLSP) at Johns Hopkins University (JHU).

Limitation

We do not use any neural-based aligner e.g the awesome-aligner(Dou and Neubig, 2021), because we want to make the aligning part simple and efficient. We believe that some modern methods or involving grammatical information could help achieve better aligning results but it is not the point of this paper. Although the construction of the dictionary is much less computationally expensive, it must be completed before the training and requires additional parallel data, which might cause inconsistency with the domain of the training text. The randomness introduced by the code-switching substitution may affect the overall performance, even though our method has considered the correlation between switched tokens and original context.

References

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2018. Towards better understanding of gradient-based attribution methods for deep neural networks. In *International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *EMNLP*.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. 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.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Milan Gritta and Ignacio Iacobacci. 2021. XeroAlign: Zero-shot cross-lingual transformer alignment. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 371–381, Online. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *ArXiv*, abs/2003.11080.

Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. 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 2485–2494, Hong Kong, China. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. In *NeurIPS*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020a. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020b. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. In *AAAI*.

Fuli Luo, Wei Wang, Jiahao Liu, Yijia Liu, Bin Bi, Songfang Huang, Fei Huang, and Luo Si. 2021. VECO: Variable and flexible cross-lingual pre-training for language understanding and generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3980–3994, Online. Association for Computational Linguistics.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. 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.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. 2020. Cosda-ml: Multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3853–3860. International Joint Conferences on Artificial Intelligence Organization. Main track.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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 833–844, Hong Kong, China. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Jian Yang, Shuming Ma, Dongdong Zhang, ShuangZhi Wu, Zhoujun Li, and Ming Zhou. 2020a. Alternating language modeling for cross-lingual pre-training. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9386–9393.

Jian Yang, Yuwei Yin, Shuming Ma, Haoyang Huang, Dongdong Zhang, Zhoujun Li, and Furu Wei. 2021. Multilingual agreement for multilingual neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 233–239, Online. Association for Computational Linguistics.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. 2020b. CSP:code-switching pre-training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2624–2636, Online. Association for Computational Linguistics.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2019. Cross-lingual dependency parsing using code-mixed TreeBank. 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 997–1006, Hong Kong, China. Association for Computational Linguistics.