

Dictionaries to the Rescue: Cross-Lingual Vocabulary Transfer for Low-Resource Languages Using Bilingual Dictionaries

Haruki Sakajo¹, Yusuke Ide¹, Justin Vasselli¹,
Yusuke Sakai¹, Yingtao Tian² Hidetaka Kamigaito¹, Taro Watanabe¹

¹Nara Institute of Science and Technology (NAIST), ²Sakana AI
sakajo.haruki.sd9@naist.ac.jp
{sakai.yusuke.sr9, kamigaito.h, taro}@is.naist.jp

Abstract

Cross-lingual vocabulary transfer plays a promising role in adapting pre-trained language models to new languages, including low-resource languages. Existing approaches that utilize monolingual or parallel corpora face challenges when applied to languages with limited resources. In this work, we propose a simple yet effective vocabulary transfer method that utilizes bilingual dictionaries, which are available for many languages, thanks to descriptive linguists. Our proposed method leverages a property of BPE tokenizers where removing a subword from the vocabulary causes a fallback to shorter subwords. The embeddings of target subwords are estimated iteratively by progressively removing them from the tokenizer. The experimental results show that our approach outperforms existing methods for low-resource languages, demonstrating the effectiveness of a dictionary-based approach for cross-lingual vocabulary transfer¹.

1 Introduction

Vocabulary transfer, or adaptation, is a method to bridge the gap between languages in language models (Wang et al., 2020; Chau et al., 2020) to improve their performance for new languages. Vocabulary transfer can also address token over-fragmentation (Ahia et al., 2023; Petrov et al., 2023; Yamaguchi et al., 2024a,b; Han et al., 2025), where a sentence is split into spuriously many tokens, causing slower inference and more significant cost. Previous works rely on monolingual or parallel corpora to align source and target languages or focus on vocabulary overlaps between them (Minixhofer et al., 2022; Dobler and de Melo, 2023; Pham et al., 2024; Liu et al., 2024; Remy et al., 2024; Minixhofer et al., 2024; Han et al., 2025; Moroni et al., 2025). However, these methods face a significant challenge when dealing with low-resource

¹Code is available at https://github.com/sj-h4/dict_trans_tokenizer.

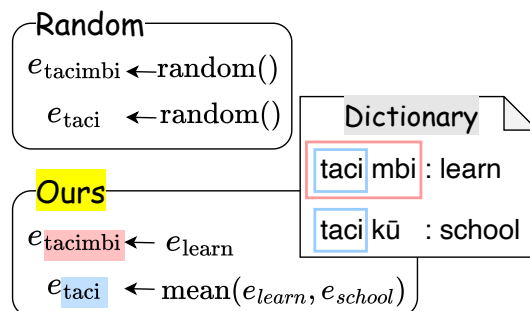


Figure 1: Conceptual illustration of our proposed method. Embeddings of target subwords are initialized using those of corresponding source subwords.

languages, where parallel corpora are often unavailable (Artetxe et al., 2017; Fang and Cohn, 2017), or when there is little lexical overlap between source and target languages that use different scripts.

Dictionaries offer a potential solution to this issue. When reading in an unfamiliar language, bilingual dictionaries can aid in comprehension by connecting words in the unknown language to their equivalents in a known language, and they can also enhance the performance of language models (Vasselli et al., 2025). Such dictionaries are often available, even when other linguistic resources are scarce, thanks to descriptive linguists, who produce them as part of their work in documenting languages (Adams et al., 2017).

In this light, we propose an embedding initialization method for cross-lingual vocabulary transfer based on bilingual dictionaries as illustrated in Figure 1. Our proposed method utilizes dictionaries to map as many target subwords as possible to source subwords for embedding initialization. This method is inspired by Trans-Tokenizer (Remy et al., 2024) that maps target subwords to source subwords through word alignment using parallel corpora. Our proposed method differs from Trans-Tokenizer in that we utilize dictionaries, which

are more available for low-resource languages and have helpful features for subword mapping, and we estimate the target embeddings iteratively. To perform iterative mapping, we utilize the BPE tokenizer algorithm (Sennrich et al., 2016), but progressively remove subwords, forcing the tokenizer to decompose words into shorter and shorter subwords. We estimate the embeddings for as many target subwords as possible by removing subwords mapped to source subwords from the vocabulary and reapplying tokenization. The experimental results demonstrate that our dictionary-based approach outperforms existing methods and multilingual language models in terms of F1 score on a downstream task and perplexity, across low-resource languages with different scripts. Our finding is that our dictionary-based method can improve the performance in low-resource languages that are genealogically distant from English and show typologically isolating or agglutinative characteristics (e.g., Uyghur, Khmer, and Manchu).

2 Background and Related Work

Cross-lingual vocabulary transfer consists of subword embedding initialization and language-adaptive pre-training.

Embedding Initialization. Embedding initialization is a fundamental component of cross-lingual vocabulary transfer. WECHSEL (Minixhofer et al., 2022) initializes target language subword embeddings using static embeddings obtained from monolingual corpora and bilingual dictionaries between source and target languages. In contrast, FOCUS (Dobler and de Melo, 2023) initializes target subword embeddings without bilingual dictionaries by leveraging subword overlap between source and target languages and static subword embeddings. UniBridge (Pham et al., 2024) explores vocabulary size optimization and initializes subword embeddings, considering both syntactic and semantic aspects. While it utilizes subword overlap for initialization, it also employs static embeddings from monolingual corpora to consider similarity for languages with rare subword overlap. Trans-Tokenizer (Remy et al., 2024) initializes target subword embeddings using word-level alignment from parallel corpora. ZeTT (Minixhofer et al., 2024) trains a hypernetwork that predicts the embeddings for target tokenizers with zero-shot. However, each approach has limitations for low-resource languages: WECHSEL requires substan-

tial monolingual corpora and bilingual dictionaries, FOCUS and UniBridge face challenges with languages using different scripts or having minimal subword overlap with the source language, and Trans-Tokenizer is not applicable to languages lacking parallel corpora with the source language.

Language Adaptive Pre-Training (LAPT). After embedding initialization, most approaches train the embeddings or all weights in a target model. This process is called Language-Adaptive Pretraining (LAPT) (Chau et al., 2020) or continuous pre-training. FOCUS, UniBridge, Trans-Tokenizer, and Yamaguchi et al. (2024b) initialize subword embeddings with each approach, followed by LAPT. UniBridge incorporates Masked Language Model (MLM) loss and KL divergence in its LAPT process, and others use MLM loss or Causal Language Model (CLM) loss. Trans-Tokenizer and Yamaguchi et al. (2024b) train the top and bottom two layers in CLMs with LoRA (Hu et al., 2022). Yamaguchi et al. (2024b) also demonstrates that continuous pre-training with multi-token prediction (Gloeckle et al., 2024) sometimes improves the performance and does not cause performance degradation.

3 Proposed Method

In this study, we propose a simple approach for low-resource languages utilizing bilingual dictionaries, which comprise of words (entries) in a target language and their definitions in a source language. In this approach, we suppose that source models are trained with high-resource languages and have tokenizers with pre-trained embeddings. The proposed method consists of three parts: training a tokenizer for the target language, aligning subwords between languages, and initializing target language subword embeddings.

3.1 Tokenizer Training

We train byte-level BPE tokenizers (Sennrich et al., 2016) using dictionary entries to obtain a tokenizer for the target language. We employ byte-level BPE tokenizers to ensure that no tokens are out of vocabulary, even when training on limited resources.

3.2 Subword Mapping

The subword mapping algorithm is illustrated in Algorithm 1. This process aims to maximize the number of target subwords that can be mapped to source

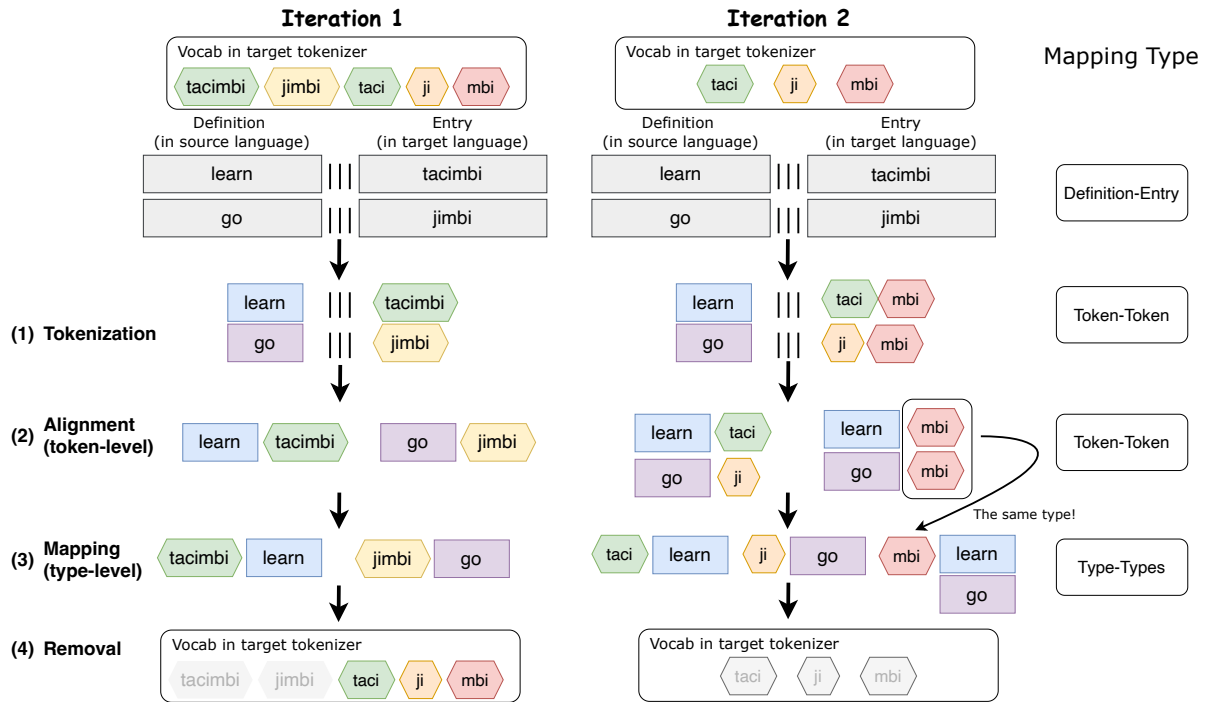


Figure 2: An illustration of the subword mapping part in the proposed method. We (1) tokenize definitions and entries, (2) align subwords using tokenized definition-entry pairs, (3) map a target subword to source subwords based on the alignment result, and (4) remove mapped target subwords from a target tokenizer. We repeat these processes until no target subwords are newly mapped. The target language is Manchu in this illustration, and “-mbi” is a vowel suffix.

subwords, reducing the number of unmapped subwords. Since unmapped target subwords are assigned the unknown token’s embedding from the source language, our goal is to minimize their occurrence by increasing the coverage of mapped target subwords as much as possible. This mapping process consists of four steps: tokenization, alignment, mapping, and removal, as illustrated in Figure 2.

(1) Tokenization. First, we tokenize dictionary entries and definitions using the trained target language tokenizer and the source model’s tokenizer, respectively (line 2 in Algorithm 1).

(2) Alignment. We treat the entry-definition pairs as a parallel corpus and employ `fast_align` (Dyer et al., 2013), which is based on the IBM translation models (Brown et al., 1993), IBM Model 2, as an alignment tool to obtain target-source subword pairs and their counts (line 3). In this step, the subwords are handled as tokens, not types.

(3) Mapping. Based on the target-source subword pairs obtained in step 2, we create a one-to-many mapping for all target language subwords at the type level (line 4), and then update the subword

Algorithm 1: Subword Mapping

Input: S : source tokenizer

T : target tokenizer

\mathcal{D} : dictionary (entry-definition pair)

\mathcal{M} : target-source subwords mappings (one-to-many) and the count for each mapping

Output: Target-Source one-many subword mapping

```

1 do
2    $\mathcal{C} \leftarrow \text{TokenizeCorpus}(S, T, \mathcal{D})$ ;
3    $\mathcal{A} \leftarrow \text{FastAlign}(\mathcal{C})$ ;
4    $\mathcal{M}_{\text{new}} \leftarrow \text{MapSubwords}(\mathcal{A})$ ;
5    $\mathcal{M} \leftarrow \text{UpdateMapping}(\mathcal{M}, \mathcal{M}_{\text{new}})$ ;
6    $T \leftarrow \text{DeleteTokens}(T, \mathcal{M}_{\text{new}})$ ;
7 while  $|\mathcal{M}_{\text{new}}| > 0$ ;
8 return  $\mathcal{M}$ ;

```

mapping (line 5). The update process includes updating target-source subword pairs at the type level and aggregating their count.

(4) Removal. After creating target-source subword mappings, we remove the mapped target sub-

words from the target tokenizer’s vocabulary and the merged pairs that contain the subwords (line 6). The aim is to assign source subwords to unmapped target subwords in the next loop. Since BPE tokenizers iteratively merge subwords into longer subwords, only the longest subwords are mapped by default, and their constituent subwords remain unmapped if this removal is skipped. Removing subwords enables the mapping of shorter subwords, as BPE tokenizers fall back to shorter segments when a subword is no longer available in the vocabulary.

We repeat these steps until no new subwords are mapped. We assign the UNK token from the source language model for target language subwords without corresponding source language subwords in the tokenizer’s vocabulary.

This subword mapping method is different from the Trans-Tokenizer. Our proposed method employs subword-level alignment because dictionaries are primarily composed of words, and the words are short. It also estimates mappings for shorter subwords, while the Trans-Tokenizer employs word-level alignment and estimates mapping heuristically.

3.3 Initializing Subword Embeddings

Let A_t be the set of source language subwords corresponding to a target language subword t , E_s^S and E_t^T be the sets of embeddings for a source language subword s and a target language subword t , respectively. Let $c(x, y)$ be the count of a pair of x and y . The relative counts of subword s in \mathcal{M}_t is defined as:

$$c(s|t) = \frac{c(s, t)}{\sum_{x \in \mathcal{M}_t} c(x, t)}$$

Using the embedding $e_s^S \in E^S$ of the source subword $s \in A_t$, the embedding $e_t^T \in E^T$ of the target subword t is initialized as a weighted average of corresponding source subwords embeddings²:

$$e_t^T = \sum_{s \in \mathcal{M}_t} c(s|t) \cdot e_s^S$$

We finally copy the embeddings of the special tokens, digits, and punctuations from the source model to the target model and then replace the embeddings in the source model with the initialized embeddings.

²Note that Llama 3 tokenizer does not have the UNK token, so we initialize the subword randomly.

Language	Script	Entries
German	Latin	101,997
Japanese	Kanji, Kana	25,969
Old English	Latin	7,592
Uyghur	Arabic	1,131
Sanskrit	Devanagari	5,282
Khmer	Khmer	5,656
Manchu	Latin	21,620

Table 1: The script and the number of entries in the dictionary for each language.

Language	FOCUS			Ours
	XLM	Llama 3.1	Gemma 2	
German	1,948,497	2,063,441	1,720,272	2,282,288
Japanese	2,320,596	2,554,873	2,275,944	4,291,810
Old English	398,508	416,885	327,382	660,516
Uyghur	1,440,550	1,582,594	1,359,842	3,656,931
Sanskrit	1,373,894	2,851,798	1,142,922	4,612,928
Khmer	1,983,542	4,203,481	1,955,756	7,589,786
Manchu	191,649	180,748	77,737	209,420

Table 2: The number of tokens in the data for LAPT for each language.

This embedding initialization method is different from FOCUS. FOCUS initializes target subword embeddings using source-target subword mapping obtained from overlapping subwords between source and target languages and static embeddings from FastText (Bojanowski et al., 2017). In contrast, our method uses the mapping trained with dictionaries.

4 Experimental Setups

Languages. We use English as a source language and German, Japanese, Old English, Uyghur, Sanskrit, Khmer, and Manchu as target languages. We treat German and Japanese as high-resource languages and other languages as low-resource languages. As source models, we use RoBERTa (base) (Liu et al., 2019), XLM-R (base) (Conneau et al., 2020), Llama 3.1 8B (Grattafiori et al., 2024), and Gemma 2 9B (Team et al., 2024). The detailed information about the models is provided in Appendix A

Evaluation Metrics. We evaluate the performance of our initialization method using F1 for the NER performance of a masked language model (RoBERTa) and perplexity for causal language models (Llama 3.1 and Gemma 2). The perplexity is normalized by word length, not subword length, for fair comparison across tokenizers. This is be-

	German	Japanese	Old English	Uyghur	Sanskrit	Khmer	Manchu
RoBERTa	<u>89.61</u>	<u>75.33</u>	62.39	38.73	<u>51.48</u>	27.58	73.52
XLM-R	90.27	81.28	37.59	28.30	48.85	34.78	65.32
FOCUS (XLM-R)	89.28	77.22	37.57	23.00	36.16	19.35	28.00
+ LAPT	90.00	77.46	37.57	37.16	12.33	12.33	28.03
Ours (RoBERTa)	76.99	71.93	45.43	36.06	44.23	42.21	94.87
+ LAPT	76.43	73.60	<u>52.71</u>	64.52	42.08	62.96	92.87
Ours (XLM-R)	75.98	72.66	35.10	25.07	37.53	36.55	<u>94.69</u>
+ LAPT	75.98	74.73	40.94	<u>59.41</u>	56.99	<u>58.37</u>	91.39

Table 3: Micro F1 scores of MLMs for WikiANN and ManNER.

cause comparing models with different tokenizers based on the perplexity normalized by subword length is unfair (Roh et al., 2020). We compute the word length using MeCab (Kudo et al., 2004) with unidic-lite³ for normalization for Japanese and tokenize by space for other languages. We use this perplexity for comparison between models within the same language, but not for comparison across different languages.

Datasets. As dictionary data, we use different resources for each language. For Manchu, we use data extracted from the Manchu–English dictionary (Norman, 2013)⁴. For German and Japanese, we employ the <target language>-English bilingual dictionary from MUSE (Lample et al., 2017) that is utilized in WECHSEL (Minixhofer et al., 2022)⁵. For Uyghur, Sanskrit, and Khmer, we use the <target language>-English dictionary from Wiktionary as employed in WECHSEL (Minixhofer et al., 2022)⁶. Note that the Manchu dictionary includes entries and definitions that consist of multiple words, whereas in the other dictionaries, each target language word is paired with a single word.

We use “Manwen Laodang”⁷ as a text corpus for initializing with FOCUS and training for LAPT for Manchu and Wikipedia dataset (Foundation)⁸ for other languages. For NER, we use ManNER (Lee et al., 2024) for Manchu and WikiANN (Pan et al., 2017; Rahimi et al., 2019) for other languages. Table 1 shows the writing system and the number of

entries in the dictionary for each language⁹.

Training. After embedding initialization, we conduct continuous pretraining with up to 3,000 samples for LAPT. The number of tokens for each model is shown in Table 2. For masked language models, we fine-tune all layers using the MLM loss. We also fine-tune both models with LAPT and models without LAPT for the downstream task. We train only the top and bottom two layers with LoRA (Hu et al., 2022) in the CLMs, following Remy et al. (2024) and Yamaguchi et al. (2024b). We also adopt multi-token prediction (Gloeckle et al., 2024) for training efficiency, which causes no significant performance degradation (Yamaguchi et al., 2024b). The hyperparameters are in Appendix A.

Baselines. We use FOCUS as a baseline for embedding initialization performance because this is a widely used baseline in vocabulary transfer studies (Remy et al., 2024; Yamaguchi et al., 2024b; Pham et al., 2024; Minixhofer et al., 2024). We train target tokenizers with up to 50,000 samples in the dataset. We initialize target subword embeddings with FOCUS. We use up to 50,000 samples in a dataset for initialization. We apply FOCUS to XLM-R instead of RoBERTa as an MLM baseline because FOCUS performs better for multilingual models than monolingual models. We also apply FOCUS to Llama 3.1 and Gemma 2 as CLM baselines.

5 Results and Discussion

Table 3 shows the results on NER. When the results of RoBERTa-based models were compared with those of the proposed method, the proposed method

³<https://github.com/polm/unidic-lite>

⁴<https://github.com/tyotakuki/manchuvocabdata>

⁵<https://github.com/facebookresearch/MUSE>

⁶<https://github.com/CPJKU/wechsel/tree/main/dicts>

⁷The Manwen Laodang data used in this study was collected in-house.

⁸<https://huggingface.co/datasets/wikimedia/wikipedia>. We use the 20231101 dump.

⁹Note that Manchu uses the Manchu alphabet, but the dictionary and dataset transcribe it in the Latin alphabet.

	German	Japanese	Old English	Uyghur	Sanskrit	Khmer	Manchu
Llama 3.1	<u>655.26</u>	7868.14	199411567.86	2.07×10^{24}	548192867.74	inf	2.30×10^{19}
FOCUS	13275123.06	58188456.24	1.03×10^{10}	4.36×10^{22}	2.99×10^{20}	inf	1893791141.42
+ LAPT	2375.69	574.71	57810645.31	7.53×10^{20}	12376253.13	inf	1288173.19
Ours	444876.62	<u>473.72</u>	<u>46180.42</u>	<u>18053.31</u>	<u>1503.39</u>	<u>64508.22</u>	<u>144818.36</u>
+ LAPT	88.61	4.42	406.53	168.43	3.56	4.32	502.02
Gemma2	<u>2013.66</u>	494072.35	3.36×10^{11}	3.98×10^{27}	1.02×10^{10}	inf	1.06×10^{16}
FOCUS	2.83×10^{11}	993607.28	5.17×10^{13}	1.71×10^{30}	5.09×10^{14}	inf	1025102.29
+ LAPT	14045.20	<u>356.75</u>	57810645.31	2.89×10^{23}	144650900.73	inf	42687.69
Ours	6802328.30	1340.02	<u>4685467.63</u>	<u>3163709.28</u>	<u>97101.92</u>	<u>652351.91</u>	4848206.72
+ LAPT	595.14	5.83	1324.49	83.32	10.57	16.80	509.64

Table 4: Perplexity of CLMs. The perplexity is normalized by word length, not subword length, for fair comparison across different tokenizers.

achieved better performance for Uyghur, Khmer, Sanskrit, and Manchu.

Table 4 shows the perplexity for the base and transferred models of Llama 3.1 and Gemma 2. This result indicates that the performance of the proposed method with LAPT achieves the best performance. Figure 3 illustrates the perplexity distribution of the Llama 3.1-based model for German and Manchu using our proposed method. The distributions of other languages are illustrated in Figure 6 in Appendix B.2. There are some outliers regardless of whether LAPT is performed or not, suggesting out-of-distribution (OOD) issues occur. We also visualize the embeddings in Appendix B.1.

5.1 Data Size

Table 5 shows the number of words of target languages in the data used for embedding initialization. The results shown in Tables 3, 4, and 5 demonstrate that our proposed method achieves better performance than FOCUS in the low-resource languages, with less than 10% of the data, highlighting the efficiency of the proposed method.

On the other hand, FOCUS performs better than our proposed method for high-resource languages, German and Japanese. This indicates that FOCUS can effectively initialize subword embeddings if a sufficient amount of data is available, while our proposed method can initialize subword embeddings with a limited amount of data.

5.2 Mapped Subwords Coverage

Table 6 shows the trained tokenizers’ vocabulary size and the number of mapped subwords that were transferred from Llama 3.1 to target languages. The coverage of mapped tokens varies by language and can be categorized into two groups: Japanese and Sanskrit, as well as others. Japanese and Sanskrit

Language	FOCUS	Ours
German	21,582,818	101,997
Japanese	3,118,885	25,969
Old English	353,515	7,592
Uyghur	2,771,058	1,131
Sanskrit	2,812,121	5,282
Khmer	1,937,229	5,656
Manchu	397,659	21,620

Table 5: The number of words in the data of target languages for embedding initialization. We count strings separated by spaces as a word.

Language	Tokenizer size	Mapped tokens
German	50,265	43,656 (86.85 %)
Japanese	33,449	32,043(95.80 %)
Old English	11,427	10,059 (88.03 %)
Uyghur	2,693	2,372 (88.08 %)
Sanskrit	1,719	1,672 (97.27 %)
Khmer	1,634	1,465 (89.67 %)
Manchu	20,578	15,918 (77.35 %)

Table 6: The vocabulary size in tokenizers and the number of mapped tokens during transferring Llama 3.1.

have relatively regular stem changes. On the other hand, German has complex inflection, and Uyghur and Manchu have vowel harmony, which makes tokenization inefficient. These linguistic features cause the coverage of mapped tokens to be lower.

This result indicates that the Llama 3.1-based model performs better with higher mapped subword coverage for the tokenizer size. The results of Gemma 2 also demonstrate that unmapped tokens cause performance degradation.

Language	RoBERTa	XLM-R	Llama 3.1	Gemma 2	FOCUS			Ours
					XLM	Llama 3.1	Gemma 2	
German	3,686,547	2,426,841	2,983,799	2,598,287	2,069,545	2,198,015	1,827,442	2,429,275
Japanese	7,720,302	3,616,523	4,129,765	3,608,275	2,479,331	2,735,561	2,431,308	4,585,446
Old English	100,320	83,383	93,820	86,062	46,263	47,782	37,764	75,784
Uyghur	3,482,499	837,816	2,189,895	1,750,318	472,816	525,931	455,439	1,264,060
Sanskrit	4,482,539	946,657	1,541,635	1,343,833	592,570	1,230,733	497,080	19,974,128
Khmer	9,987,789	1,318,386	5,773,353	3,533,035	736,043	1,674,177	726,099	3,100,085
Manchu	104,660	87,861	95,225	85,902	52,104	49,198	21,111	56,912

Table 7: The number of subwords in the test data for perplexity for each language. Note that the same tokenizer, trained on dictionary entries, was used for each language for our proposed method during experiments.

Base Model	RoBERTa				Llama 3.1				Gemma 2			
	iter 1	iter2	iter 3	iter 4	iter 1	iter2	iter 3	iter 4	iter 1	iter2	iter 3	iter 4
German	43,591	62	3	0	44,134	63	1	0	44,652	63	1	0
Japanese	21,003	10,987	34	0	21,003	11,008	34	0	21,002	11,013	28	0
Old English	7,520	2,503	17	10	7,523	2,508	17	10	7,522	2,508	17	10
Uyghur	1,152	1,079	109	27	1,152	1,079	109	27	1,152	1,079	109	27
Sanskrit	1,575	90	4	2	1,575	93	3	0	1,575	94	3	0
Khmer	1,370	77	3	0	1,385	78	2	0	1,384	76	3	0
Manchu	15,053	846	4	0	15,066	848	4	0	15,068	851	4	0

Table 8: The number of mapped subwords in the iteration until the fourth iteration.

5.3 Token Efficiency

Table 7 shows that our approach reduces the number of subwords for low-resource languages compared to Llama 3.1, although FOCUS achieves an even greater reduction. Our proposed method trains a dedicated tokenizer for the target languages compared with the Llama 3.1 tokenizer. As a result, our tokenization is more efficient than that of a multilingual tokenizer. This result demonstrates that our approach can address token over-fragmentation, although performance improvement is limited to several languages.

5.4 Multilingual Model vs. Monolingual Model as a Source Model

Table 3 shows the micro F1 scores of RoBERTa-based models and XLM-R-based models. This result indicates that our proposed method works better with a monolingual model for languages except Japanese and Sanskrit than with a multilingual model. Our proposed method uses only English information to estimate the embeddings of target subwords. Thus, monolingual models are suitable.

5.5 Language Characteristics

A common characteristic among the languages that saw the most benefit from our approach—Uyghur,

Khmer, and Manchu—is that they are genealogically distant from English. Additionally, Uyghur and Khmer use different writing systems from English. Although this difference can limit the RoBERTa and XLM-R performance, our proposed method improves performance.

However, Japanese is also genealogically distant from English and uses a different script, yet RoBERTa outperformed the proposed method. A possible reason for this is the large number of character types in Japanese. When a tokenizer is trained on limited data, the wide variety of characters can prevent subwords from achieving sufficiently robust representations. This limitation likely contributed to the constrained performance of our proposed method in Japanese.

Table 4 indicates that our proposed method can handle languages with any script, thanks to language-specific tokenizers.

5.6 Morphological Typological Features

Our proposed method utilizes bilingual dictionaries, but these dictionaries generally only contain a single word form, excluding conjugated or inflected variants. This feature might also make it challenging to handle inflection or particles. We analyze the results with respect to two morpholog-

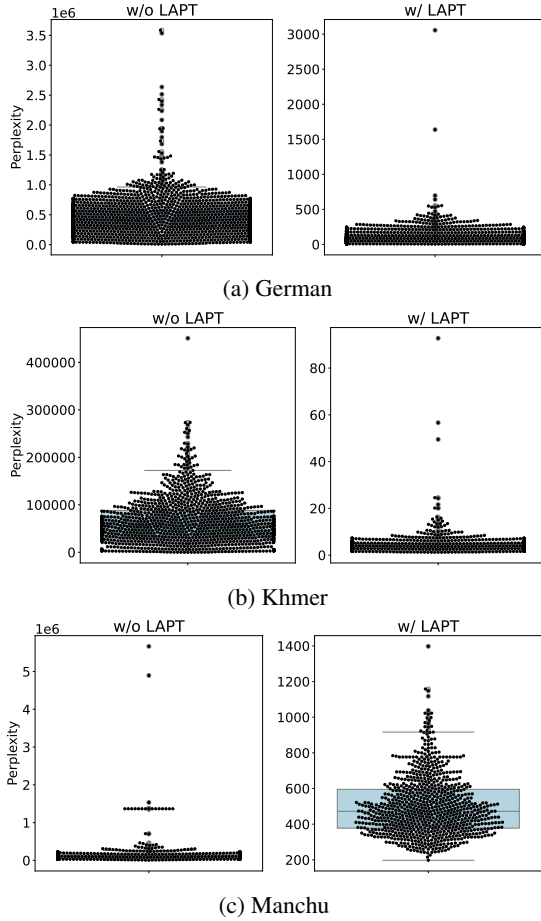


Figure 3: The perplexity distribution and box plot of Llama 3.1-based our proposed method.

ical types: isolating-synthetic and agglutinative-fusional (Whaley, 1997).

Synthesis. In languages used in the experiments, Khmer is isolating, German is relatively synthetic, and others are synthetic. Since isolating languages do not have inflectional changes, and words are generally used in the form listed in the dictionaries, such languages benefit from our proposed method shown in Table 3.

Fusion. German, Old English, and Sanskrit have stronger fusional (inflected) characteristics, and Japanese, Uyghur, and Manchu have stronger agglutinative characteristics. In fusional languages, affixes carry multiple grammatical functions, requiring more training data to capture these functions than in agglutinative languages, where each affix corresponds to a single grammatical function. As a result, the performance improvement by our proposed method for German, Old English, and Sanskrit is limited, as shown in Table 3.

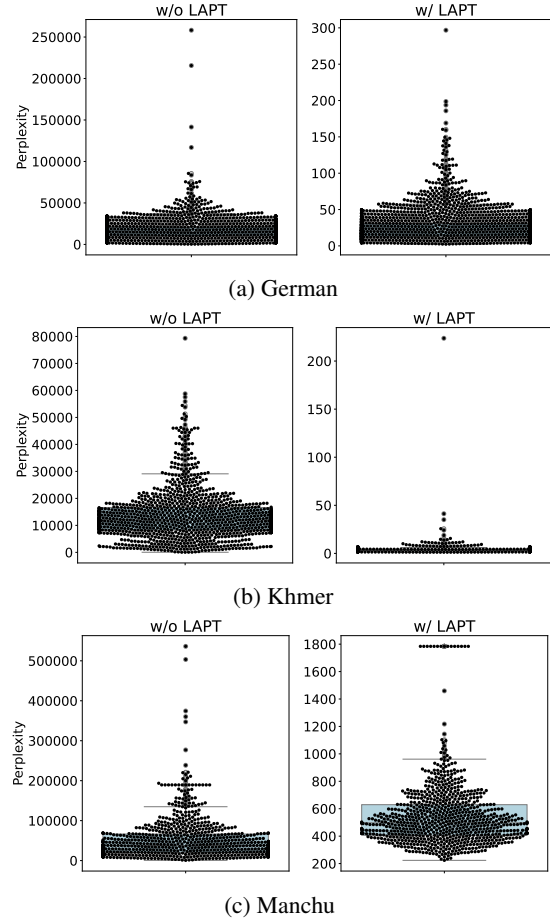


Figure 4: The perplexity distribution and box plot of Llama 3.1-based our proposed method without the removal step.

5.7 Ablation Study

LAPT. LAPT improves NER performance for Japanese, Old English, Uyghur, and Khmer, while it has challenges for German, Sanskrit, and Manchu, as shown in Table 3. Considering that we fine-tune the models for NER, this suggests that LAPT plays an insignificant role. This result is because the word order is insufficient for MLMs, and we also fine-tuned the model for NER. On the other hand, LAPT improves the perplexity for all languages, as shown in Table 4. This result suggests that LAPT tunes the model for target languages by incorporating word order, as the subword initialization does not consider word order. On the other hand, Figure 3 implies that there might be out-of-distribution samples with limited training data.

Mapped Subwords during Transferring Table 8 shows the number of mapped subwords in each iteration. The number remains almost the

	German	Japanese	Old English	Uyghur	Sanskrit	Khmer	Manchu
Ours (w/o LAPT)	76.99	71.93	45.43	36.06	44.23	42.21	94.87
- Removal	74.48	55.30	46.83	31.16	43.81	37.50	95.25
Ours (w/ LAPT)	76.43	73.60	52.71	64.52	42.08	62.96	92.87
- Removal	59.96	29.70	29.75	49.89	53.70	57.27	41.58

Table 9: Micro F1 scores of MLMs for WikiANN and ManNER of RoBERTa-based models when skipping the (4) Removing step in our proposed method, finishing the subword mapping at the first iteration.

	German	Japanese	Old English	Uyghur	Sanskrit	Khmer	Manchu
Llama 3.1							
Ours (w/o LAPT)	444,876.62	473.72	46180.42	18,053.31	1,503.39	64,508.22	144,818.36
- Removal	16,709.97	19.73	17,395.35	7,126.41	3,404.88	13,652.20	52,778.58
Ours (w/ LAPT)	88.61	4.42	406.53	168.43	3.56	4.32	502.02
- Removal	25.00	2.98	28.25	55.62	3.07	3.67	549.36
Gemma 2							
Ours (w/o LAPT)	6,802,328.30	1,340.02	4,685,467.63	3,163,709.28	97,101.92	652,351.91	4,848,206.72
- Removal	9,314.48	131.73	50,985.49	51,610.23	35,789.10	5,966.91	403,457.47
Ours (w/ LAPT)	595.14	5.83	1,324.49	83.32	10.57	16.80	509.64
- Removal	18.98	5.98	37.22	1,650,481.13	6.91	3.71	1,238.87

Table 10: The perplexity when skipping the (4) Removing step in our proposed method and finishing the subword mapping at the first iteration.

same across different models, indicating that iterative mapping is robust across base models’ tokenizers for the same language.

Subword Removal. Table 9 and Table 10 show the results of the ablation study about the mapped subword removal step in our proposed method. Figure 4 shows that OOV issues also occur without the removal. The distribution of other languages and analysis are in Appendix B.2. These results indicate that the removal of mapped subwords and iteration play a crucial role in MLMs, specifically for LAPT. On the other hand, the removal and iteration negatively affect CLMs, except in Japanese and Manchu. The number of mapped subwords after the second loop varies across languages, as shown in Table 8. For MLMs, the number of mapped subwords is essential, so the removal works well. For example, approximately only 60% of subwords in the vocabulary of the Japanese tokenizer are mapped in the first loop, which causes the performance decrease. These results also align with the analysis in Section 5.2. The exacerbation of perplexity indicates that the shorter subword mapping is noise for CLMs. However, skipping the removal sometimes causes inefficient fine-tuning, as in Uyghur. We conclude that removal can improve MLM performance and enhance the model’s

ability to handle extremely low-resource languages that are not included in the training data of pre-trained models.

6 Conclusion

In this study, we propose a dictionary-based cross-lingual vocabulary transfer approach. This method utilizes bilingual dictionaries, which are available even for low-resource languages. It iteratively maps a subword in a target language to subwords in a source language and initializes the target subword embeddings. The experimental results demonstrate that our approach effectively initializes the subword embeddings for low-resource languages, indicating it is a promising approach for building language models for those languages.

Limitations

Dictionaries and Languages. Dictionaries have only the unmarked form and not the inflected form in general, suggesting that handling inflection is challenging for our dictionary-based method. Additionally, some online dictionaries (e.g., MUSE) suffer from quality issues (Kementchedjieva et al., 2019). However, as this study demonstrates, the dictionary-based method performs better for low-resource languages than the existing method we

used as a baseline. This indicates that dictionaries are helpful for cross-lingual vocabulary transfer for low-resource languages despite the lack of grammatical information.

Language Set. In the experiments, we used seven languages: German, Japanese, Old English, Uyghur, Sanskrit, Khmer, and Manchu. The number of selected languages is limited, and there are other languages with different linguistic features to test. However, we include the language, Manchu, whose resources are less available on the Internet. This means that we tested the language in which the training data of language models contains less data, which is valuable for low-resource languages.

Generalizability. Our proposed method shows challenges for several languages and might raise questions about the generalizability. However, it is valuable if the method improves the performance of languages that face challenges with existing methods.

Evaluation. We evaluated CLMs by perplexity, which might raise questions about fair comparison when we compare a model with other models with different tokenizers. The performance of downstream tasks can be used to evaluate CLMs (Yamaguchi et al., 2024b; Minixhofer et al., 2024; Han et al., 2025). However, we aim to apply our proposed method to low-resource languages, including extremely low-resource languages (e.g., Manchu), and the downstream tasks’ datasets for CLMs are not available for those languages. Thus, we use the perplexity normalized by word length, not subword length, for fair comparison.

Performance Improvement. Section 5 shows that the performance improvement is limited with our proposed method for some languages. There is room for performance improvement by refining the alignment method or hyperparameter tuning. However, the limited improvement is a known problem in cross-lingual vocabulary transfer for low-resource languages (Yamaguchi et al., 2024b). We aim to demonstrate the potential of a dictionary-based approach for cross-lingual vocabulary transfer in low-resource languages, and performance improvement will be the subject of a future study.

Model Expansion Instead of Replacement. In this study, we replace the source vocabulary with the target vocabulary instead of extending the source vocabulary. Vocabulary expansion is some-

times needed for multilingualism. However, vocabulary replacement performs better than expansion for target languages (Dobler and de Melo, 2023; Yamaguchi et al., 2024a), and sometimes performs better even in a source language (Yamaguchi et al., 2024a). In this study, we incorporate vocabulary replacement for this reason.

Tokenization. Since the trained tokenizer using dictionary entries has meaningful tokens in our proposed method, morpheme-aware tokenization (Libovický and Helcl, 2024; Asgari et al., 2025) can be helpful. Other tokenizers can also be used, such as SentencePiece (Kudo and Richardson, 2018). However, in this study, we trained BPE tokenizers, which are widely used tokenizer types, to compare with existing models.

Ethical Considerations

Dataset. In this study, we use bilingual dictionaries and corpora from several sources. We follow the given licenses for each dataset. These datasets might contain identical or sensitive information, even though they are used widely. The model’s behavior in this study should be assessed before releasing it as an application.

Use of AI Assistants. In this study, we have used GitHub Copilot for coding support.

Acknowledgement

We thank the anonymous reviewers for their valuable comments and suggestions. This project is partly supported by the Gemma Academic Program.

References

- Oliver Adams, Adam Makarucha, Graham Neubig, Steven Bird, and Trevor Cohn. 2017. [Cross-lingual word embeddings for low-resource language modeling](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 937–947, Valencia, Spain. Association for Computational Linguistics.
- Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. [Do all languages cost the same? tokenization in the era of commercial language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

- Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. [Learning bilingual word embeddings with \(almost\) no bilingual data](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.
- Ehsaneddin Asgari, Yassine El Kheir, and Mohammad Ali Sadraei Javaheri. 2025. [Morphbpe: A morpho-aware tokenizer bridging linguistic complexity for efficient llm training across morphologies](#). *Preprint*, arXiv:2502.00894.
- 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.
- Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. [The mathematics of statistical machine translation: Parameter estimation](#). *Computational Linguistics*, 19(2):263–311.
- 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.
- Konstantin Dobler and Gerard de Melo. 2023. [FOCUS: Effective embedding initialization for monolingual specialization of multilingual models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13440–13454, Singapore. Association for Computational Linguistics.
- 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.
- Meng Fang and Trevor Cohn. 2017. [Model transfer for tagging low-resource languages using a bilingual dictionary](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 587–593, Vancouver, Canada. Association for Computational Linguistics.
- Wikimedia Foundation. [Wikimedia downloads](#).
- Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Roziere, David Lopez-Paz, and Gabriel Synnaeve. 2024. [Better & faster large language models via multi-token prediction](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 15706–15734. PMLR.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- HyoJung Han, Akiko Eriguchi, Haoran Xu, Hieu Hoang, Marine Carpuat, and Huda Khayrallah. 2025. [Adapters for altering LLM vocabularies: What languages benefit the most?](#) In *The Thirteenth International Conference on Learning Representations*.
- Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Yova Kementchedjhiya, Mareike Hartmann, and Anders Søgaard. 2019. [Lost in evaluation: Misleading benchmarks for bilingual dictionary induction](#). 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 3336–3341, Hong Kong, China. Association for Computational Linguistics.
- Taku Kudo and John Richardson. 2018. [SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. [Applying conditional random fields to Japanese morphological analysis](#). In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237, Barcelona, Spain. Association for Computational Linguistics.
- Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc’Aurelio Ranzato. 2017. [Unsupervised machine translation using monolingual corpora only](#). *Preprint*, arXiv:1711.00043.
- Sangah Lee, Sungjoo Byun, Jean Seo, and Minha Kang. 2024. [ManNER & ManPOS: Pioneering NLP for endangered Manchu language](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11030–11039, Torino, Italia. ELRA and ICCL.

- Jindřich Libovický and Jindřich Helcl. 2024. [Lexically grounded subword segmentation](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7403–7420, Miami, Florida, USA. Association for Computational Linguistics.
- Yihong Liu, Peiqin Lin, Mingyang Wang, and Hinrich Schuetze. 2024. [OFA: A framework of initializing unseen subword embeddings for efficient large-scale multilingual continued pretraining](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1067–1097, Mexico City, Mexico. Association for Computational Linguistics.
- 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](#). *Preprint*, arXiv:1907.11692.
- Benjamin Minixhofer, Fabian Paischer, and Navid Reksabsaz. 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 Minixhofer, Edoardo M. Ponti, and Ivan Vulić. 2024. [Zero-shot tokenizer transfer](#). In *Advances in Neural Information Processing Systems*, volume 37, pages 46791–46818. Curran Associates, Inc.
- Luca Moroni, Giovanni Puccetti, Pere-Lluís Hugué Cabot, Andrei Stefan Bejgu, Alessio Miaschi, Edoardo Barba, Felice Dell’Orletta, Andrea Esuli, and Roberto Navigli. 2025. [Optimizing LLMs for Italian: Reducing token fertility and enhancing efficiency through vocabulary adaptation](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6646–6660, Albuquerque, New Mexico. Association for Computational Linguistics.
- Jerry Norman. 2013. *A comprehensive Manchu-English dictionary*. Harvard-Yenching Institute Monograph Series. Harvard University, Asia Center, Cambridge, MA.
- 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.
- Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2023. [Language model tokenizers introduce unfairness between languages](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Trinh Pham, Khoi Le, and Anh Tuan Luu. 2024. [UniB-ridge: A unified approach to cross-lingual transfer learning for low-resource languages](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3168–3184, Bangkok, Thailand. Association for Computational Linguistics.
- Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. [Massively multilingual transfer for NER](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.
- François Remy, Pieter Delobelle, Hayastan Avetisyan, Alfiya Khabibullina, Miryam de Lhoneux, and Thomas Demeester. 2024. [Trans-tokenization and cross-lingual vocabulary transfers: Language adaptation of LLMs for low-resource NLP](#). In *First Conference on Language Modeling*.
- Jihyeon Roh, Sang-Hoon Oh, and Soo-Young Lee. 2020. [Unigram-normalized perplexity as a language model performance measure with different vocabulary sizes](#). *Preprint*, arXiv:2011.13220.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, and 179 others. 2024. [Gemma 2: Improving open language models at a practical size](#). *Preprint*, arXiv:2408.00118.
- Laurens van der Maaten and Geoffrey Hinton. 2008. [Visualizing data using t-sne](#). *Journal of Machine Learning Research*, 9(86):2579–2605.
- Justin Vasselli, Haruki Sakajo, Arturo Martínez Peguero, Frederikus Hudi, and Taro Watanabe. 2025. [Leveraging dictionaries and grammar rules for the creation of educational materials for indigenous languages](#). In *Proceedings of the Fifth Workshop on NLP for Indigenous Languages of the Americas (Americas-NLP)*, pages 112–118, Albuquerque, New Mexico. 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.
- L.J. Whaley. 1997. *Introduction to Typology: The Unity and Diversity of Language*. SAGE Publications.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Atsuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. 2024a. [An empirical study on cross-lingual vocabulary adaptation for efficient language model inference](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 6760–6785, Miami, Florida, USA. Association for Computational Linguistics.

Atsuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. 2024b. [How can we effectively expand the vocabulary of llms with 0.01gb of target language text?](#) *Preprint*, arXiv:2406.11477.

A Details of Experimental Setups

We use the Hugging Face Transformers library (Wolf et al., 2020) to utilize the language models. Table 11 lists the models used in this study along with their corresponding Hugging Face IDs. The hyperparameters used in the experiments are in Tables 12 and 13. We execute the EM algorithm seven times with fast_align. We use a single NVIDIA GeForce RTX 3090 GPU for training MLMs and inference of MLMs and CLMs. We also use eight NVIDIA L4 GPUs, a single NVIDIA A100-SXM4-40GB GPU, or one or two NVIDIA RTX A6000 or NVIDIA RTX 6000 Ada Generation GPUs for training of CLMs.

B Details of the Results

B.1 Visualization

Figure 5 visualizes the subword embedding with t-SNE (van der Maaten and Hinton, 2008) for Manchu, emphasizing several subwords’ points. This figure demonstrates that the subwords that have similar meanings or concepts to each other are positioned close to each other.

B.2 Removal and Perplexity Distribution

Figure 6 illustrates the perplexity distribution. There are some outliers even after the LAPT was performed. Figure 7 illustrates the perplexity distribution without the removal step. The distribution is similar to that with the removal step.

Language Models	HuggingFace ID
RoBERTa (base)	FacebookAI/roberta-base
XLNet (base)	FacebookAI/xlnet-roberta-base
Llama 3.1 8B	meta-llama/Llama-3.1-8B
Gemma 2 9B	google/gemma-2-9b

Table 11: Lists of the models with Hugging Face IDs we used in this study.

Parameter	Value
Batch size	8, 16
Epochs	2, 50
Sequence length	256, 512
Learning rate	1×10^{-4}
Learning rate scheduler	cosine
Adam β_1	0.9
Adam β_2	0.999
Adam ϵ	1×10^{-8}
Precision	bf16
Seed	42

Table 12: Hyperparameters for LAPT. We train CLMs with batch size 8, 2 epochs, and sequence length 512, and MLMs with batch size 16, 50 epochs, and sequence length 256.

Parameter	Value
Batch size	64
Epoch	25
Learning rate	5e-5
Learning rate scheduler	linear
Adam β_1	0.9
Adam β_2	0.999
Adam ϵ	1×10^{-8}
Seed	42

Table 13: Hyperparameters for task adaptation.

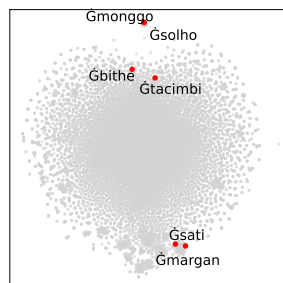


Figure 5: The subword embedding visualization with t-SNE for Manchu after transferring from RoBERTa without LAPT. “monggo”, “slho”, “bithé”, “tacimbi”, “sati”, “margan” mean “Mongol”, “Korea”, “book”, “lern”, “bear”, and “deer” respectively.

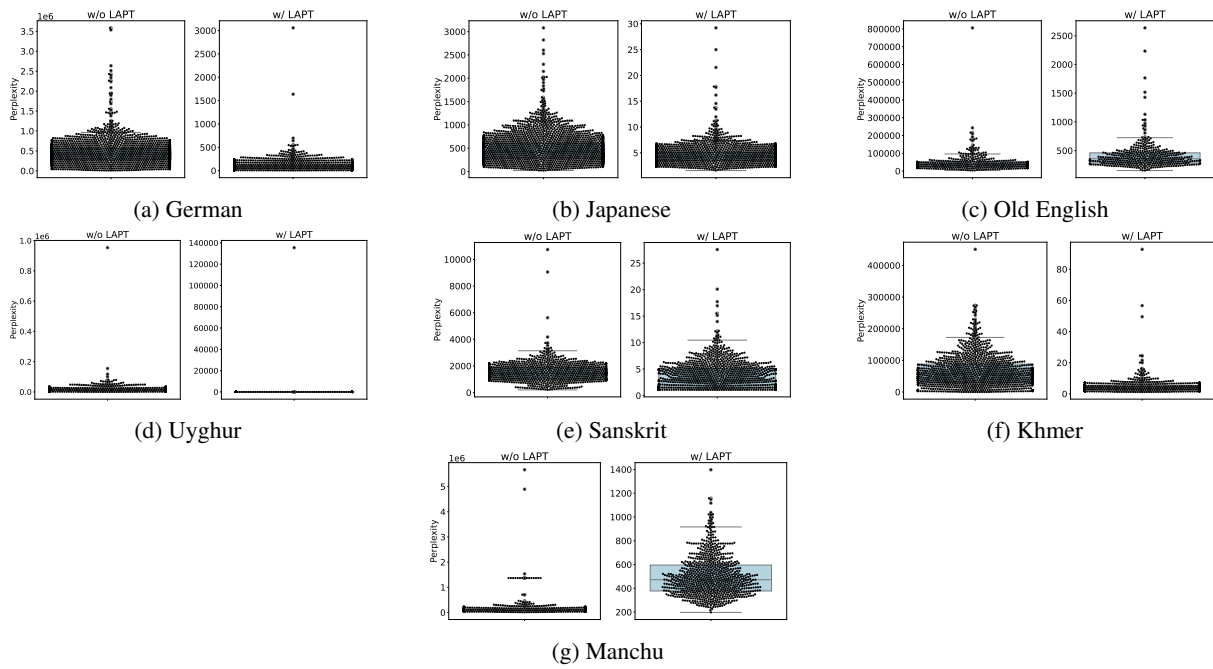


Figure 6: The perplexity distribution and box plot of Llama 3.1-based our proposed method.

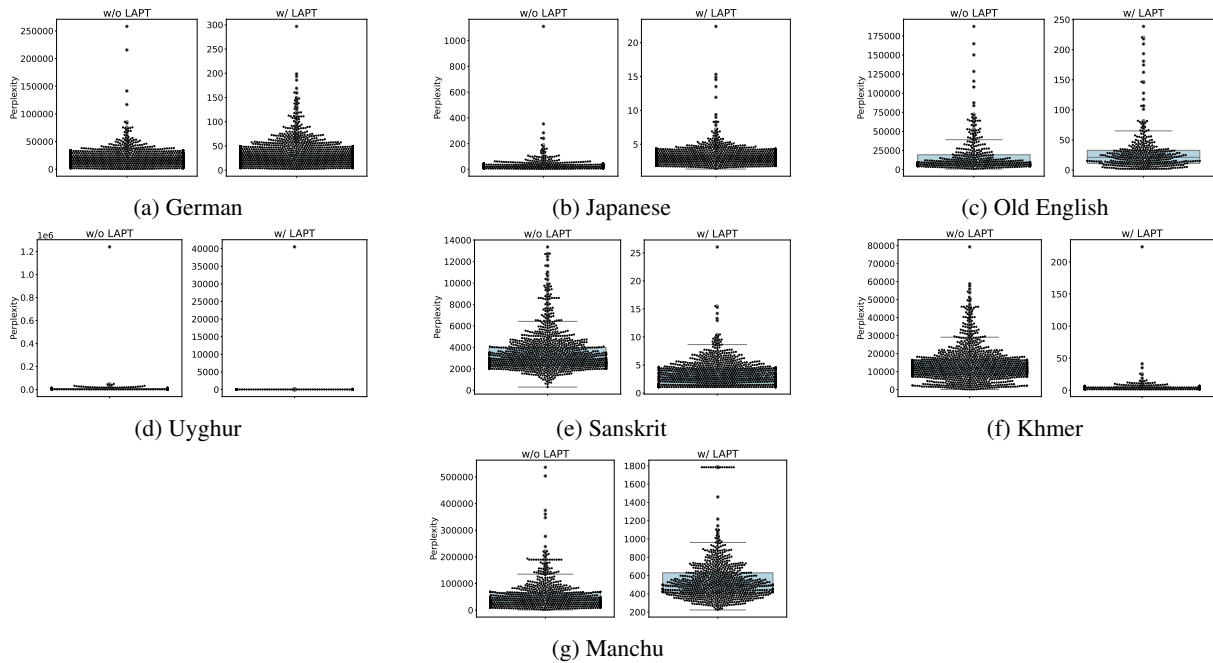


Figure 7: The perplexity distribution and box plot of Llama 3.1-based our proposed method without the removal.