Oolong: Investigating What Makes Crosslingual Transfer Hard with Controlled Studies

Zhengxuan Wu[*], **Isabel Papadimitriou**[*], **Alex Tamkin**[*†]
Stanford University
{wuzhengx, isabelvp, atamkin}@stanford.edu

Abstract

Little is known about what makes crosslingual transfer hard, since factors like tokenization, morphology, and syntax all change at once between languages. To disentangle the impact of these factors, we propose a set of *controlled transfer studies*: we systematically transform GLUE tasks to alter different factors one at a time, then measure the resulting drops in a pretrained model's downstream performance. In contrast to prior work suggesting little effect from syntax on knowledge transfer, we find significant impacts from syntactic shifts (3-6% drop), though models quickly adapt with continued pretraining on a small dataset. However, we find that by far the most impactful factor for crosslingual transfer is the challenge of aligning the new embeddings with the existing transformer layers (18% drop), with little additional effect from switching tokenizers (<2% drop) or word morphologies (<2% drop). Moreover, continued pretraining with a small dataset is not very effective at closing this gap—suggesting that new directions are needed for solving this problem.

1 Introduction

What makes it hard for neural networks to learn new languages? Despite their strengths, large-scale pretrained language models (LLMs) require very large datasets for pretraining, making it challenging to train LLMs from scratch for low-resource languages (Devlin et al., 2018; Liu et al., 2019; Lacoste et al., 2019; Clark et al., 2020). For such languages, an appealing approach is to *transfer* knowledge from an LLM trained for a high-resource language, since such models may transfer knowledge even under extreme conditions (Papadimitriou and Jurafsky, 2020; Tamkin et al., 2020). A range of training strategies and model architectures have been explored to enable such crosslingual transfer of English LLMs, leveraging different techniques including adaptive pretraining (Reimers and Gurevych, 2020), and embedding re-training (Artetxe et al., 2020; Tran, 2020). However, progress is challenging as it is hard to distinguish factors that effect such transfer.

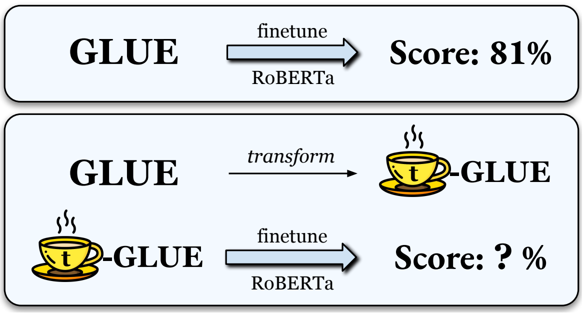

Figure 1: **Controlled transfer studies paradigm.** We systematically transform GLUE tasks (t-GLUE) to target one linguistic factor, then finetune a pretrained RoBERTa model on that dataset. The resulting drop in performance indicates the importance of that factor to crosslingual transfer. See Table 1 for the list of transformations.

In this paper, we present a set of *controlled transfer studies* for determining what makes crosslingual transfer hard. We focus on three factors salient to crosslingual transfer: the **embedding layer**, the **tokenizer**, and **syntactic shifts**. We construct a set of systematically transformed versions of GLUE (t-GLUEs) targeting each of these factors, and observe how the finetuning performance of a pre-trained English `RoBERTa` (Liu et al., 2019) model degrades as a result of each of these transformations. Crucially, our method allows us to disentangle the effects of correlated factors: while all factors would change at once if we transfer between natural languages, our transformations allow us to pinpoint what causes difficulties for transfer.

First, we focus on vocabulary effects: what is

In Chinese, “Oolong” can refer to an unexpected change or development. [*]Equal contribution. [†]Corresponding author.

it that makes the embeddings important in crosslingual transfer? We separately test the effects of having a consistent word embeddings layer, the morphological consistency of the tokenizer, and the all-around quality of tokenization. Surprisingly, we find that the effect of word embedding alignment overshadows any aspect of tokenizer quality that we test. Second, we test how all of these vocabulary-level factors compare to the effects of structural syntactic perturbations, and find that structural perturbations have a significantly smaller effect on transfer ability than vocabulary effects.

2 Related Work

Cross-lingual transfer studies on multi-lingual models such as Multilingual BERT (Devlin et al., 2018) demonstrate the utility of multilingual training for producing models with parallel representations that ease zero-shot transfer (Pires et al., 2019). However, it is unlikely that a unified model can cover the more than 6,900 languages in the world (Feng et al., 2020; Wang et al., 2020). Enabling crosslingual transfer from one language to another remains an important tool for expanding large-scale models to more languages, and we aim to examine such cross-lingual transfer with well-defined experiments covering distinct linguistic factors.

Our experiments build off previous efforts that try to enable crosslingual transfer from pretrained monolingual LLMs to new languages (Artetxe et al., 2018, 2020; Tran, 2020; Reimers and Gurevych, 2020; Gogoulou et al., 2021). For example, BERT shows effective but limited transfer when attached to multi-lingual tokenizers (Reimers and Gurevych, 2020). Recent work further improves this approach through continued multistage pretraining on foreign languages, which helps the English LLMs acquire new word embeddings (Tran, 2020). In this work, we aim to provide quantitative insights into how these factors, such as static embeddings similarity, affect crosslingual transfer through controlled studies.

3 Transformed English (t-Englishes)

Figure 1 shows our main experimental paradigm. We study cross-lingual knowledge transfer from English to t-Englishes: versions of English that are systematically altered through different types of transformations. Here we explain and motivate each t-English variant that we study, organized by the main component they consider in crosslingual transfer: the embedding layer, the tokenizer, and syntactic shifts. A full list with examples can be found in Table 1.

3.1 Embedding Layer

Previous works have consistently found that good embeddings are crucial for enabling effective crosslingual transfer (Tran, 2020; Artetxe et al., 2020). However, these gains could be due to several factors, including better initialization statistics (Raghu et al., 2019), or to a learned alignment between the learned embeddings and the pretraiend transformer layers (Wu et al., 2021). We consider a baseline setting, where we reinitalize the embedding layer such that the optimization during fine-tuning has to fully reconstruct the semantic space, and compare it against two other perturbations which preserve the statistics of the embedding weights: 1) a **token swap** scenario, where we scramble the identities of tokens so that the meaning of each token is not represented by its row in the word embeddings matrix, and 2) a more challenging **word swap** scenario where the identities of whole words are scrambled before tokenization. In the word swap scenario, the morphological and sub-word consistency of the tokenizer is lost: if the word “cat” gets swapped to “audible”, and the word “bravery” gets swapped to “visible”, then after tokenization the “-ible” subword does not represent a consistent morpheme.

3.2 Tokenizer

Tokenizers have been shown to play an important role in crosslingual transfer for multi-lingual models (Rust et al., 2020). In this section, we investigate how changes in tokenization between the source and target languages impacts downstream performance, holding all other factors constant. To do so, we substitute the Byte-Pair Encoding (BPE) (Sennrich et al., 2015) of the `RoBERTa` model with the WordPiece tokenizer (Wu et al., 2016) used by `BERT` (Devlin et al., 2018) and the SentencePiece tokenizer (Kudo and Richardson, 2018) used by `Albert` (Lan et al., 2019). We also experiment with two non-English tokenizers, which produce lower-quality tokenizations for English: (see Appendix A.1): the French `FlauBERT` (Le et al., 2020) and the Dutch `DutchBERT` (de Vries et al., 2019). Examples of each tokenization scheme are presented in Table 1. For new tokenizers, word embedding weights are reinitialized accordingly.

Transformation Type	**Sentence / Sequence**
Original English	*"the film unfolds with all the mounting tension of an expert thriller , until the tragedy beneath it all gradually reveals itself ."*
♠RoBERTa Tokenizer	*"the film unfolds with all the mounting tension of an expert thriller , until the tragedy beneath it all gradually reveals itself ."*
♠BERT Tokenizer	*"the film un fold s with all the mounting tension of an expert thriller , until the tragedy beneath it all gradually reveals itself ."*
♠Albert Tokenizer	*"the film unfold s with all the mounting tension of an expert thriller , until the tragedy beneath it all gradually reveals itself ."*
♠FlauBERT Tokenizer	*"the film un fol ds with all the mou n ting tension of an expert thriller , un til the tr age dy bene ath it all gradu ally re ve als it self ."*
♠DutchBERT Tokenizer	*"the film u n f old s with all the mo unt ing te n sion of a n expert thriller , u n til the trage d y ben e ath i t all gra d u ally rev e als i t sel f ."*
♠Token Swap	*"intimacy Turbo unction Prime discredited sometimes pora extraordinarily UD Adventure stall arger humming illy sometimes distinction brook gruesome discredited atel Flag Stones wait Also"*
Word Swap	*"objectivist 13th robespierre inchmickery tang objectivist ramu bobadilla legione plaaf injures excavation kianja 461 objectivist cyanophilous gringotts clangers tang cautleyi peddie coromandel patria"*
$\{\mathrm{N_{fr}}, \mathrm{V_{fr}}\}$	*"the film with all the of an expert , until the beneath all gradually . itself reveals it tragedy thriller tension mounting unfolds"*
$\{\mathrm{N_{ja}}, \mathrm{V_{ja}}\}$	*"the film unfolds with all the tension of an thriller , until the tragedy beneath it all gradually itself . reveals expert mounting"*
$\{\mathrm{N_{fr}}, \mathrm{V_{ja}}\}$	*"the film unfolds with all the of an expert , until the beneath all gradually . itself reveals it tragedy thriller tension mounting"*
Random Order	*"an all all gradually beneath thriller with reveals . until tension tragedy mounting the it of the the expert , unfolds itself film"*
Reverse Order	*". itself reveals gradually all it beneath tragedy the until , thriller expert an of tension mounting the all with unfolds film the"*

Table 1: An example from the SST-2 dataset and its transformed variants. ♠ Instead of the original English sentence, we show the tokenized sequence. Special pre-fixes and post-fixes such as $\dot{G}$, $\#\#$, _ and $\langle/w\rangle$ are ignored in a tokenized sequence for simplicity.

3.3 Syntactic Shifts

While syntax is an crucial aspect of language, studies have also shown syntactic typology to be surprisingly non-predictive of transfer quality (Pham et al., 2021), and other studies have shown LLMs to be largely word-order invariant (Sinha et al., 2021). We investigate a set of syntactic transformations that isolate syntactic word-order shifts from the other factors that can vary between languages such as tokenization, static embeddings, and morphological representation. We use the open-source package Galactic-Dependencies (Wang and Eisner, 2016) to transform the word order of our English training corpora to match the noun-phrase order and the verb-phrase order of French and Japanese ($\{\mathbf{N}_{\mathrm{fr}}, \mathbf{V}_{\mathrm{fr}}\}$ and $\{\mathbf{N}_{\mathrm{ja}}, \mathbf{V}_{\mathrm{ja}}\}$ in Table 1) and also perform a mixed transformation with French noun order and Japanese verb order ($\{\mathbf{N}_{\mathrm{fr}}, \mathbf{V}_{\mathrm{ja}}\}$ in Table 1). We use the open-source package Stanza (Qi et al., 2020) for sentence segmentation and parsing before scrambling word orders. We also test the the stronger transformation of completely reversing the word order, as well as randomly shuffling the word order to produce a lower bound on the amount of useful syntactic information.

4 When Does Continued Pretraining Help?

Instead of finetuning LLMs directly on target language tasks, continued pretraining on a target language corpus may help with model adaptation (Artetxe et al., 2020; Tran, 2020). Here, we examine how continued pretraining fares as a potential solution to the different shifts we consider in Section 3; this corresponds to the setting where there might exist a moderate amount of unlabeled data available for a low-resource language. Formally, we continued pretraining RoBERTa using the masked language modeling objective on a t-English corpus before evaluating its finetuning performance with downstream tasks. We use a subset of WikiText-103M corpus (Merity et al., 2016) containing approximately 15% of examples for mid-tuning.[1] For each t-English variant, we ensure the number of steps for continued pretraining are kept constant.

5 Results

We use the GLUE benchmark (Wang et al., 2018) to evaluate model performance, which consists of nine different NLP tasks. We produce t-GLUEs by transforming both the training and validation sets, and report scores on the transformed validation sets after finetuning our pre-trained or continued pretrained models. Each experiment is run three times with different random seeds. We include details in Appendix A.2.

5.1 Good Embeddings Are Most of What You Need

As expected, we find a very large drop in GLUE performance (-40%) from reinitializing the em-

[1] For comparison, the pretraining data for $\mathrm{BERT_{BASE}}$ contains 3.3B tokens (Devlin et al., 2018). Here, we have about 15M tokens which is about 0.45% of its pretraining data.

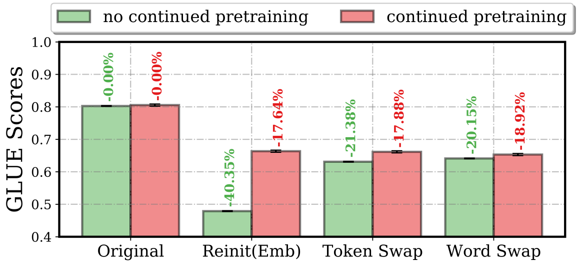

Figure 2: Averaged GLUE scores for t-Englishes with scrambled word identities. The scores with original English sentences are included for comparison. Error bar shows standard deviation across 3 distinct runs with different random seeds.

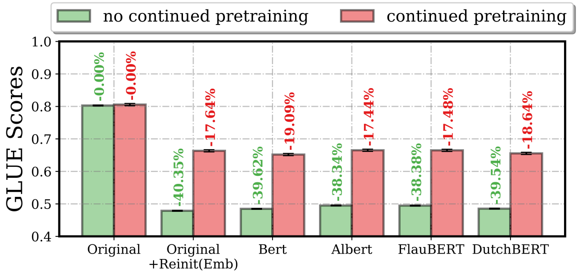

Figure 3: Averaged GLUE scores for t-Englishes with tokenizer substitutions. The scores with original English sentences are included for comparison. Error bar shows standard deviation across 3 distinct runs with different random seeds.

beddings layer (Figure 2), indicating the importance of good embeddings to crosslingual transfer. However, the token swapping results reveal that only half of that drop (-21%) is attributable to the token-specific information stored in each embedding, with the remainder due to merely having embeddings that lie within a plausible "language space" for the model. Surprisingly, scrambling the whole words rather than tokens and breaking the morphological consistency of the resulting tokenization does not have a larger effect than the one-to-one token swap, suggesting minimal additional contribution of subword information in this regime.

Interestingly, we find that continued pretraining closes the gap between the reinitialized and swapped conditions. While this eliminates over half of the performance drop of for the reinitialized t-English (-40% to -18%) the gain is much smaller for the swapped varaints (-21% to -18%). This suggests that other solutions to the word alignment problem besides continued pretraining will be needed in resource-scarce settings.

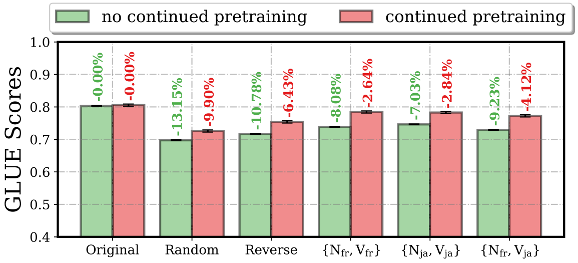

Figure 4: Averaged GLUE scores for t-Englishes with syntactic shifts. The scores with original English sentences are included for comparison. Error bar shows standard deviation across 3 distinct runs with different random seeds.

5.2 Bad Embeddings Can Ruin A Good Tokenizer

Surprisingly, we see virtually no effect of using different different tokenizers (Figure 3), despite some of the tokenizers being very different from the original BPE tokenizer. Similarly, continued pretraining also fails to reveal differences when different tokenizers are used. These findings suggest that performance drops seen during crosslingual transfer are mostly due to the information loss of reinitializing the embedding weights, rather than the quality of tokenization. Tokenization may thus be a "lower-order bit" for crosslingual transfer, which has little impact until good word embeddings are learned.

5.3 Syntax Matters, But Not Too Much

As shown in Figure 4, a `RoBERTa` model finetuned on randomly ordered GLUE sentences experiences a performance drop of 13%. This is a markedly smaller drop than the word-shuffling experiments (-19%), indicating that a bag-of-words classifier with good embeddings performs better than a model which experiences no syntactic shifts but that lacks good word embeddings.

This performance gap closes appreciably as we perform more structured syntactic shifts such as reversing the sentence (a drop of 10%), or systematically permuting word orders using the dependency tree (a drop of between 7% and 9%). Rather than being invariant to word orders across natural language understanding tasks (Sinha et al., 2021; Pham et al., 2021), we instead find that BERT-based models are in fact sensitive to word order, at least for the tasks in the GLUE benchmark. In addition, we find that continued pretraining can close the performance gap to all but a few percentage points for tree-based structural shifts. These results suggest that syntactic shifts have real but limited impact on crosslingual transfer, compared to embedding layer effects.

6 Conclusions

In this paper, we propose a novel paradigm to study cross-lingual transfer through transformations which simulate the linguistic changes across languages. In contrast to what prior work implies, we find significant effects from syntactic shifts and no effect from tokenizer shifts on cross-lingual transfer. However, our results suggest that solving the embedding alignment problem is the "high-order bit" for crosslingual transfer: it has the largest impact on finetuning performance and is the least improved by continued pretraining. Thus, future progress on solving this problem in large-scale transformers may have outsized impact.

7 Acknowledgements

We would like to thank Nelson Liu, Mirac Suzgun and Tolúlọpẹ́ Ògúnrẹ̀mí for useful discussions and comments on drafts.

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-second AAAI conference on artificial intelligence*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT Model.

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

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Evangelia Gogoulou, Ariel Ekgren, Tim Isbister, and Magnus Sahlgren. 2021. Cross-lingual transfer of monolingual models. *arXiv preprint arXiv:2109.07348*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. Flaubert: Unsupervised language model pre-training for french. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. ROBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Isabel Papadimitriou and Dan Jurafsky. 2020. Learning music helps you read: Using transfer to study linguistic structure in language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6829–6839.

Thang Pham, Trung Bui, Long Mai, and Anh Nguyen. 2021. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1145–1160, Online. 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*, pages 4996–5001.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. 2019. Transfusion: Understanding transfer learning for medical imaging. *Advances in neural information processing systems*, 32.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2020. How good is your tokenizer? on the monolingual performance of multilingual language models. *arXiv preprint arXiv:2012.15613*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *arXiv preprint arXiv:2104.06644*.

Alex Tamkin, Trisha Singh, Davide Giovanardi, and Noah Goodman. 2020. Investigating transferability in pretrained language models. *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Ke Tran. 2020. From english to foreign languages: Transferring pre-trained language models. *arXiv preprint arXiv:2002.07306*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Dingquan Wang and Jason Eisner. 2016. The galactic dependencies treebanks: Getting more data by synthesizing new languages. *Transactions of the Association for Computational Linguistics*, 4:491–505.

Zirui Wang, Zachary C Lipton, and Yulia Tsvetkov. 2020. On negative interference in multilingual language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhengxuan Wu, Nelson F Liu, and Christopher Potts. 2021. Identifying the limits of cross-domain knowledge transfer for pretrained models. *arXiv preprint arXiv:2104.08410*.

A Appendix

A.1 Sequence Length Distribution

As described in Section 3.2, we try four different tokenizers to substitute for our RoBERTa (Liu et al., 2019) model that uses the Byte-Pair Encoding (BPE) (Sennrich et al., 2015) tokenizer. Specifically, we substitue with the WordPiece tokenizer (Wu et al., 2016) used by BERT (Devlin et al., 2018) (i.e., BERT Tokenizer in Table 1) and the SentencePiece tokenizer (Kudo and Richardson, 2018) used by Albert (Lan et al., 2019) (i.e., **Albert Tokenizer** in Table 1). Additionally, we substitute with two new non-English tokenizers including the French FlauBERT (Le et al., 2020) (**FlauBERT Tokenizer** in Table 1) and the Dutch DutchBERT (de Vries et al., 2019) (DutchBERT Tokenizer in Table 1). As shown in Figure 5, we plot the distributions of sequence lengths as a measure of the heterogeneity introduced by new tokenizers to ensure variences across tokenized sequence lengths. Specifically, we see there are inferior tokenizers such as FlauBERT Tokenizer with a 22.15% increase in sequence length. Our results are consistent with previous findings (Rust et al., 2020) where sequence length distributions are closer

A.2 Training Set-up

Downstream Task. We use the GLUE benchmark (Wang et al., 2018) to evaluate model performance, which covers nine different NLP tasks. We report scores on the development sets for each task by fine-tuning our pre-trained or mid-tuned models. We fine-tune for 5 epochs for the smaller datasets (WNLI and MRPC) and 3 epochs for the others. For the performance metrics, we use Matthew's Correlation for CoLA, Pearson correlation for STS-B, and accuracy for all the other datasets.

Hyperparameter and Infrastructure. For each of the mid-tuning and fine-tuning experiments, we collect averaged results from 3 runs with distinct random seeds. We tune our models with two learning rates $\{2e^{-5}, 4e^{-5}\}$, and report the best results from these two learning rates. Fine-tuning with 9 GLUE tasks takes about 8 hours on 4 NVIDIA Titan 12G GPUs. Mid-tuning with our subset of

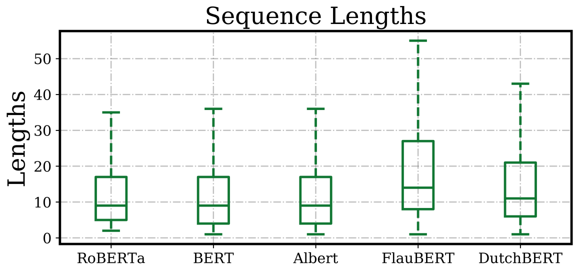

Figure 5: Distributions of sequence lengths by different tokenizers.

WikiText-103M corpus takes about 18 hours with the same infrastructure.

A.3 GLUE Task Performance

Table 2 shows performance break-down for individual GLUE task under different transformations as described in Section 3.

	Original	**Token Swap**	**Word Swap**	**Reinit(Emb)**	`Bert`	`Albert`	`FlauBERT`	`DutchBERT`	**Random**	**Reverse**	$\{\mathbf{N_{fr}}, \mathbf{V_{fr}}\}$	$\{\mathbf{N_{ja}}, \mathbf{V_{ja}}\}$	$\{\mathbf{N_{fr}}, \mathbf{V_{ja}}\}$
CoLA	.58(.01)	.00(.00)	.00(.00)	.00(.00)	.00(.00)	.00(.00)	.00(.00)	.00(.00)	.04(.05)	.01(.01)	.16(.01)	.21(.01)	.12(.01)
CoLA$_{\mathbf{c.p.}}$	.59(.01)	.05(.07)	.02(.02)	.06(.05)	.00(.00)	.00(.00)	.01(.01)	.00(.00)	.22(.04)	.35(.01)	.45(.03)	.47(.01)	.44(.01)
MNLI	.88(.00)	.34(.01)	.50(.08)	.53(.03)	.54(.01)	.53(.01)	.67(.01)	.68(.00)	.82(.00)	.85(.00)	.86(.00)	.86(.00)	.85(.00)
MNLI$_{\mathbf{c.p.}}$	.88(.00)	.72(.01)	.72(.01)	.73(.00)	.73(.01)	.71(.00)	.71(.01)	.69(.00)	.82(.00)	.86(.00)	.86(.00)	.86(.00)	.86(.00)
MRPC	.88(.01)	.68(.00)	.68(.00)	.68(.00)	.68(.00)	.68(.00)	.76(.01)	.77(.01)	.77(.01)	.85(.02)	.85(.01)	.86(.01)	.83(.00)
MRPC$_{\mathbf{c.p.}}$	.87(.00)	.83(.00)	.80(.04)	.79(.01)	.82(.01)	.80(.01)	.83(.01)	.78(.01)	.81(.01)	.87(.01)	.87(.01)	.87(.01)	.86(.00)
QNLI	.93(.00)	.60(.01)	.54(.02)	.54(.04)	.55(.03)	.52(.01)	.79(.01)	.79(.00)	.88(.00)	.89(.00)	.90(.00)	.91(.00)	.90(.00)
QNLI$_{\mathbf{c.p.}}$	.93(.00)	.83(.01)	.82(.01)	.82(.00)	.83(.00)	.82(.00)	.82(.00)	.81(.00)	.88(.00)	.91(.00)	.91(.00)	.92(.00)	.91(.00)
QQP	.91(.00)	.77(.00)	.77(.00)	.77(.00)	.76(.00)	.75(.00)	.85(.00)	.86(.00)	.90(.00)	.91(.00)	.90(.00)	.91(.00)	.90(.00)
QQP$_{\mathbf{c.p.}}$	.91(.00)	.87(.00)	.87(.00)	.87(.00)	.87(.00)	.87(.00)	.86(.00)	.87(.00)	.90(.00)	.91(.00)	.91(.00)	.91(.00)	.91(.00)
RTE	.65(.02)	.51(.03)	.51(.03)	.53(.00)	.53(.00)	.53(.01)	.54(.02)	.56(.02)	.57(.01)	.60(.02)	.60(.00)	.61(.01)	.59(.05)
RTE$_{\mathbf{c.p.}}$	.67(.01)	.56(.01)	.53(.01)	.54(.03)	.57(.01)	.59(.02)	.57(.03)	.57(.02)	.59(.02)	.58(.02)	.69(.01)	.64(.05)	.65(.03)
SST-2	.94(.00)	.79(.01)	.75(.02)	.79(.03)	.73(.04)	.68(.05)	.77(.01)	.78(.00)	.86(.01)	.91(.00)	.92(.00)	.92(.00)	.92(.00)
SST-2$_{\mathbf{c.p.}}$	.94(.00)	.83(.01)	.85(.01)	.85(.01)	.83(.00)	.82(.00)	.82(.01)	.81(.01)	.88(.00)	.93(.00)	.93(.00)	.93(.00)	.92(.00)
STS-B	.89(.00)	.06(.01)	.06(.00)	.06(.02)	.09(.02)	.08(.02)	.74(.01)	.77(.00)	.87(.00)	.87(.00)	.88(.00)	.88(.00)	.88(.00)
STS-B$_{\mathbf{c.p.}}$	.89(.00)	.76(.01)	.73(.03)	.77(.01)	.79(.01)	.78(.00)	.77(.00)	.79(.00)	.88(.00)	.87(.00)	.89(.00)	.89(.00)	.89(.00)
WNLI	.56(.00)	.56(.00)	.56(.00)	.56(.00)	.56(.00)	.58(.03)	.56(.00)	.56(.01)	.55(.01)	.56(.01)	.56(.00)	.56(.00)	.56(.01)
WNLI$_{\mathbf{c.p.}}$	.56(.01)	.52(.06)	.53(.05)	.53(.03)	.55(.02)	.51(.07)	.56(.00)	.56(.00)	.55(.01)	.51(.07)	.56(.01)	.56(.00)	.53(.05)

Table 2: GLUE scores for t-English with different types of interventions including scrambled word identities, syntactic shifts, and tokenizer substitutions with standard deviation (SD) for all tasks across 3 distinct runs with different random seeds. The scores with original English sentences are included for comparison. **c.p.** indicates finetuning results with continued pretrained models.