Embedding Structure Matters: Comparing Methods to Adapt Multilingual Vocabularies to New Languages

C.M. Downey[α] **Terra Blevins**[β] **Nora Goldfine**[α] **Shane Steinert-Threlkeld**[α]
[α]Department of Linguistics, University of Washington
[β]Paul G. Allen School of Computer Science & Engineering, University of Washington
{cmdowney,shanest}@uw.edu
blvns@cs.washington.edu
ngoldfine@gmail.com

Abstract

Pre-trained multilingual language models underpin a large portion of modern NLP tools outside of English. A strong baseline for specializing these models for specific languages is Language-Adaptive Pre-Training (LAPT). However, retaining a large cross-lingual vocabulary and embedding matrix comes at considerable excess computational cost during adaptation. In this study, we propose several simple techniques to replace a cross-lingual vocabulary with a compact, language-specific one. Namely, we address strategies for re-initializing the token embedding matrix after vocabulary specialization. We then provide a systematic experimental comparison of our techniques, in addition to the recently-proposed FOCUS method. We demonstrate that: 1) Embedding-replacement techniques in the monolingual transfer literature are inadequate for adapting multilingual models. 2) Replacing cross-lingual vocabularies with smaller specialized ones provides an efficient method to improve performance in low-resource languages. 3) Simple embedding re-initialization techniques based on script-wise sub-distributions rival techniques such as FOCUS, which rely on similarity scores obtained from an auxiliary model.

1 Introduction

For languages other than English and a handful of other very high-resource languages, pre-trained multilingual language models form the backbone of most current NLP systems. These models address the relative data scarcity in most non-English languages by pooling text data across many languages to train a single model that (in theory) covers all training languages (Devlin, 2019; Conneau and Lample, 2019; Conneau et al., 2020; Liu et al., 2020; Scao et al., 2023, i.a.). These models often include language-agnostic tokenization and an increased vocabulary capacity over monolingual models (Conneau et al., 2020).

However, Wu and Dredze (2020) show that these massively multilingual models still underperform on lower-resource languages. Recent efforts to cover these languages instead pre-train models that are specialized to specific languages or language families (Ogueji et al., 2021; Ogunremi et al., 2023). These approaches nonetheless require training a new model from scratch and do not leverage transferable information in existing models.

Our study builds on a line of work which instead *adapts* a pre-trained cross-lingual model (such as XLM-R; Conneau et al., 2020) to a single language, or a smaller set of languages. Language-Adaptive Pre-Training (LAPT)—continuing the MLM or CLM pre-training task on only the target language(s)—is a simple and strong baseline in this regard (Chau et al., 2020).

However, LAPT with no change to the cross-lingual vocabulary comes with considerable excess computational cost: when adapting to a single language or small subset of languages, only a small fraction of the cross-lingual vocabulary is used. The excess vocabulary still contributes to the computational cost on both the forward and backward pass, and embedding/output matrices often constitute a large fraction of the total trainable model parameters (for XLM-R-base, 192M / 278M ≈ 69% of parameters). Additionally, the information-theoretic tokenization modules for cross-lingual models are usually under-optimized for any given language, and especially low-resource languages (Ács, 2019; Conneau and Lample, 2019, i.a.)

For this reason, we propose several simple techniques to replace the large cross-lingual vocabulary of a pre-trained model with a compact, language-specific one during model specialization. Training a new SentencePiece or BPE tokenizer poses no special difficulties. However, re-initializing the embedding matrix for a new vocabulary, which will almost certainly introduce many new tokens lacking pre-trained embeddings, poses significant

Proceedings of the The 3rd Workshop on Multi-lingual Representation Learning (MRL), pages 268–281

challenges. We compare several methods for such embedding re-initialization.

After reviewing related literature in Section 2, we conduct a qualitative exploration of the pre-trained embedding space for a standard multilingual model: XLM-R (Section 3.1). This exploration informs our formalization of simple techniques to align new vocabulary embeddings with the pre-trained embedding distribution of our base model (Section 3.2). We then provide a systematic experimental comparison of the embedding re-initialization techniques we propose, plus the recently proposed FOCUS re-initialization method (Dobler and de Melo, 2023, Section 4). Our experiments cover a wide selection of low- and mid-resource target languages (i.e. those that have the most to gain from language specialization).[1]

The results of our experiments (Sections 5, 6) demonstrate the following: 1) Embedding-replacement techniques proposed in the monolingual model adaptation literature are inadequate for adapting multilingual models. 2) Replacing large cross-lingual vocabularies with smaller language-specific ones provides a computationally-efficient method to improve task performance in low-resource languages. 3) The simple re-initialization techniques we propose here, based on script-wise embedding sub-distributions, rival techniques such as FOCUS, which rely on model-driven semantic similarity.

2 Related Work

Pre-trained Model Adaptation Extensive work has proposed re-using and modifying pre-trained models for new settings in order to retain existing model knowledge and reduce pre-training costs. Gururangan et al. (2020) show that continued training on domain-specific data effectively adapts pre-trained models to new domains in both high- and low-resource settings. This approach is also used to adapt models to new languages (i.e. Language-Adaptive Pre-Training / LAPT; Chau et al., 2020).

Other approaches involve training new, language-specific adapter layers to augment a frozen monolingual (Artetxe et al., 2020) or multilingual encoder (Pfeiffer et al., 2020; Üstün et al., 2020; Faisal and Anastasopoulos, 2022). A comparison of these cross-lingual adaptation approaches (Ebrahimi and Kann, 2021) found that continued pre-training often outperforms more complex setups, even in low-resource settings. With this in mind, our experiments evaluate the success of models tuned for target languages with LAPT, starting from variable initializations depending on a choice of embedding adaptation technique.

Cross-lingual Vocabulary Adaptation A major limitation in adapting pre-trained models to new languages is the subword vocabulary, which often fails to cover an unseen script (Pfeiffer et al., 2021) or tokenizes target text inefficiently (Ács, 2019). Muller et al. (2021) demonstrate that script is an extremely important factor in predicting transfer success. Specifically, the pre-trained coverage of closely-related languages improves transfer, but only if the target language is written in the same script as its pre-trained relative.

One adaptation technique is to initialize new subword embeddings that cover the target language, e.g. by expanding the existing vocabulary with new tokens as necessary, then training the new (randomly initialized) embeddings (Chau et al., 2020; Wang et al., 2020). When transferring a monolingual model to a new language, Artetxe et al. (2020) and de Vries and Nissim (2021) instead completely re-initialize the embedding matrix, corresponding to a new subword vocabulary. These embeddings are then trained into alignment with the pre-trained, frozen transformer encoder. We show that this technique is not successful when adapting a multilingual model (Section 5).

Other work reuses information in pre-trained embeddings rather than initializing new ones at random. This may include scaling up smaller embedding spaces from models trained on the target language (de Vries and Nissim, 2021; Ostendorff and Rehm, 2023) or copying embeddings from the original vocabulary where there is exact vocabulary overlap (Pfeiffer et al., 2021). When transferring to a target language written in a poorly-covered script, Muller et al. (2021) show that transliterating the target to the script of a well-covered relative can lead to significant performance gains.

Finally, recent work has proposed more complex methods for mapping source embeddings onto semantically similar ones in the target space either through cross-lingually aligned static word embeddings (e.g. the WESCHEL method; Minixhofer et al., 2022) or with bilingual lexicons (Zeng et al., 2023). In concurrent work to ours, Dobler and de Melo (2023) extend WECHSEL with the FO-

[1]The software used to run all experiments may be found at https://github.com/cmdowney88/EmbeddingStructure

CUS method to specialize multilingual vocabularies to a single language. Ostendorff and Rehm (2023) use a cross-lingual progressive transfer learning approach to combine information from the source embeddings and a smaller target language model to initialize higher-dimension target embeddings. Unlike earlier initialization methods and our proposed setup, these methods all require additional information outside the source model and often require significant additional compute. We compare one method from this family (FOCUS) to our proposed heuristic-based initialization schemes.

3 Vocabulary Replacement & Embedding Re-initialization

Research transferring monolingual models from one language to another (e.g. Artetxe et al., 2020; de Vries and Nissim, 2021), has shown that random re-initialization of embeddings +LAPT is sufficient. However, our experiments show that this technique performs poorly when transferring from a multilingual model (Section 5). For this reason, we propose several simple techniques for initializing new embeddings based on a qualitative exploration of the embedding space for XLM-R (Section 3.1), and include the more complex FOCUS technique, developed concurrently with our work, for comparison (Dobler and de Melo, 2023).

3.1 XLM-R Embedding-Space Analysis

To better understand the task of initializing new embeddings for a multilingual model, we explore the token-embedding space of XLM-R through PCA projection. Our hypothesis is that multilingual models do not process all languages homogeneously. This seems to be demonstrated in Figures 1a and 1b, where word embeddings are colored by their respective Unicode script block. We see that the highest-resource scripts in XLM-R (Common, Latin, and Cyrillic) have relatively divergent distributions, while others cluster closer together. This heterogeneity may help explain the finding from Muller et al. (2021) that pre-trained models do not transfer well to even closely-related target languages if the target script does not match that of the pre-trained relative.

Secondly, each script can be further divided into two sub-distributions, roughly corresponding to a shift in the second principal component. Figure 1c shows that this division corresponds to whether a token is word-initial or word-medial. To preserve whitespace information, SentencePiece tokens include a leading underscore to indicate tokens that should be preceded by a space (word-initial tokens).[2] Although the model does not have access to the internal makeup of its tokens, we hypothesize that it learns to discern which tokens can begin a word and which cannot.

Thus when proposing methods to initialize new embeddings for XLM-R, we hypothesize that initializing according script- and position-wise sub-distributions will help to align new vocabulary items with the pre-trained embedding distribution.

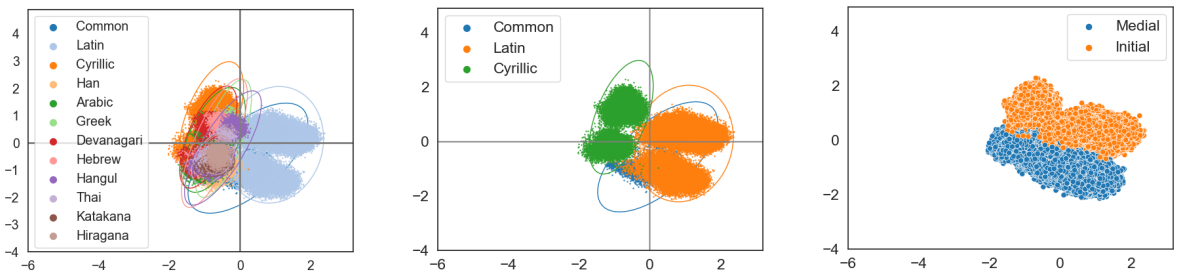
3.2 Embedding Re-initialization Techniques

We now formalize simple techniques for embedding re-initialization based on our exploration of XLM-R's embedding space, as well as one recently proposed technique based on an auxiliary embedding model (FOCUS). Figure 2 provides PCA visualizations of the re-initialized embeddings from each technique on a subword vocabulary specialized for languages of the Uralic family (we experiment with these languages in Section 4). The visualization for these languages' respective scripts (Common, Latin, Cyrillic) in the base model can be found in Figure 1b for comparison.

Re-initialization by Identity REINIT-IDENT first identifies tokens in the new vocabulary that exactly match a token in the original vocabulary, then sets the new embeddings of shared tokens to be identical to those in the original embedding table (Figure 2a). This is a common approach to preserve information from the original model, even when the other embeddings are randomly re-initialized (e.g., Pfeiffer et al., 2021). When identity re-initialization is applied in conjunction with another technique (such as REINIT-SCRIPT), identity takes precedence.

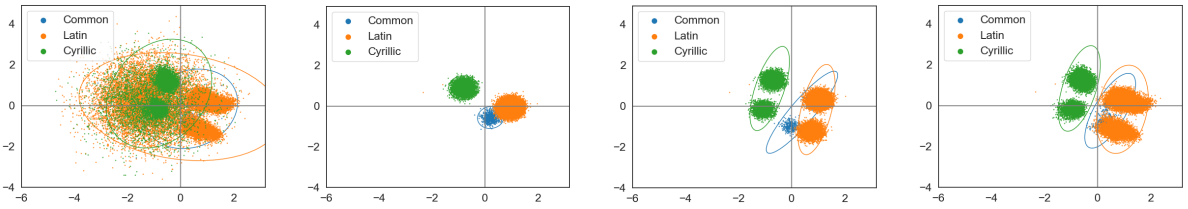
Re-initialization by Script For REINIT-SCRIPT, all base XLM-R tokens are first categorized by Unicode block, as a stand-in for identifying the script/orthography. We then calculate the mean and standard deviation for each script in the original embedding space. Finally, new token embeddings for each script are distributed according to a Normal distribution with the corresponding mean and standard deviation (Figure 2b).

[2] E.g., "_the" and "the" are word-initial and word-medial tokens of the same character sequence.



(a) (b) (c)

Figure 1: PCA visualizations of the embedding space for XLM-R. Subplots: (a) Distribution of embeddings for the 12 most common Unicode scripts. (b) Plot reduced to only Common, Latin, and Cyrillic scripts for simplicity. (c) Embeddings colored by whether the token begins a word (initial) or occurs in the middle of one (medial)



(a) IDENT (b) SCRIPT (c) SCRIPT+POSN (d) SCRIPT+POSN+IDENT

Figure 2: PCA visualizations embedding re-initialized using the heuristic techniques introduced in Section 3.2

Re-initialization by Position REINIT-POSN is based on the observation that within each script, embeddings seem to cluster according their word-initial vs. word-medial status (Figure 1c). Similarly to REINIT-SCRIPT, we identify the mean and standard deviation of embeddings that belong to each category. Because positional status seems to be a sub-cluster within script clusters, we only use REINIT-POSN in combination with REINIT-SCRIPT. The mean and standard deviation for each (script, position) combination is calculated and new embeddings are initialized accordingly (Figure 2c).

FOCUS Re-initialization In addition to the heuristic-based methods introduced above, we investigate a pre-existing method for embedding transfer, termed FOCUS (Dobler and de Melo, 2023). FOCUS works by extrapolating from the embedding space of an existing model, like our heuristic methods, but further introduces an auxiliary embedding model trained on the new language(s). This auxiliary model (based on FastText; Bojanowski et al., 2017) is used to obtain similarity measures between the new vocabulary items. Embeddings corresponding to overlapping tokens in the new vocabulary keep their values from the source model (REINIT-IDENT). Completely new tokens are initialized as a weighted combination of the overlapping items, with weights obtained according to similarity in the auxiliary model.

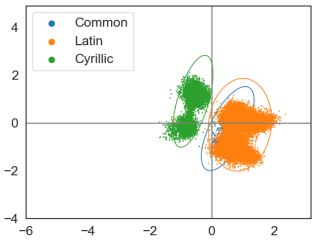


Figure 3: PCA: REINIT-FOCUS embeddings

Random Re-initialization Embeddings not initialized through the above methods are initialized according to a Standard Normal Distribution about the origin. This includes the non-overlapping tokens when REINIT-IDENT is applied on its own, and REINIT-RANDOM, where all embeddings are initialized this way.

Inspection of re-initialized embeddings Figures 2 and 3 show PCA visualizations for the re-initialization techniques described here. Figure 2a shows that while REINIT-IDENT captures some of the pre-trained embedding structure, a large number also remain randomly scattered throughout the space. REINIT-SCRIPT (2b) initializes all embeddings in a Normal distribution about the centroid for each script, but misses key embedding structure, such as the fact that each script has two position-

wise sub-distributions. REINIT-SCRIPT+POSN (2c) takes these sub-distributions into account, forming six Normal clusters instead of three.[3] Finally, REINIT-SCRIPT+POSN+IDENT (2d) and FOCUS (3) give the closest emulation of the original XLM-R embedding structure (1b).

4 Experiments

In our experiments, we replace the large cross-lingual embedding matrix of XLM-R and re-initialize it for a new, language-specific vocabulary. We then conduct LAPT to specialize the model for the new language(s), and evaluate performance on downstream tasks. We consider both multilingual→monolingual and multilingual→multilingual transfer scenarios, the latter being transfer to a much smaller set of languages than the original cross-lingual training set. We compare our vocabulary-replacement techniques against the baseline performance of XLM-R off-the-shelf, as well as LAPT while retaining the original, full-sized vocabulary.

Another manipulation we consider is whether the transformer-specific parameters are frozen during LAPT. This follows from the literature on transferring monolingual models, which proposes freezing the encoder parameters and only training the new embedding matrix to mitigate catastrophic forgetting during transfer learning (Artetxe et al., 2020; de Vries and Nissim, 2021). In our tables, we denote LAPT with trainable transformer layers as LAPT-FULL, and training with the transformer frozen (but trainable embeddings) as LAPT-EMB.

Target Languages We select our target languages for a wide selection of language families, scripts, typological characteristics, and resource availability, while still having standard evaluation sets for comparison. Training data for all languages is obtained from OSCAR v.22.01 (Abadji et al., 2022). For our lowest-resource languages, supplemental data is obtained from monolingual splits of the OPUS translation corpus (Tiedemann and Nygaard, 2004) and the Johns Hopkins University Bible Corpus (McCarthy et al., 2020). More data curation details may be found in Appendix A.

Our multilingual→monolingual transfer languages can be found in Table 1. In these experiments, the replacement vocabulary and LAPT training are constrained to a single target language. In addition, we include two multilingual→multilingual experiments. In the first, we simply transfer to the set of languages used in our monolingual experiments. Most of these languages are unrelated and cover a variety of scripts and levels of resource-availability. In the second, we transfer to a set of languages belonging to a single language family — Uralic. These languages come from the same ancestor language, and share broad grammatical features, but also use both Cyrillic and Latin scripts. These differing settings are designed to demonstrate whether language relatedness has an effect on the success of multilingual vocabulary-replacement techniques.

Vocabulary Replacement / Re-initialization When replacing model vocabulary, we train new Sentencepiece models on a subset of the training data. For targets with less than 1GB of data, we use the entire dataset. For those with more, we use a random subset of about 250MB. For multilingual models, we sample 5 million lines according to the same distribution as the training data. All new Sentencepiece models have a total vocabulary size of 32,770 including special tokens. We then initialize the embedding matrix for each new vocabulary according to one or a combination of the techniques described in Section 3.[4]

Training All of our experiments use XLM-R as a starting point (base size; Conneau et al., 2020). We conduct LAPT for 100k training steps, with evaluation checkpoints every 1000 steps. For LAPT-FULL experiments, the transformer blocks are frozen for the first 10k steps, then unfrozen for the last 90k, so that the model does not overfit to initial (possibly poor) embedding initializations. For LAPT-EMB experiments, transformer blocks remain frozen throughout training. The checkpoint obtaining the best MLM loss on a development set is selected for task fine-tuning and evaluation.

For multilingual training, we sample languages according to a multinomial distribution parameterized by $\alpha = 0.2$, following Conneau and Lample (2019), Conneau et al. (2020), i.a. Languages are sampled sentence-wise rather than batch-wise.

Evaluation We evaluate model quality with POS-tagging and NER tasks. For each task and each language, the trained model is fine-tuned on task

[3] Figure 5b in the Appendix verifies that these clusters capture the initial vs. medial token distinction

[4] The auxiliary FastText model for FOCUS initialization is trained on the same set as the vocabulary

training data until evaluation set convergence or the maximum number of epochs is reached, across four random seeds. POS performance is evaluated on Universal Dependencies (UD) treebanks (de Marneffe et al., 2021), and NER is measured on the WikiAnn benchmark (Pan et al., 2017).

5 Results

The results for monolingual adaptation can be found in Tables 1-2 and general multilingual adaptation in Tables 3-4. Because the results for multilingual adaptation to the Uralic family mostly echo overall trends, we provide these results in Appendix C.[5] In order to adhere to our overall computational budget, we only conduct full-vocabulary LAPT experiments for three languages in the monolingual setting.[6]

We first note that across re-initialization methods, LAPT-FULL always outperforms LAPT-EMB. I.e. training with trainable transformer layers outperforms training with frozen ones, despite the risk of catastrophic forgetting with the former. This trend persists across monolingual and multilingual experiments. For example, REINIT-FOCUS+IDENT shows a 6.9 average POS accuracy drop between LAPT-FULL and LAPT-EMB (Table 1).

Second, although FOCUS is the best performing re-initialization method when averaged across languages, for individual languages, it does not perform significantly differently than script-based methods. For instance, Armenian and Telugu POS tagging with script-based initialization performs on-par with or better than FOCUS (Tables 1, 3).[7] In the case of the very low-resource language Erzya, script-based methods mostly outperform FOCUS.[8]

Third, for the languages with the largest amount of data in XLM-R (Estonian, Hebrew, and Russian), the off-the-shelf performance of XLM-R (top row) is slightly better than any re-initialization method. This is not unexpected, since we can expect the highest-resource languages in XLM-R to receive adequate vocabulary coverage, and their embeddings are likely the most robustly trained.

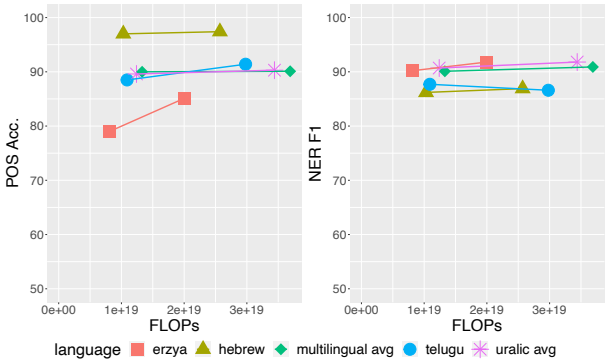


Figure 4: Evaluation scores plotted against total floating point operations of LAPT (computational cost). Left point represents cost of LAPT with reduced vocabulary, right point with full vocabulary

Finally, LAPT with the full, original XLM-R vocabulary, results in marginally better performance than other techniques. On one hand, this might be surprising given the inefficiency with which cross-lingual vocabularies often tokenize low-resource languages (Ács, 2019). On the other hand, these original pre-trained embeddings are also likely robustly aligned with the transformer encoder, which might contribute to slightly better performance.

Part of the motivation for this work, however, is to investigate *efficient* ways to specialize multilingual models. LAPT with the full XLM-R vocabulary is much more computationally costly than training new vocabulary. Figure 4 shows the trade-off between computation (in FLOPs) and performance gain in our experiments: the (often) small gains in performance we see from fine-tuning with the original vocabulary come at the cost of two to three times more FLOPs during adaptation.

Erzya POS performance provides one exception to the pattern of full-vocab LAPT providing only marginal benefits (85.1 accuracy with the full vocabulary vs. 79.0 with the reduced vocabulary). This seems surprising, given Erzya is not included in XLM-R's pre-training data, and intuitively should benefit the most from a specialized vocabulary. It could be that the reduced vocabulary size of 32k is sub-optimal for this particular target language, and/or that the new vocabulary does not overlap enough with the original (full-size) one to inherit useful Cyrillic-script embeddings. Investigating the dynamics of target vocabulary size dur-

[5] While training on related languages may be beneficial for low-resource Uralic languages like Erzya, family-based training vs. general multilingual training does not seem to alter the relative ranking of embedding initialization techniques, which is our primary research interest

[6] We select Erzya, Telugu, and Hebrew for these full-size experiments, spanning very-low, low, and medium resource-availability levels

[7] Overall performance/ranking of SCRIPT+POSN+IDENT vs. SCRIPT+IDENT remains uncertain. For LAPT-FULL averaged across languages, the former performs better in 2/3 POS settings, but only 1/3 NER settings

[8] However, script-based methods show significant variation on Erzya POS after multilingual training (Table 3)

LAPT	REINIT	Armenian	Basque	Erzya	Estonian	Hebrew	Russian	North Sami	Telugu	Avg
*	*	93.4 ± 2.2	95.1 ± 0.7	56.3 ± 5.3	95.6 ± 0.1	97.5 ± 0.1	98.6 ± 0.1	71.2 ± 1.8	83.8 ± 0.1	86.4
FULL	*	-	-	85.1 ± 1.8	-	97.5 ± 0.1	-	-	91.4 ± 4.3	-
FULL	FOCUS+IDENT	**92.3 ± 1.9**	**96.0 ± 0.6**	76.1 ± 2.0	**95.1 ± 0.3**	**97.2 ± 0.1**	**98.4 ± 0.1**	**92.1 ± 0.8**	**86.9 ± 3.5**	**91.7**
FULL	SCRIPT+POSN+IDENT	**93.1 ± 1.7**	93.8 ± 0.5	**79.0 ± 0.7**	94.0 ± 0.2	96.7 ± 0.1	98.2 ± 0.04	86.9 ± 0.7	**88.5 ± 3.2**	91.3
FULL	SCRIPT+IDENT	91.7 ± 1.9	93.6 ± 0.3	70.8 ± 12.8	94.0 ± 0.1	96.7 ± 0.1	98.1 ± 0.1	83.4 ± 1.3	**87.1 ± 3.4**	89.4
FULL	SCRIPT+POSN	90.9 ± 2.0	92.1 ± 0.7	74.6 ± 2.2	90.4 ± 0.6	95.4 ± 0.1	97.2 ± 0.02	78.7 ± 0.5	**87.5 ± 1.4**	88.3
FULL	SCRIPT	89.6 ± 1.5	90.9 ± 0.2	71.5 ± 2.1	89.4 ± 0.9	95.0 ± 0.05	96.9 ± 0.03	77.9 ± 0.2	84.0 ± 1.5	86.9
FULL	IDENT	81.6 ± 0.4	83.6 ± 0.6	59.1 ± 3.1	86.4 ± 0.4	91.1 ± 0.1	96.2 ± 0.04	70.7 ± 0.5	78.0 ± 2.5	80.9
FULL	RANDOM	67.4 ± 2.0	72.7 ± 0.6	53.3 ± 2.8	72.0 ± 0.1	81.0 ± 0.6	86.5 ± 0.6	64.7 ± 0.9	76.4 ± 1.0	72.4
EMB	FOCUS+IDENT	**92.3 ± 1.7**	**95.1 ± 0.6**	48.6 ± 0.1	**94.5 ± 0.05**	**96.9 ± 0.3**	**98.3 ± 0.04**	**73.6 ± 1.6**	**86.2 ± 3.8**	**84.8**
EMB	SCRIPT+POSN+IDENT	87.6 ± 1.3	88.2 ± 0.7	**55.6 ± 4.8**	89.6 ± 0.1	95.3 ± 0.1	97.1 ± 0.05	69.8 ± 1.4	81.8 ± 1.2	82.5
EMB	SCRIPT+IDENT	87.7 ± 1.8	87.9 ± 0.4	**53.8 ± 5.4**	89.2 ± 0.5	95.2 ± 0.1	97.0 ± 0.1	68.6 ± 1.8	82.0 ± 1.3	82.0
EMB	SCRIPT+POSN	56.5 ± 7.6	61.3 ± 12.0	48.7 ± 0.1	71.4 ± 1.4	82.5 ± 0.3	92.1 ± 0.4	59.8 ± 1.5	70.1 ± 7.4	69.4
EMB	SCRIPT	47.6 ± 6.4	59.6 ± 8.1	48.6 ± 0.1	65.7 ± 5.2	80.4 ± 2.2	89.7 ± 1.0	55.5 ± 5.0	73.4 ± 5.5	67.6
EMB	IDENT	80.3 ± 1.1	80.1 ± 0.6	47.9 ± 1.5	82.5 ± 1.8	88.7 ± 0.2	95.2 ± 0.4	60.6 ± 1.2	76.6 ± 1.4	75.9
EMB	RANDOM	47.6 ± 1.8	55.2 ± 2.8	46.3 ± 0.2	63.5 ± 1.8	67.6 ± 2.5	80.2 ± 0.6	44.7 ± 4.0	56.7 ± 6.7	59.2

Table 1: Monolingual Language-Adaptive Pre-Training (LAPT): POS tagging accuracy after fine-tuning. * indicates XLM-R off-the-shelf. Within each division, best result and results within 1 standard deviation are bolded; overall best result indicated with added underline. Best result determined by *mean - stdev*. LAPT with full XLM-R vocab only conducted for three languages due to prohibitive computational cost

LAPT	REINIT	Armenian	Basque	Erzya	Estonian	Hebrew	Russian	Telugu	Avg
*	*	94.1 ± 0.1	94.3 ± 0.1	89.5 ± 0.6	93.3 ± 0.2	85.9 ± 0.1	90.9 ± 0.2	85.4 ± 0.5	90.5
FULL	*	-	-	91.8 ± 0.5	-	86.9 ± 0.1	-	86.6 ± 1.9	-
FULL	FOCUS+IDENT	**95.1 ± 0.9**	**94.9 ± 0.4**	**89.9 ± 0.8**	**92.6 ± 0.2**	**86.2 ± 0.3**	**90.6 ± 0.1**	**87.7 ± 0.5**	**91.0**
FULL	SCRIPT+POSN+IDENT	93.9 ± 0.1	94.3 ± 0.2	**90.2 ± 0.7**	92.0 ± 0.3	83.2 ± 0.4	89.8 ± 0.2	83.5 ± 1.8	89.6
FULL	SCRIPT+IDENT	93.8 ± 0.3	94.3 ± 0.1	**89.8 ± 0.2**	89.3 ± 0.2	83.4 ± 0.3	89.4 ± 0.2	84.0 ± 0.5	89.5
FULL	SCRIPT+POSN	92.0 ± 0.6	92.1 ± 0.04	89.1 ± 0.5	88.3 ± 0.4	78.7 ± 0.1	86.5 ± 0.1	81.0 ± 0.9	86.8
FULL	SCRIPT	91.4 ± 0.4	91.1 ± 0.1	87.7 ± 0.5	87.5 ± 0.2	78.5 ± 0.2	85.7 ± 0.1	79.6 ± 1.1	85.9
FULL	IDENT	86.2 ± 0.4	90.7 ± 0.2	79.0 ± 0.6	89.3 ± 0.2	72.0 ± 0.4	86.7 ± 0.1	69.3 ± 0.4	81.9
FULL	RANDOM	74.1 ± 1.4	81.5 ± 0.3	72.6 ± 3.3	45.8 ± 27.2	54.4 ± 0.9	70.3 ± 0.7	47.2 ± 8.2	63.7
EMB	FOCUS+IDENT	**93.5 ± 0.5**	**94.2 ± 0.2**	81.7 ± 2.2	**92.0 ± 0.2**	**84.9 ± 0.1**	**90.3 ± 0.1**	**86.1 ± 0.3**	**89.0**
EMB	SCRIPT+POSN+IDENT	91.5 ± 0.2	92.3 ± 0.1	**87.2 ± 0.3**	89.8 ± 0.2	79.1 ± 0.2	88.9 ± 0.1	74.1 ± 1.2	86.1
EMB	SCRIPT+IDENT	90.9 ± 0.3	92.0 ± 0.3	86.1 ± 1.0	89.6 ± 0.3	78.7 ± 0.3	88.6 ± 0.1	79.1 ± 0.5	86.4
EMB	SCRIPT+POSN	86.5 ± 0.4	87.3 ± 0.3	84.1 ± 1.2	81.8 ± 0.8	71.0 ± 0.9	81.0 ± 0.2	64.3 ± 1.9	79.4
EMB	SCRIPT	83.9 ± 0.4	73.0 ± 0.8	84.0 ± 1.2	79.5 ± 0.9	67.8 ± 0.6	77.4 ± 0.2	56.8 ± 3.2	74.6
EMB	IDENT	80.9 ± 0.8	87.9 ± 0.4	61.8 ± 3.8	85.3 ± 0.3	64.8 ± 1.4	84.8 ± 0.4	54.9 ± 1.5	74.3
EMB	RANDOM	59.6 ± 2.5	0.0 ± 0.0	51.8 ± 2.7	0.0 ± 0.0	17.1 ± 17.2	47.5 ± 6.9	22.4 ± 5.5	28.3

Table 2: Monolingual LAPT: entity-wise NER F1 score after fine-tuning. A score of 0.0 results from the model learning to output only class O (not a named entity) which is the majority class. Sami does not have enough NER data for fine-tuning

ing vocabulary specialization would be a fruitful direction for future work.

6 Discussion

Embedding-only training is inadequate for multilingual model transfer Our experiments show that language transfer methods developed for monolingual models, which freeze the transformer blocks and re-train only the embedding matrix (Artetxe et al., 2020; de Vries and Nissim, 2021), yield poor results when transferring a multilingual model. This work in the monolingual literature not only keeps transformer layers frozen, but initializes new embeddings randomly. This setup (LAPT-EMB, REINIT-RANDOM) performs much worse than the off-the-shelf baseline in all of our experiments.

It is worth noting that Artetxe et al. (2020) do not necessarily suggest that freezing the main model is the *optimal* language transfer method. However, it does demonstrate that for monolingual→monolingual adaptation, embedding-only training is competitive with an off-the-shelf multilingual model. We see no such comparability in our experiments. We believe this is partly caused by the heterogeneity of the XLM-R embeddings, where different languages (or at least scripts) are encoded in different spaces. When new embeddings are randomly and homogeneously initialized, they fail to align with the pre-trained subspaces expected by the frozen transformer.

Vocab replacement efficiently specializes models We demonstrate that for languages inadequately covered by a pre-trained multilingual model, replacing and re-training the cross-lingual model vocabulary with a language-specific one is a computationally efficient way to create a compact model specialized for the target language(s). In our monolingual adaptation experiments, vocabulary replacement performs better than off-the-shelf XLM-R in 5/8 languages for POS tagging and 5/7 languages

LAPT	REINIT	Armenian	Basque	Erzya	Estonian	Hebrew	Russian	North Sami	Telugu	Avg
*	*	93.4 ± 2.2	95.1 ± 0.7	56.3 ± 5.3	95.6 ± 0.1	97.5 ± 0.1	98.6 ± 0.1	71.2 ± 1.8	83.8 ± 0.1	86.4
FULL	*	91.3 ± 0.1	95.9 ± 0.6	71.7 ± 5.3	95.5 ± 0.2	97.4 ± 0.2	98.6 ± 0.04	80.6 ± 1.4	89.7 ± 3.6	90.1
FULL	FOCUS+IDENT	91.0 ± 0.1	**95.8 ± 0.1**	**72.5 ± 1.3**	**95.5 ± 0.2**	**97.1 ± 0.1**	**98.4 ± 0.03**	**80.4 ± 1.2**	**89.4 ± 3.2**	**90.0**
FULL	SCRIPT+POSN+IDENT	**92.9 ± 2.1**	95.0 ± 0.6	**63.6 ± 9.8**	94.8 ± 0.3	97.0 ± 0.1	**98.4 ± 0.04**	**80.4 ± 1.1**	**89.6 ± 2.6**	89.0
FULL	SCRIPT+IDENT	**93.8 ± 1.8**	95.3 ± 0.03	**66.1 ± 10.2**	94.7 ± 0.2	**97.1 ± 0.1**	**98.4 ± 0.03**	**80.1 ± 1.2**	**91.7 ± 0.8**	89.7
FULL	SCRIPT+POSN	85.3 ± 3.5	87.9 ± 3.5	70.5 ± 1.5	89.0 ± 0.8	93.7 ± 0.6	97.2 ± 0.01	72.8 ± 2.1	81.6 ± 0.4	84.7
FULL	SCRIPT	83.3 ± 1.9	85.8 ± 2.7	66.6 ± 1.9	85.4 ± 1.7	90.5 ± 0.8	96.8 ± 0.03	68.6 ± 1.1	81.0 ± 0.3	82.2
FULL	IDENT	**93.2 ± 0.7**	93.0 ± 0.5	58.1 ± 0.9	93.6 ± 0.2	96.6 ± 0.1	98.3 ± 0.03	71.5 ± 1.2	89.0 ± 4.1	86.7
FULL	RANDOM	64.5 ± 2.9	67.4 ± 0.4	50.0 ± 4.6	71.9 ± 0.3	80.0 ± 0.8	84.6 ± 0.9	62.7 ± 0.5	75.0 ± 6.2	70.2
EMB	FOCUS+IDENT	**93.1 ± 2.2**	**95.2 ± 0.7**	**63.7 ± 2.0**	**94.7 ± 0.1**	**97.1 ± 0.04**	**98.5 ± 0.03**	71.2 ± 2.1	**87.5 ± 2.9**	**86.8**
EMB	SCRIPT+POSN+IDENT	**91.3 ± 1.6**	93.5 ± 0.6	57.2 ± 7.0	93.5 ± 0.1	96.7 ± 0.03	98.3 ± 0.1	**74.5 ± 1.1**	**85.6 ± 2.9**	85.6
EMB	SCRIPT+IDENT	**92.2 ± 2.0**	93.2 ± 0.7	58.5 ± 6.9	93.3 ± 0.1	96.9 ± 0.1	98.3 ± 0.02	**72.0 ± 3.0**	**86.5 ± 2.4**	85.5
EMB	SCRIPT+POSN	61.5 ± 1.9	76.0 ± 1.3	51.9 ± 3.1	75.7 ± 0.2	87.2 ± 1.2	95.3 ± 0.3	65.3 ± 0.2	77.3 ± 0.3	75.5
EMB	SCRIPT	44.7 ± 0.0	71.0 ± 1.0	48.5 ± 0.2	73.5 ± 2.2	83.6 ± 0.3	93.5 ± 0.5	63.8 ± 1.4	77.7 ± 0.5	73.1
EMB	IDENT	89.4 ± 0.8	90.5 ± 0.6	49.3 ± 4.6	91.8 ± 0.5	96.2 ± 0.1	98.1 ± 0.1	65.6 ± 1.1	84.0 ± 1.7	82.2
EMB	RANDOM	48.7 ± 2.4	61.2 ± 5.6	46.0 ± 0.3	66.3 ± 3.9	73.7 ± 3.4	85.1 ± 1.2	44.7 ± 4.6	67.5 ± 5.0	63.5

Table 3: Multilingual LAPT: POS tagging accuracy after fine-tuning

LAPT	REINIT	Armenian	Basque	Erzya	Estonian	Hebrew	Russian	Telugu	Avg
*	*	94.1 ± 0.1	94.3 ± 0.1	89.5 ± 0.6	93.3 ± 0.2	85.9 ± 0.1	90.9 ± 0.2	85.4 ± 0.5	90.5
FULL	*	94.0 ± 0.5	94.5 ± 0.2	90.5 ± 0.3	93.7 ± 0.2	86.2 ± 0.1	91.1 ± 0.2	85.9 ± 0.7	90.9
FULL	FOCUS+IDENT	**94.2 ± 0.3**	**94.0 ± 0.2**	**89.6 ± 1.0**	**92.0 ± 0.5**	**85.2 ± 0.1**	**90.0 ± 0.5**	**85.4 ± 0.4**	**90.1**
FULL	SCRIPT+POSN+IDENT	**94.1 ± 0.2**	**94.0 ± 0.1**	88.8 ± 0.9	**92.3 ± 0.1**	85.0 ± 0.2	**90.4 ± 0.1**	84.8 ± 0.4	89.9
FULL	SCRIPT+IDENT	**94.2 ± 0.2**	**94.1 ± 0.2**	**90.1 ± 0.6**	**92.4 ± 0.1**	84.9 ± 0.3	90.3 ± 0.1	84.5 ± 0.2	90.0
FULL	SCRIPT+POSN	91.2 ± 0.5	91.5 ± 0.1	88.9 ± 0.5	88.4 ± 0.4	77.3 ± 0.4	86.3 ± 0.1	76.2 ± 0.4	85.7
FULL	SCRIPT	90.9 ± 0.1	91.3 ± 0.3	86.4 ± 1.9	87.7 ± 0.2	75.8 ± 0.3	85.7 ± 0.1	75.1 ± 0.9	84.7
FULL	IDENT	93.2 ± 0.1	93.4 ± 0.2	80.9 ± 2.4	91.5 ± 0.4	83.5 ± 0.3	89.8 ± 0.1	83.2 ± 0.5	87.9
FULL	RANDOM	69.9 ± 4.4	80.9 ± 0.5	75.2 ± 1.5	70.5 ± 2.1	37.7 ± 21.8	68.6 ± 0.7	42.1 ± 1.6	63.6
EMB	FOCUS+IDENT	**93.9 ± 0.3**	**93.7 ± 0.2**	**89.7 ± 0.4**	**91.9 ± 0.4**	**84.8 ± 0.2**	**89.9 ± 0.3**	**85.2 ± 0.5**	**89.9**
EMB	SCRIPT+POSN+IDENT	**93.7 ± 0.2**	93.5 ± 0.1	87.2 ± 1.0	**91.9 ± 0.2**	84.0 ± 0.2	**89.9 ± 0.2**	84.0 ± 0.5	89.2
EMB	SCRIPT+IDENT	93.3 ± 0.5	93.4 ± 0.2	85.8 ± 1.4	**91.9 ± 0.3**	83.7 ± 0.2	**89.9 ± 0.1**	82.5 ± 1.3	88.7
EMB	SCRIPT+POSN	87.5 ± 0.3	88.8 ± 0.3	81.0 ± 3.1	84.8 ± 0.4	72.8 ± 0.1	82.7 ± 0.3	67.1 ± 1.3	80.7
EMB	SCRIPT	85.2 ± 0.3	81.3 ± 7.1	80.0 ± 1.1	84.3 ± 0.3	68.3 ± 0.9	80.6 ± 1.0	59.7 ± 3.5	77.1
EMB	IDENT	91.2 ± 0.3	92.3 ± 0.2	76.7 ± 1.3	90.8 ± 0.3	81.6 ± 0.2	89.3 ± 0.2	78.6 ± 1.8	85.8
EMB	RANDOM	62.8 ± 0.9	74.9 ± 1.6	66.1 ± 1.1	62.7 ± 1.9	23.9 ± 18.2	53.1 ± 4.7	37.7 ± 2.6	54.4

Table 4: Multilingual LAPT: entity-wise NER F1 score after fine-tuning

for NER. Only the high-resource languages of Estonian, Hebrew, and Russian seem to be adequately covered in XLM-R to outperform our specialization techniques. Language-Adaptive Pre-Training with the full (cross-lingual) XLM-R vocabulary often produces marginally better results overall, but at a much greater computational cost, and without making the model more compact in size. Further training and inference after LAPT will continue to suffer from the memory and compute wasted on unused vocabulary items, which constitute a large percentage of the total model parameters.

Script-distribution initialization rivals semantic similarity methods We introduced several methods for embedding re-initialization in Section 3, namely using the insight that token embeddings for XLM-R cluster by script and position within a word, then distributing new vocabulary items according to these pre-trained sub-distributions. We compare this to the FOCUS re-initialization method, which initializes new embeddings as a weighted combination of existing ones according to similarity scores from an auxiliary model.

Averaged across languages, FOCUS yields the best performance in downstream tasks by a slight margin. Within languages, it often overlaps significantly with the performance of our script-distribution methods. For very low-resource languages like Erzya, script-based methods even show a slight advantage. This seems to show that, at least in combination with LAPT, the majority of the benefit in re-initialization can be achieved by a method that takes the structure of the pre-trained embedding distribution into account, whether or not it uses advanced methods to precisely initialize the representations of new vocabulary items.

We do note that the advantage of FOCUS is more clear-cut when LAPT is conducted with transformer blocks frozen. This lends credence to the idea that FOCUS more precisely mimics the embedding distribution expected by the pre-trained transformer. However, the overall best results come when the transformer blocks are unfrozen/trainable.

Fully random initialization performs poorly Finally, our experiments demonstrate that fully random re-initialization of embeddings during vocabulary replacement leads to overall poor performance. Across LAPT-FULL experiments, random initial-

ization performs an average of 19.4 points worse than the next-best re-initialization method, and 24.7 points worse than the off-the-shelf baseline. The poor performance of random initialization has been noted in other works such as Dobler and de Melo (2023), but we emphasize that even incredibly simple methods such as REINIT-IDENT and REINIT-SCRIPT work far better than the random baseline.

7 Conclusion

This work presents a systematic comparison of methods to specialize the subword vocabularies and embeddings of multilingual models for new languages. We propose simple methods for re-initializing embeddings, motivated by a qualitative exploration of the XLM-R embedding space. Our experiments show that (1) updating the encoder layers during LAPT is crucial for downstream performance, (2) vocabulary replacement provides a computationally-efficient method to improve task performance in low-resource languages, and (3) our re-initialization techniques employing script-wise sub-distributions perform on par with more involved similarity-based methods. We hope these findings can be built upon in future work on multilingual model specialization, with the goal of providing the best performance for under-resourced languages while also making language modeling more accessible through more manageable compute cost and model sizes.

Limitations

One limitation of our work is the relatively narrow set of evaluation tasks available for our languages of interest. The model-adaptation techniques we compare here are most applicable to low- and medium-resource languages that are not optimally covered by pre-existing multilingual models. For most of these languages, the only standard evaluation datasets that exist are for relatively low-level tasks like Part of Speech tagging and Named Entity Recognition. Evaluation of embedding-reinitialization techniques could be improved in future work if datasets for higher-level tasks like Natural Language Inference, question answering, and paraphrase detection were curated for these under-resourced languages.

We also make several simplifying choices to maintain a feasible scope for our work. First, we conduct model adaptation from only a single base model: XLM-R. A valuable addition in future work would be to determine whether the trends we observe here generalize to other model types (i.e. causal and seq2seq language models) and to larger model scales. Secondly, we consider only one size for newly-initialized target vocabularies (32k). Because effective per-language vocabulary allocation has been shown to be an important factor in multilingual modeling (Conneau et al., 2020, i.a.), investigating the dynamics of target vocabulary size during vocabulary re-initialization will be important for future work on this topic.

Acknowledgements

We thank Ibrahim Sharaf, Anita Silva, and Peter Zuckerman for early investigation of data availability for low-resource languages. We are also gracious to Emily P. Ahn, Gina-Anne Levow, Sara Ng, and our anonymous MRL reviewers for useful feedback and discussion.

References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4344–4355, Marseille, France. European Language Resources Association.

Judit Ács. 2019. Exploring BERT's vocabulary.

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, Online. Association for Computational Linguistics.

Mikko Aulamo, Sami Virpioja, and Jörg Tiedemann. 2020. OpusFilter: A configurable parallel corpus filtering toolbox. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 150–156, Online. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. Parsing with multilingual BERT, a small corpus, and a small treebank. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. 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. 2020. 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 and Guillaume Lample. 2019. Cross-lingual Language Model Pretraining. In *Advances in Neural Information Processing Systems*, volume 32, Vancouver, Canada. Curran Associates, Inc.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Wietse de Vries and Malvina Nissim. 2021. As good as new. how to successfully recycle English GPT-2 to make models for other languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846, Online. Association for Computational Linguistics.

Jacob Devlin. 2019. Multilingual BERT Readme. https://github.com/google-research/bert/blob/master/multilingual.md.

Konstantin Dobler and Gerard de Melo. 2023. Focus: Effective embedding initialization for specializing pretrained multilingual models on a single language. *arXiv preprint arXiv:2305.14481*.

Abteen Ebrahimi and Katharina Kann. 2021. How to adapt your pretrained multilingual model to 1600 languages. 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 4555–4567, Online. Association for Computational Linguistics.

Fahim Faisal and Antonios Anastasopoulos. 2022. Phylogeny-inspired adaptation of multilingual models to new languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 434–452, Online only. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*, San Diego, CA, USA.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

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

Arya D. McCarthy, Rachel Wicks, Dylan Lewis, Aaron Mueller, Winston Wu, Oliver Adams, Garrett Nicolai, Matt Post, and David Yarowsky. 2020. The Johns Hopkins University Bible corpus: 1600+ tongues for typological exploration. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2884–2892, Marseille, France. European Language Resources Association.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. 2022. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3992–4006, Seattle, United States. Association for Computational Linguistics.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. 2021. Small data? no problem! exploring the viability

of pretrained multilingual language models for low-resourced languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tolulope Ogunremi, Dan Jurafsky, and Christopher Manning. 2023. Mini but mighty: Efficient multilingual pretraining with linguistically-informed data selection. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1251–1266, Dubrovnik, Croatia. Association for Computational Linguistics.

Malte Ostendorff and Georg Rehm. 2023. Efficient Language Model Training through Cross-Lingual and Progressive Transfer Learning. ArXiv:2301.09626 [cs].

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.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2021. UNKs everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10186–10203, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, and Inter Alia. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Jörg Tiedemann and Lars Nygaard. 2004. The OPUS corpus - parallel and free: `http://logos.uio.no/opus`. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. UDapter: Language adaptation for truly Universal Dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315, Online. Association for Computational Linguistics.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. 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.

Qingcheng Zeng, Lucas Garay, Peilin Zhou, Dading Chong, Yining Hua, Jiageng Wu, Yikang Pan, Han Zhou, Rob Voigt, and Jie Yang. 2023. GreenPLM: Cross-Lingual Transfer of Monolingual Pre-Trained Language Models at Almost No Cost. volume 6, pages 6290–6298. ISSN: 1045-0823.

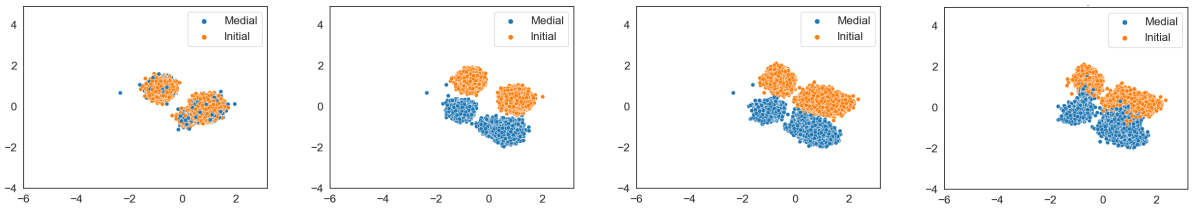
A Data Details

General information about the language data used in this study can be found in Table 5. All training data used in our experiments is cleaned and deduplicated using the OpusFilter package (Aulamo et al., 2020). For the lowest-resource languages (Erzya and Sami) we additionally filter out lines that are identified as English with a probability of 90% or higher, since positive automatic language-identification for low-resource languages is likely not robust (Kreutzer et al., 2022). We additionally filter out lines composed of less than 2 tokens, lines with an average token length of greater than 16 characters, lines with tokens longer than 32 characters, and lines composed of fewer than 50% alphabetic characters.

For POS tagging evaluation, most languages have a standard train/dev/test split curated the original Universal Dependencies dataset (de Marneffe et al., 2021). Erzya, however, only has a standard train/test split. To form a dev split, we randomly sample 300 sentences from the train split. The WikiAnn dataset (Pan et al., 2017) does not ship with standard train/dev/test splits, so we create random 85/5/10% splits of each language for this purpose, with a minimum dev/test size of 256 and 512 sentences respectively.

B Training Details

The main details of our experimental process can be found in Section 4. Here we provide our choice of hyperparameters and other details relevant to reproducibility. The code used to run all experiments will be released in a later version of this



(a) SCRIPT (b) SCRIPT+POSN (c) SCRIPT+POSN+IDENT (d) FOCUS

Figure 5: PCA visualization of re-initialized embeddings with word-initial vs word-medial tokens highlighted. For REINIT-SCRIPT, position-wise clustering seen in the base XLM-R embeddings (Figure 1c) is not captured. REINIT-SCRIPT+POSN and REINIT-SCRIPT+POSN+IDENT show expected positional clustering. REINIT-FOCUS seems to allow slightly more positional overlap

Language	Code	Family	Script	XLM-R Data (GB)	LAPT Data (GB)
Armenian	hy	Indo-European	Armenian	5.5	1.2
Basque	eu	isolate	Latin	2.0	0.35
Erzya	myv	Uralic	Cyrillic	0	0.006
Estonian	et	Uralic	Latin	6.1	3.0
Finnish	fi	Uralic	Latin	54.3	9.1
Hebrew	he	Afro-Asiatic	Hebrew	31.6	7.7
Hungarian	hu	Uralic	Latin	58.4	13.0
Russian	ru	Indo-European	Cyrillic	278.0	10.0
Sami	sme	Uralic	Latin	0	0.004
Telugu	te	Dravidian	Telugu	4.7	0.9

Table 5: Training data breakdown by language. XLM-R data is the amount of data used in the pre-training of that model. LAPT data is the amount used for training in our current experiments, after cleaning/deduplicating.

paper. All models are trained and fine-tuned on Nvidia Quadro RTX 6000 GPUs using the Adam optimizer (Kingma and Ba, 2015).

Hyperparameters for Language-Adaptive Pre-Training (LAPT) can be found in Table 6. If NaN losses were encountered during training, `max_gradient_norm` was reduced to 0.5. For multilingual sampling during training, each language's training data is capped at approximately 2GB.

Hyperparameters for task fine-tuning on POS and NER are in Table 7. For NER, the reported evaluation metric is entity-wise F1, meaning tokens with label `O` are ignored. In order to prevent models from learning to output only the majority class `O` during training, the loss for the `O` tokens in each batch is down-weighted to have the same influence as the tokens that actually correspond to a named entity. We cap fine-tuning training data at 32,768 sequences.

C Uralic Results

The results for multilingual adaptation to the Uralic family can be found in Tables 8 and 9. These results mostly follow the trends discussed in Section 5 (LAPT-EMB consistently underperforms LAPT-FULL, off-the-shelf performance is best for high-resource languages, LAPT with full cross-lingual vocab performs marginally better than other methods). It should be noted that for both Erzya and Hungarian, the best POS accuracy is achieved with SCRIPT+POSN+IDENT initialization (better even than LAPT with the fully cross-lingual vocabulary). Results for the very low-resource language Erzya are generally higher than with multilingual training on unrelated languages, which could suggest a benefit to training with closely-related languages. This observation does not clearly hold for Sami (the other very low-resource language), however. Note that Russian is not a Uralic language — we include it for multilingual training in order to robustly train embeddings for the Cyrillic script, in which Erzya is written. Erzya is also spoken primarily within the Russian Federation, making loan-words likely.

Hyperparameter	Value
mlm_masking_prob	0.15
max_sequence_length	256
learning_rate	1e-5
lr_schedule	linear
batch_size	200
max_gradient_norm	1.0

Table 6: Hyperparameters for model training (LAPT)

Hyperparameter	Value
max_sequence_length	256
learning_rate	5e-6
lr_schedule	constant
max_epochs	64
eval_interval (epochs)	2
patience (epochs)	8 (POS) / 4 (NER)
batch_size	72
max_gradient_norm	1.0

Table 7: Hyperparameters for model task fine-tuning

LAPT	REINIT	Erzya	Estonian	Finnish	Hungarian	North Sami	Russian	Avg
*	*	56.3 ± 5.3	95.6 ± 0.1	97.5 ± 0.1	93.7 ± 1.5	71.2 ± 1.8	98.6 ± 0.1	85.9
FULL	*	72.5 ± 2.6	95.8 ± 0.1	97.7 ± 0.2	94.1 ± 1.9	82.9 ± 0.4	98.6 ± 0.04	90.3
FULL	FOCUS+IDENT	**73.8 ± 2.7**	**95.3 ± 0.2**	**97.2 ± 0.1**	92.5 ± 1.6	**80.1 ± 1.4**	**98.4 ± 0.04**	**89.6**
FULL	SCRIPT+POSN+IDENT	**73.0 ± 1.4**	94.7 ± 0.3	96.6 ± 0.1	**94.8 ± 0.7**	**78.0 ± 2.3**	**98.4 ± 0.01**	89.3
FULL	SCRIPT+IDENT	67.7 ± 11.0	94.3 ± 0.3	96.4 ± 0.1	**94.7 ± 0.7**	**78.8 ± 2.2**	**98.4 ± 0.03**	88.4
FULL	SCRIPT+POSN	71.2 ± 2.7	88.7 ± 0.4	90.6 ± 0.1	86.8 ± 0.4	72.9 ± 2.0	97.2 ± 0.02	84.7
FULL	SCRIPT	65.9 ± 4.6	85.6 ± 1.3	89.1 ± 0.3	85.2 ± 0.2	73.5 ± 1.6	96.9 ± 0.05	82.7
FULL	IDENT	59.8 ± 1.2	92.2 ± 0.03	95.2 ± 0.04	91.8 ± 2.8	68.9 ± 0.9	98.2 ± 0.03	84.3
FULL	RANDOM	53.7 ± 3.2	71.9 ± 0.6	73.1 ± 0.2	59.6 ± 1.6	63.9 ± 0.9	84.9 ± 1.9	67.8
EMB	FOCUS+IDENT	**66.3 ± 1.2**	**94.7 ± 0.1**	**96.8 ± 0.2**	**94.2 ± 0.8**	**73.3 ± 1.6**	**98.4 ± 0.05**	**87.3**
EMB	SCRIPT+POSN+IDENT	64.2 ± 2.8	93.0 ± 0.1	95.5 ± 0.03	**93.6 ± 0.8**	**72.7 ± 2.6**	98.3 ± 0.05	86.2
EMB	SCRIPT+IDENT	55.8 ± 4.1	92.8 ± 0.2	95.4 ± 0.04	92.3 ± 1.6	69.8 ± 1.6	98.3 ± 0.04	84.1
EMB	SCRIPT+POSN	54.5 ± 4.3	74.2 ± 0.8	79.5 ± 0.7	62.1 ± 2.6	65.2 ± 2.0	94.8 ± 0.4	71.7
EMB	SCRIPT	48.7 ± 0.04	56.9 ± 15.6	71.6 ± 3.2	54.3 ± 4.4	58.0 ± 1.7	91.4 ± 1.8	63.5
EMB	IDENT	49.2 ± 1.7	90.6 ± 0.4	94.4 ± 0.03	84.8 ± 2.9	64.7 ± 1.3	97.9 ± 0.1	80.3
EMB	RANDOM	48.6 ± 0.2	64.5 ± 4.1	66.4 ± 1.2	43.6 ± 0.1	45.8 ± 4.2	84.0 ± 1.4	58.8

Table 8: Uralic family multilingual LAPT: POS tagging accuracy after fine-tuning

LAPT	REINIT	Erzya	Estonian	Finnish	Hungarian	Russian	Avg
*	*	89.5 ± 0.6	93.3 ± 0.2	90.7 ± 0.1	92.4 ± 0.1	90.9 ± 0.2	91.4
FULL	*	90.5 ± 0.5	93.8 ± 0.2	91.0 ± 0.2	92.4 ± 0.3	91.0 ± 0.2	91.8
FULL	FOCUS+IDENT	**89.4 ± 1.7**	**92.5 ± 0.1**	**89.8 ± 0.2**	**91.2 ± 0.4**	**90.4 ± 0.1**	**90.7**
FULL	SCRIPT+POSN+IDENT	88.7 ± 0.5	92.2 ± 0.4	89.2 ± 0.2	90.9 ± 0.2	90.1 ± 0.1	90.2
FULL	SCRIPT+IDENT	**89.3 ± 0.4**	**92.7 ± 0.3**	89.2 ± 0.4	**91.3 ± 0.1**	90.0 ± 0.2	90.5
FULL	SCRIPT+POSN	**89.5 ± 1.0**	87.9 ± 0.2	84.2 ± 0.3	86.3 ± 0.3	86.2 ± 0.2	86.8
FULL	SCRIPT	**88.9 ± 0.8**	87.5 ± 0.3	83.3 ± 0.1	86.3 ± 0.2	85.5 ± 0.1	86.3
FULL	IDENT	81.1 ± 0.8	91.6 ± 0.1	88.2 ± 0.2	90.7 ± 0.3	89.6 ± 0.1	88.2
FULL	RANDOM	73.7 ± 2.7	53.1 ± 30.7	0.0 ± 0.0	32.9 ± 33.0	65.1 ± 2.2	45.0
EMB	FOCUS+IDENT	**88.6 ± 0.6**	**92.4 ± 0.3**	**89.6 ± 0.1**	**91.1 ± 0.1**	**90.0 ± 0.1**	**90.3**
EMB	SCRIPT+POSN+IDENT	86.6 ± 1.1	91.4 ± 0.2	88.8 ± 0.3	90.5 ± 0.2	89.9 ± 0.1	89.4
EMB	SCRIPT+IDENT	87.0 ± 1.3	91.8 ± 0.1	88.6 ± 0.3	**91.0 ± 0.2**	89.6 ± 0.2	89.6
EMB	SCRIPT+POSN	85.0 ± 1.2	84.2 ± 0.4	78.1 ± 0.3	81.9 ± 0.5	82.1 ± 0.2	82.3
EMB	SCRIPT	82.9 ± 2.6	82.4 ± 1.3	72.5 ± 1.3	80.7 ± 0.4	79.0 ± 0.2	79.5
EMB	IDENT	71.0 ± 4.4	90.1 ± 0.3	87.0 ± 0.4	89.9 ± 0.2	88.7 ± 0.1	85.3
EMB	RANDOM	64.9 ± 1.9	0.0 ± 0.0	13.6 ± 23.5	0.0 ± 0.0	54.4 ± 2.2	26.6

Table 9: Uralic family multilingual LAPT: entity-wise NER F1 score after fine-tuning. A score of 0.0 results from the model learning to output only class `O` (not a named entity) which is the majority class. Sami does not have enough NER data for fine-tuning