UNIFYING VOCABULARY OF LARGE LANGUAGE MODEL WITH STATISTICAL TOKEN-LEVEL ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) achieve great success across many general tasks, but the mismatch among different vocabularies hinders further applications like token-level distillation and inference with various models. To align the vocabularies of LLMs, we propose a simple yet effective method named **UnifyVocab** to replace the vocabulary of an LLM at a limited cost. A new vocabulary alignment method is devised first to align the source vocabulary to the target one. We then rearrange the corresponding parameters like embeddings, and progressively fine-tune the model. Experimental results on models across multiple parameter scales demonstrate the effectiveness and generalization of UnifyVocab, which costs as few as 10B tokens to recover 98.02% performance of the vanilla models on average. We further find that unifying the vocabularies significantly facilitates the token-level distillation which remarkably boosts (+4.4%) the model with only 235M tokens. Moreover, our method provides a better initialization of multilingual vocabulary for LLMs to adapt to new languages.

1 INTRODUCTION

Large language models like LLaMA, GPT-4, and Qwen (Touvron et al., 2023b; OpenAI, 2023; Qwen, 2024) show impressive general abilities. These models have specific strengths and weaknesses, which arise from their pre-training corpus and method. However, the mismatch among their vocabularies impedes the deep knowledge transfer between these models like token-level distillation and ensemble. Thus, it is important to unify the vocabulary of the large language model at a low cost.

The vocabulary of the language model is kept unchanged after pre-training unless adapted to a new language. It is common to append new tokens to improve the effectiveness of encoding on a new language (Tran, 2020; Wang et al., 2020; Chau et al., 2020; Minixhofer et al., 2022; Cui et al., 2023; Liu et al., 2024).

In this paper, we introduce a method called **UnifyVocab** to replace the vocabulary of large language models from a view of token-token co-occurrences. As the general process to train an LLM, the pre-training corpus is first tokenized into token IDs, and then input into the model. Given the same pre-training corpus, different tokenizers result in various sequences of token IDs, while the semantic and syntactic information is preserved in the token-token co-occurrence. Therefore, UnifyVocab strives to align the token IDs from the original vocabulary and the target ones based on the global token-token co-occurrence matrix (Pennington et al., 2014). We further propose a metric to evaluate the performance of the token-token alignment matrix. The new embedding and language modeling head of LLMs ("lm_head" in the transformers (Wolf, 2019)) are initialized from the re-arranged parameters using the learned alignment matrix. Further adaptation process for the new vocabulary is divided into a progressive two-stage procedure to improve the stability of convergence.

Given a target vocabulary for substitution, results on models across different scales show that as few as 10B tokens are needed for our method to recover 98.02% performance of vanilla models on average. The training process of UnifyVocab is 1.92x faster than the best baseline method. Unifying vocabulary further facilitates the token-level distillation between models, which is 4.4% better than the sentence-level distillation on the same corpus. In addition, the model trained on the English corpus obtains a good initialization for the multilingual vocabulary, decreasing the perplexity from $2.9e^5$ to $2e^2$, and could adapt to new languages with only 4B tokens using UnifyVocab.

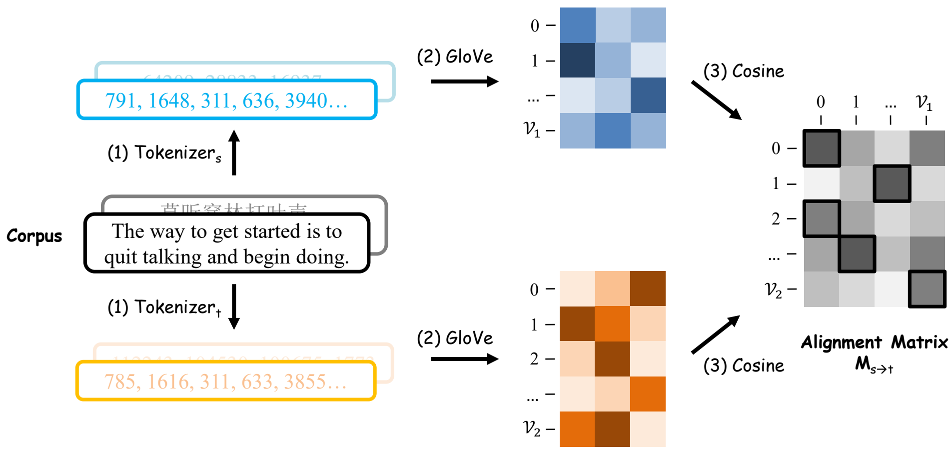


Figure 1: Illustration of UnifyVocab to align the token IDs from different vocabularies. We train token representations on the tokenized corpus, and align token IDs by the cosine similarity. It is noted that the IDs of tokens belonging to both vocabularies are directly replaced without alignment.

To sum up, our contributions are as follows:

- We propose a general method to align token IDs between two vocabularies and replace the vocabulary of the large language model from the token-token co-occurrence view, which costs as few as 10B tokens in the new vocabulary adaptation.
- We introduce a metric to evaluate the performance of token-level alignment, which is found proportional to the initial loss of pre-training.
- Experimental results show that our method promotes deep knowledge transfer between models like token-level distillation, and even the cross-lingual knowledge transfer among multiple languages.

2 UnifyVocab

2.1 Vocabulary Alignment

As shown in Figure 1, there are three steps in UnifyVocab to align two vocabularies of language models from the token-token co-occurrence information. We denote the source tokenizer as Tokenizer$_s$, which has $\mathcal{V}_s$ tokens, and the target tokenizer as Tokenizer$_t$ with $\mathcal{V}_t$ tokens, correspondingly.

Step 1: Tokenization The comprehensiveness of the pre-training corpus is important to obtain a well-trained token representation. An unbalanced corpus makes it hard to train the representation of tokens in the tail of vocabulary. Thus, the corpus used in this work is empirically composed of multilingual corpus CulturaX[40%] (Nguyen et al., 2023), code corpus The Stack[30%] (Kocetkov et al., 2023), and math corpus Proof-Pile-2[30%] (Azerbayev et al., 2024). We tokenize the mixed corpus using various tokenizers of different LLMs, and obtain multiple sequences of token IDs for the same corpus. The default token amount of corpus used in this step is 1B, which is investigated in Appendix B.1.

Step 2: Token Representation Learning We adopt GloVe (Pennington et al., 2014) to train the representation of the tokens from Step 1. The main reason is that GloVe considers more global statistical information than those slide window methods like CBOW and fastText (Mikolov et al., 2013a;b; Bojanowski et al., 2017). The details of training settings for GloVe vectors are reported in Appendix A.

Step 3: Token Alignment Based on the assumption that token representations capture the semantic information in the token, we align token IDs using the pair-wise cosine similarity of learned token representations. It should be noted that the ID of tokens belonging to both vocabularies are directly

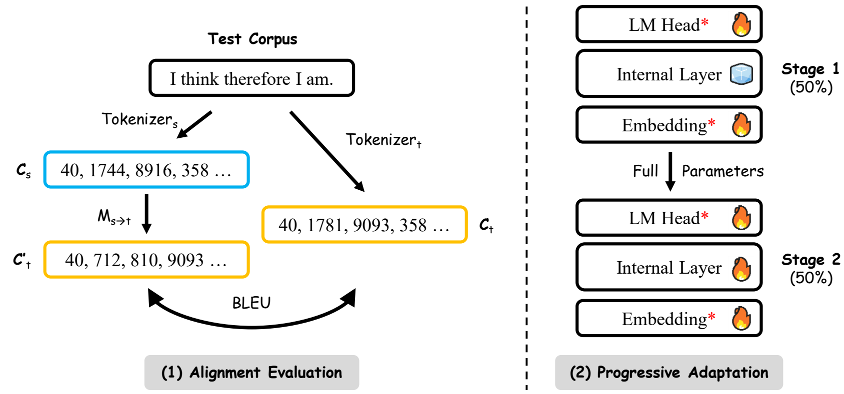


Figure 2: (1) We choose BLEU to evaluate the performance of alignment matrix $M_{s\to t}$ (2) The embedding and lm_head are tuned at the first half part of the tuning process, which follows the full parameter tuning. * indicates the parameter of each target token is initialized from the one of the most similar source token by alignment matrix $M_{s\to t}$.

replaced without the need to align. We denote the token-token alignment matrix $M_{s\to t}$, which maps the token id from the source vocabulary to the one with the highest cosine similarity from the target vocabulary.

2.2 Alignment Evaluation

Figure 2(1) illustrates our metric to evaluate the performance of alignment matrix $M_{s\to t}$. We first tokenize the test corpus $\mathcal{C}$ using different tokenizers, which results in $\mathcal{C}_s$ and $\mathcal{C}_t$. The token ID corpus $\mathcal{C}_s$ from the source tokenizer is converted by the alignment matrix $M_{s\to t}$, and comes to the corpus $\mathcal{C}_t^{'}$. The higher BLEU score between $\mathcal{C}_t^{'}$ and the corpus $\mathcal{C}_t$ from the Tokenizer$_t$, the better alignment matrix $M_{s\to t}$ is. The other two metrics, BLEU-1 and BertScore, to evaluate the performance of alignment matrix are investigated in the Appendnix B.4.

2.3 Progressive Adaptation

Given the alignment matrix $M_{s\to t}$, the parameters of each token in the target vocabulary are initialized from the ones of the most similar source token. We find that these re-arranged embedding and lm_head provide a good initialization for the new model (Section 3.2 and 4.2). Figure 2(2) illustrates the two stages designed for a LLM to adapt to the new vocabulary. The re-arranged embedding and lm_head are tuned first to avoid loss spike and improve the stability during tuning (Figure 5(c)). The other parameters of internal layer are further tuned together in the last half part. We acknowledge that a better designed adaptation method can bring a higher performance, which can be investigated in the future.

3 Experiments

3.1 Experiments Settings

Large Language models We adopt the fully open-source language model series Pythia (Biderman et al., 2023) as base models in this work. It is noted that this work does not intend to achieve the state-of-the-art performance of large language models but rather investigate an effective method to replace the tokenizer. To achieve token-level knowledge transfer from other capable large language models, the tokenizers (vocabularies) of Gemma (Team et al., 2024), Qwen2 (Yang et al., 2024), LLaMA2 (Touvron et al., 2023b), and LLaMA3 (Meta, 2024) are selected as the target tokenizer to replace. We report hyper-parameters in Appendix A.

Corpus To reduce the risk of distribution shift from the training data, we choose the vanilla pre-training corpus (Gao et al., 2020; Soboleva et al., 2023; Kocetkov et al., 2023) of the base model

Pythia in the fine-tuning process. Corpora from downstream tasks and multiple languages are applied in token-level distillation and cross-lingual transfer experiments (Section 4).

Evaluation Tasks Following the common practices to evaluate large language models (Lin et al., 2022; Biderman et al., 2023; Zhang et al., 2024), there are 10 datasets, including commonsense reasoning (Conneau et al., 2018; Clark et al., 2018; Mihaylov et al., 2018; Zellers et al., 2019; Ponti et al., 2020; Bisk et al., 2020; Sakaguchi et al., 2020; Tikhonov & Ryabinin, 2021) and reading comprehension (Clark et al., 2019) tasks, used in this work. To avoid the randomness from the prompt and evaluation method, we adopt the default prompt from the commonly used language model evaluation harness framework (Gao et al., 2024).

Baselines We introduce the following methods from the cross-lingual vocabulary adaptation domain as baseline methods in this work:

- **Random Initialization** for each token $t \in \{\mathcal{V}_t \setminus (\mathcal{V}_t \cap \mathcal{V}_s)\}$ employs the default initialization method of huggingface transformers and reuses the parameters of token $t \in \{\mathcal{V}_t \cap \mathcal{V}_s\}$, which belongs to overlapping vocabularies.
- **Random Permutation** initializes each token $t \in \{\mathcal{V}_t \setminus (\mathcal{V}_t \cap \mathcal{V}_s)\}$ using the parameter of a randomly chosen token from the source vocabulary. The parameters of shared tokens are also reused.
- **WECHSEL** (Minixhofer et al., 2022) transfers embeddings of source tokens into target tokens by tokenizing and recomposing additional word embeddings W_s and W_t, which are aligned with a bilingual dictionary.
- **OFA** (Liu et al., 2024) factorizes the embeddings of source model E_s into the primitive embeddings P and source coordinates F_s that is further re-composed by multilingual word embeddings W to the target coordinates F_t. The assembled primitive embeddings P and target coordinates F_t comes the target embeddings E_t.
- **Focus** (Dobler & de Melo, 2023) initializes the embedding parameters of token $t \in \{\mathcal{V}_t \setminus (\mathcal{V}_t \cap \mathcal{V}_s)\}$ using the weighted sum of the ones from the token $t \in \{\mathcal{V}_t \cap \mathcal{V}_s\}$. It largely depends on the size of $\| \mathcal{V}_t \cap \mathcal{V}_s \|$, and performs poorly when the overlapping percentage of $\mathcal{V}_t$ and $\mathcal{V}_s$ is low.
- **ZeTT** (Minixhofer et al., 2024) trains an additional hypernetwork H_θ to generate the parameters for each token $t \in \mathcal{V}_t$. The added hypernetwork brings a lot of training cost.

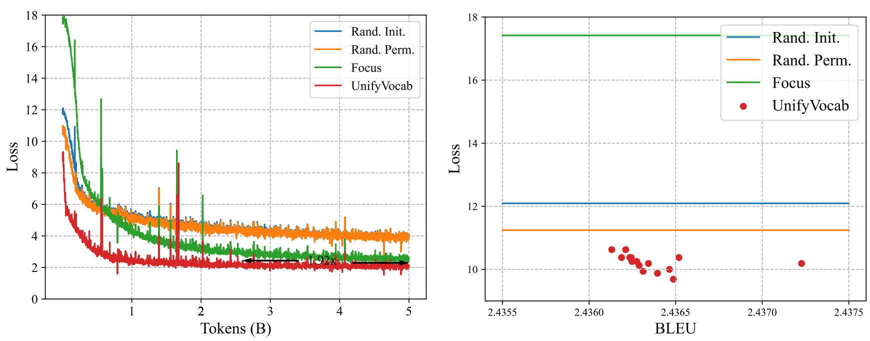
Table 1: The main results of replacing the vocabulary of Pythia to Gemma using 10B tokens from the Pile corpus. "w/ SlimPajama" adopts 1B tokens from SlimPajama to train GloVe embeddings."+ Align Rep." adds alignment process for GloVe embedding before calculating cosine similarity following Moschella et al. (2023). The best performance among the vocabulary adaptation methods is displayed in **bold**.

	ARC-E		BoolQ		HellaSwag		OpenbookQA		PIQA		WinoGrande		Avg	
Model	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**
Pythia$_{1B}$	56.82	58.71	60.43	57.37	37.68	37.66	18.80	19.00	70.40	71.49	53.20	52.01	49.55	49.37
w/ Rand. Init.	31.36	31.61	37.83	49.11	26.35	26.40	14.00	12.60	54.57	55.33	49.17	49.17	35.55	37.37
w/ Rand. Perm.	31.69	32.95	37.77	54.80	26.43	26.39	14.00	12.60	55.50	55.98	47.04	50.67	35.40	38.90
w/ OFA	38.17	37.79	55.14	52.35	28.29	28.62	14.40	12.20	58.43	58.54	49.96	50.99	40.73	40.08
w/ WECHSEL	43.35	45.33	56.61	54.34	32.53	32.41	14.80	16.20	61.70	62.89	52.01	52.72	43.50	43.98
w/ Focus	46.55	48.95	56.21	**55.78**	32.27	32.46	19.20	18.00	63.82	64.80	51.70	51.78	44.96	45.29
w/ ZeTT	47.14	49.03	57.06	53.70	34.06	34.06	18.40	19.40	64.15	65.34	52.09	51.22	45.48	45.46
w/ UnifyVocab	**54.46**	**56.86**	58.90	52.26	36.16	36.27	**21.00**	**20.20**	**67.74**	**68.50**	52.25	50.91	48.42	47.50
w/ SlimPajama	53.54	55.68	57.55	53.85	36.10	35.99	19.40	**20.20**	67.03	67.52	52.09	51.22	47.62	47.41
+ Align Rep.	54.25	56.65	**59.33**	54.68	**37.08**	**36.91**	20.20	19.40	67.36	68.17	**54.38**	**52.80**	**48.77**	**48.10**
Pythia$_{2.8B}$	63.80	67.00	63.91	65.14	45.32	45.04	24.00	25.20	74.05	74.43	58.64	60.77	54.95	56.26
w/ Rand. Init.	30.47	32.91	38.20	51.07	26.46	26.69	14.40	13.20	55.17	55.06	48.30	50.51	35.50	38.24
w/ Rand. Perm.	31.48	31.86	37.83	50.46	26.48	26.49	13.60	14.40	54.03	54.95	50.20	48.86	35.60	37.84
w/ Focus	54.29	58.16	61.44	62.84	38.38	39.09	20.00	20.20	68.44	68.28	54.62	56.04	49.53	50.77
w/ UnifyVocab	**61.62**	**65.15**	**63.82**	**65.47**	**43.13**	**43.18**	**23.40**	**25.80**	**72.14**	**72.42**	**58.17**	**61.17**	**53.71**	**55.53**

3.2 Main Results and Analyses

We first conduct experiments to replace the tokenizer of Pythia with the Gemma tokenizer using 10B tokens. Results on six datasets are shown in Table 1. Given limited tokens to fine-tune, it can be found that UnifyVocab performs better than the other three baseline methods. The average improvement of UnifyVocab over the strong baseline method ZeTT reaches 2.49%, and the 97.63% performance of the vanilla model is reserved after vocabulary replacement. Replace the corpus to train the GloVe embedding with 1B SlimPajama (Soboleva et al., 2023) tokens comes to a comparable results. It demonstrates the robustness of our method on the pre-training corpus for token embedding and alignment matrix. We find that the performance can be further advanced by aligning the GloVe embedding into the relative representation using 300 common tokens occur in both vocabularies following Moschella et al. (2023), which is the row with "+ Align Rep." label.

Better alignment brings better initialization. The loss curves of Pythia$_{1B}$ with different methods during the first 5B tokens training are shown in Figure 3(a). We find that UnifyVocab brings a better initialization and decreases the first-step training loss from 17.8 (Focus) to 9.5. Moreover, the training process with UnifyVocab is faster than the ones with other methods, which reaches 2.75 with only 2.6B tokens and is 1.92x (5B/2.6B) speed up than the method Focus.



(a) Initialization method comparison (b) The relationship of initial loss and BLEU

Figure 3: The training loss of Pythia$_{2.8b}$ with different methods (a) and $M_{s \to t}$ learned using UnifyVocab, which is denoted by red point (b).

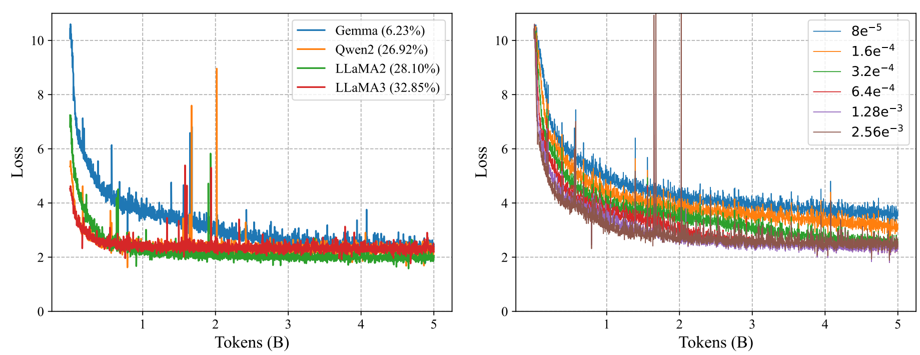
Table 2: The benchmark results of UnifyVocab using 10B tokens from the Pile corpus. The overlapping ratio between the vocabulary of Pythia and other models are 6.23%(Gemma), 26.92%(Qwen2), 28.10%(LLaMA2), 32.85%(LLaMA3).

Model	$\#\mathcal{V}$ (k)	ARC-E 0	ARC-E 5	BoolQ 0	BoolQ 5	HellaSwag 0	HellaSwag 5	OpenbookQA 0	OpenbookQA 5	PIQA 0	PIQA 5	WinoGrande 0	WinoGrande 5	Avg 0	Avg 5
Pythia$_{1B}$	50.3	56.82	58.71	60.43	57.37	37.68	37.66	18.80	19.00	70.40	71.49	53.20	52.01	49.55	49.37
→ Gemma	256.0	54.46	56.86	**58.90**	52.26	36.16	36.27	**21.00**	20.20	67.74	68.50	52.25	50.91	48.42	47.50
→ Qwen2	152.1	54.46	57.07	54.80	49.79	37.18	37.04	19.20	18.40	68.44	**70.24**	53.35	52.80	47.91	47.56
→ LLaMA2	32.0	49.45	52.02	58.32	55.75	35.38	35.45	18.80	17.80	66.32	66.65	53.91	50.91	47.03	46.43
→ LLaMA3	128.0	**54.63**	**57.28**	55.84	**53.70**	**37.34**	**37.43**	20.20	**20.40**	**69.04**	70.18	**54.46**	**53.43**	**48.59**	**48.74**
Pythia$_{2.8B}$	50.3	63.80	67.00	63.91	65.14	45.32	45.04	24.00	25.20	74.05	74.43	58.64	60.77	54.95	56.26
→ Gemma	256.0	61.62	65.15	63.82	**65.47**	43.13	43.18	23.40	**25.80**	72.14	72.42	58.17	**61.17**	53.71	**55.53**
→ Qwen2	152.1	**62.54**	**66.04**	62.35	63.55	44.46	44.39	23.20	24.60	**73.50**	**73.56**	**59.04**	59.59	54.18	55.29
→ LLaMA3	128.0	61.83	64.60	**64.40**	63.94	**44.62**	**44.59**	**23.80**	25.60	73.45	73.29	57.54	58.72	**54.27**	55.12
Pythia$_{6.9B}$	50.3	65.99	69.23	62.84	62.02	47.56	47.64	25.00	27.00	74.65	75.41	60.46	62.43	56.08	57.29
→ Gemma	256.0	65.40	68.35	62.39	59.57	45.75	45.86	22.00	25.60	73.39	74.10	60.38	61.17	54.89	55.77
→ Qwen2	152.1	65.57	**68.43**	**64.07**	57.61	46.84	46.91	**25.60**	25.40	73.45	74.65	61.17	63.14	56.12	56.02
→ LLaMA3	128.0	**66.46**	68.35	63.79	**60.64**	**47.28**	**47.31**	**25.60**	**28.20**	**74.48**	**75.84**	**61.48**	**63.30**	**56.52**	**57.27**

We further investigate the impact of the learned alignment matrix $M_{s\to t}$ by changing the hyper-parameters of GloVe. It is noted that different alignment matrices $M_{s\to t}$ bring different initial parameters, and also come to different BLEU scores on the same evaluation corpus. Figure 3(b) illustrates the negative relationship between the first-step training loss and the BLEU. In other words, the higher the BLEU score for the alignment matrix $M_{s\to t}$, the better the initial parameter is. The other metrics like BLEU-1 and BertScore are also used to evaluate the alignment metrix, and also show a negative relationship with the initial training loss in Appendix B.4.

More overlapping comes to faster convergence and higher performance. The UnifyVocab is further applied to the other three target tokenizers: Qwen2, LLaMA2, and LLaMA3. Table 2 reports the performance of models after replacing vocabulary on six datasets. UnifyVocab recovers 98.02% performance of the base model on average with only 10B tokens. Given a target vocabulary with more tokens than the one of Pythia (50.3k), it can be found that a higher overlapping ratio brings a better performance of model replaced (97.62% for Gemma to 99.07% for LLaMA3). The zero-shot in-context learning results for $\text{Pythia}_{6.9\text{B}}$ with LLaMA3 vocabulary even surpass the vanilla base model. The results of $\text{Pythia}_{1\text{B}}$ with LLaMA2 vocabulary are only 94.47%, which is inferior to the average result of 98.02%. We argue that it may come from the missing 75.0M parameters (7.4% for $\text{Pythia}_{1\text{B}}$) after switching to a 32.0k vocabulary from the 50.3k vocabulary.

Figure 4(a) shows the training loss curve during the first 5B tokens. The replacing process of the Gemma tokenizer is the slowest, which may come from the only 6.23% overlapping ratio between two vocabularies. It is interesting to find that other conditions for three tokenizers converge with only 1B tokens under the same setting. Further analyses for the convergence of vocabulary adaptation refer to Appendix B.2, which shows a similar phenomenon.



(a) Overlapping ratio

(b) Learning rate

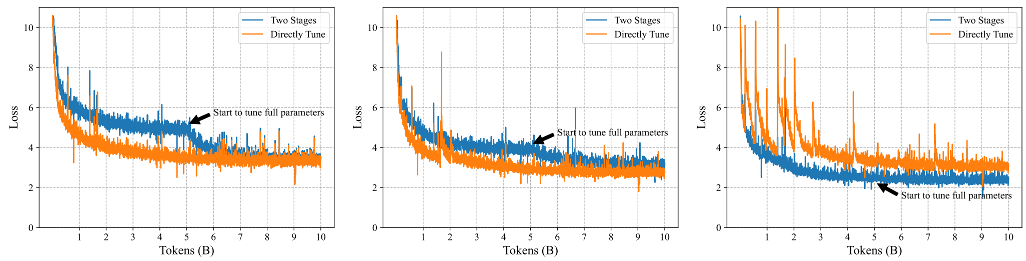
Figure 4: The training loss curve of $\text{Pythia}_{1\text{B}}$ for different overlapping ratios (a), and learning rate used during replacing to the Gemma tokenizer (b).

Two-stage tuning brings a more stable convergence. To replace the tokenizer and keep the performance of the vanilla model, we adopt only fine-tuning the vocabulary-related parameters at the first stage. The main reason for two-stage tuning is to take these parameters as the adapters for different tokenizers, and avoid the well-trained parameters of the internal layer distracted by the new initialized parameters.

Figure 5 illustrates that our two-stage tuning method makes the convergence more stable under a high learning rate like $6.4e^{-4}$, which comes to better performance after tuning on 10B tokens. It is noted that the loss spike also occurs at the first stage, fine-tuning vocabulary-related parameters only, under such a high learning rate like $2.56e^{-3}$ in Figure 4(b).

4 APPLICATIONS

In this section, we illustrate two direct applications of UnifyVocab: token-level distillation (Section 4.1) and cross-lingual knowledge transfer (Section 4.2).



(a) Learning rate 2e^{-5} (b) Learning rate 8e^{-5} (c) Learning rate 6.4e^{-4}

Figure 5: The loss curve of Pythia$_{1B}$ under two-stage tuning or direct full parameters tuning.

4.1 Token-level Distillation

To compensate for the performance gap between these capable open-source language models and Pythia, we take these models as the teacher model of Pythia after replacing the tokenizer. Training samples from downstream tasks and the corpus of Pile are used in the token-level distillation experiments. The logit of each token from the teacher model is taken as the soft label of Pythia to learn. We empirically set the proportion of training samples to 15% to avoid a significant degradation in the performance of language modeling (Wei et al., 2023).

Table 3 reports the results of two baseline methods and token-level distillation from three teacher models using 235M tokens. We can find that token-level distillation is significantly better than the one of sentence-level distillation. Given the same teacher model Qwen2$_{7B}$, the improvement of Pythia over the sentence-level distillation result reaches 4.37%, which further demonstrates the importance of unifying tokenizer between models. The knowledge transfer between models will be constrained in sentence-level distilling without unifying vocabulary. It is also noted that models with token-level distillation on strong teacher models like Qwen2 outperform the ones of direct tuning.

Table 3: The main results of token-level distillation on six downstream tasks using 235M tokens. "+Sentence distill" denotes the sentence-level distillation results with Qwen2$_{7B}$(Yang et al., 2024), which fine-tunes on the output from Qwen2$_{7B}$ given questions as prompt.

	ARC-E		**BoolQ**		**HellaSwag**		**OpenbookQA**		**PIQA**		**WinoGrande**		**Avg**	
Model	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**	**0**	**5**
Pythia$_{1B}$	56.82	58.71	60.43	57.37	37.68	37.66	18.80	19.00	70.40	71.49	53.20	52.01	49.55	49.37
+ Direct tuning	57.49	55.64	70.70	72.11	41.24	41.60	25.40	28.40	69.04	70.08	54.70	54.78	53.10	53.77
+ Sentence distill	52.27	53.41	67.49	67.06	39.03	39.08	21.80	22.80	66.97	68.99	51.85	52.17	49.90	50.58
w/ Gemma$_{7B}$	55.39	56.99	67.19	69.69	36.53	37.26	19.00	22.80	68.82	69.21	52.33	53.51	49.88	51.58
w/ Qwen2$_{7B}$	62.33	63.17	70.18	72.54	41.58	42.21	22.00	**28.20**	**73.01**	73.18	55.01	55.56	54.02	55.81
w/ LLaMA3$_{8B}$	**64.02**	**64.56**	**73.91**	**74.19**	**42.11**	**42.34**	**24.20**	27.60	72.74	**73.83**	**55.49**	**56.43**	**55.41**	**56.49**
Pythia$_{6.9B}$	65.99	69.23	62.84	62.02	47.56	47.64	25.00	27.00	74.65	75.41	60.46	62.43	56.08	57.29
+ Direct tuning	66.25	66.20	79.30	78.87	52.21	53.39	33.20	33.00	72.91	74.48	62.90	61.72	61.13	61.28
+ Sentence distill	61.70	65.36	76.64	76.88	48.98	51.33	28.20	30.40	70.18	71.55	58.96	62.19	57.44	59.62
w/ Gemma$_{7B}$	67.59	68.94	76.06	75.66	47.83	48.36	28.40	31.40	73.78	75.52	59.04	64.17	58.78	60.67
w/ Qwen2$_{7B}$	**71.72**	**73.27**	**79.85**	**80.00**	**50.78**	**51.12**	**29.20**	**34.00**	**77.26**	**77.91**	**61.33**	**64.56**	**61.69**	**63.48**
w/ LLaMA3$_{8B}$	67.05	69.78	77.83	78.78	48.83	50.15	26.00	32.00	74.21	76.22	60.22	60.93	59.02	61.31

4.2 Cross-lingual Transfer

The tokens for other languages can be aligned and initialized by the tokens with similar semantics in the source vocabulary, which can speed up the cross-lingual knowledge transfer. In this section, we use UnifyVocab to conduct cross-lingual transfer experiments using 4B tokens from the CulturaX corpus. The tokenizer of Qwen2 is used as the target tokenizer for Pythia.

Table 4: The normalized perplexity on the valid corpus of CulturaX. The perplexity is normalized to the vocabulary of Pythia following Wei et al. (2023). "**High**", "**Medium**" and "**Low**" denotes the available amount of linguistic resources.

		High					**Medium**					**Low**			
Model	**#Tune (B)**	**ar**	**de**	**en**	**ja**	**zh**	**bn**	**ko**	**th**	**uk**	**vi**	**ta**	**te**	**ur**	**Avg**
$\text{Qwen2}_{1.5B}$	–	4.7	11.1	15.7	6.0	4.6	2.4	3.3	2.6	5.7	3.3	2.8	3.4	4.0	5.3
Pythia_{1B}	–	7.6	15.4	**21.7**	9.9	13.2	3.4	5.6	4.3	6.7	6.3	2.9	3.3	5.8	8.2
w/ Focus	0	$4.1e^3$	$1.7e^5$	$1.8e^6$	$2.1e^4$	$9.6e^2$	$6.5e^4$	$1.0e^3$	$5.6e^3$	$1.6e^6$	$8.4e^2$	$5.0e^4$	$1.9e^5$	$1.9e^5$	$3.1e^5$
	4	8.3	27.1	59.7	14.0	14.0	3.6	5.9	3.8	7.3	5.9	3.5	3.6	4.3	12.4
w/ UnifyVocab	0	$1.6e^2$	$9.4e^2$	$3.6e^2$	$3.1e^2$	$1.5e^2$	89.6	94.1	94.3	$1.6e^2$	$1.1e^2$	36.1	27.8	$1.1e^2$	$2.0e^2$
	4	**6.5**	**14.1**	24.0	**9.0**	**9.2**	**2.5**	**4.5**	**3.2**	**5.3**	**4.5**	**2.3**	**2.4**	**3.8**	**7.0**
Qwen2_{7B}	–	3.9	8.1	11.8	4.9	3.8	2.1	2.9	2.3	3.8	2.9	2.3	2.6	3.3	4.2
$\text{Pythia}_{6.9B}$	–	5.9	10.8	**16.7**	7.9	9.9	3.0	4.6	3.7	4.9	4.9	2.6	2.9	4.8	6.3
w/ Focus	0	$6.9e^3$	$1.6e^5$	$1.2e^6$	$2.4e^4$	$1.3e^3$	$2.5e^4$	$7.2e^2$	$3.3e^3$	$1.9e^6$	$7.9e^2$	$1.7e^4$	$1.5e^5$	$1.2e^5$	$2.8e^5$
	4	6.8	17.6	39.3	10.8	11.1	2.5	5.0	3.3	5.2	4.8	2.3	2.5	3.7	8.8
w/ UnifyVocab	0	$1.9e^2$	$8.0e^2$	$2.8e^2$	$3.3e^2$	$1.6e^2$	85.3	97.0	94.3	$1.7e^2$	$1.1e^2$	36.1	23.8	$1.0e^2$	$1.9e^2$
	4	**5.4**	**10.1**	18.1	**7.5**	**8.0**	**2.1**	**4.0**	**2.8**	**4.1**	**3.8**	**2.1**	**2.1**	**3.1**	**5.6**

As shown in Table 4, the perplexity of Pythia initialized with UnifyVocab ($2.0e^2$) is significantly better than the one of Focus baseline ($2.9e^5$). After only 4B tokens tuning, the improvement of UnifyVocab is 13.1% over the vanilla model on average. The performance of Pythia using UnifyVocab on three low-resource languages even outperforms the ones of Qwen2 under a similar parameter amount.

Table 5 and 7 report in-context learning results on four multilingual datasets. We can find that UnifyVocab brings a better-initialized model than the baseline method Focus (+3.5%), and transfers the knowledge into other languages like Vietnamese (vi, +2.3%) and Urdu (ur, +0.9%).

It is interesting to find that the perplexity of Pythia_{1B} initialized by UnifyVocab reaches $2.0e^2$, while the in-context learning results are comparable with the ones of Focus after 4B tokens tuning. We argue that it arises from the mostly reserved English ability with UnifyVocab, which is 56.2% outperforming the 43.6% of Focus.

Table 5: Zero-shot in-context learning results of cross-lingual transfer. "#Tune(B)=0" denotes performance of the model after parameter initialization without any tuning. Refer to Table 7 in Appendix B.5 for five-shot results.

		XNLI							**XCOPA**				**XStoryCloze**				**XWinograd**			
Model	**#Tune(B)**	**en**	**de**	**zh**	**ar**	**th**	**vi**	**ur**	**en**	**th**	**vi**	**ta**	**en**	**zh**	**ar**	**te**	**en**	**zh**	**ja**	**Avg**
Pythia_{1B}	–	51.0	37.8	**42.6**	35.9	34.8	37.0	34.7	**62.4**	54.4	50.6	55.4	**64.4**	48.7	48.0	52.9	57.1	**53.2**	**59.3**	**48.9**
w/ Focus	0	32.8	32.2	33.6	33.6	33.5	32.0	32.8	49.4	51.2	48.4	54.4	46.0	47.7	**48.7**	46.5	49.7	47.2	50.3	42.8
	4	46.0	35.1	34.9	32.9	32.5	35.4	34.7	53.0	52.6	50.0	54.2	57.1	50.0	47.5	52.5	52.2	51.7	54.4	45.9
w/ UnifyVocab	0	48.4	35.9	33.4	33.1	31.8	32.5	33.8	54.6	52.0	47.4	**57.2**	58.6	46.5	46.7	51.0	54.4	50.2	50.5	45.4
	4	**51.2**	**39.0**	42.3	**38.5**	**35.8**	**38.9**	**35.7**	60.8	**55.2**	**51.8**	53.8	64.0	**51.0**	47.5	**54.1**	56.0	52.5	56.9	**49.2**
$\text{Pythia}_{6.9B}$	–	54.4	39.0	46.2	39.3	39.8	39.3	36.4	70.8	57.6	51.2	53.0	70.7	54.0	50.4	53.5	63.7	60.1	67.1	52.6
w/ Focus	0	31.5	31.3	33.0	32.6	33.4	32.2	32.6	46.4	52.4	49.0	**56.6**	44.6	47.3	48.2	47.4	48.3	46.8	51.1	42.5
	4	52.6	34.9	36.6	35.1	33.6	39.0	34.5	61.6	52.4	52.0	53.8	62.1	49.3	47.1	**54.6**	56.2	52.1	58.9	48.1
w/ UnifyVocab	0	50.9	**37.6**	34.3	34.6	33.7	33.1	33.7	60.2	52.6	48.0	55.8	63.1	47.1	47.0	50.3	59.6	48.6	51.4	46.8
	4	**55.1**	35.5	**41.6**	**39.1**	**39.6**	**42.8**	**37.1**	**70.2**	**56.0**	**53.6**	51.4	**70.4**	**52.5**	**49.1**	54.3	**61.5**	**54.0**	**60.7**	**51.4**

5 Related Works

Our work is related to word representation, large language models, and vocabulary adaption, which will be briefly introduced below.

Word Representation Based on the distributional semantic hypothesis, Bengio et al. (2003) introduced the neural probabilistic language model to learn word representation. Researchers mainly focus on improving the effectiveness during learning word representations (Mikolov et al., 2013a;b; Bojanowski et al., 2017), which provide a good initialization for neural networks like LSTM and GRU (Hochreiter, 1997; Chung et al., 2014). GloVe (Pennington et al., 2014) provides a method to train word representations from a view of global word-word co-occurrence matrix decomposition. It motivates us to train a word representation for each token and align the token ID from statistical co-occurrence information in the pre-training corpus.

Large Language Model Through scaling in the parameters and pre-training corpus (Kaplan et al., 2020; Hoffmann et al., 2022), large language models including GPT and LLaMA (Radford et al., 2018; 2019; Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023a;b; Meta, 2024; GLM et al., 2024) demonstrate impressive performance across multiple tasks. However, the knowledge transfer between different models is greatly hindered by the mismatch in the vocabulary. We aim to mitigate this problem by introducing an effective method to replace the tokenizer of a pre-trained large language model.

Vocabulary Adaption is investigated mainly in the multilingual domain, especially the cross-lingual knowledge transfer problem (Workshop et al., 2023; Muennighoff et al., 2023; Yang et al., 2023; Zhu et al., 2023; Üstün et al., 2024; Li et al., 2024). It aims to improve the encoding effectiveness of tokenizer on corpora from new languages, and is often implemented by extending the original vocabulary (Tran, 2020; Chau et al., 2020; Minixhofer et al., 2022; Dobler & de Melo, 2023; Downey et al., 2023). Most methods like Focus (Dobler & de Melo, 2023) rely on the tokens belonging to both source vocabulary and target vocabulary to initialize the other new tokens in the target vocabulary. Our method differs from these studies for the whole replacement of vocabulary using a limited corpus. It does not rely on the tokens in both source vocabulary and target vocabulary.

6 LIMITATIONS

The first limitation comes from the assumption that the pre-training data distribution is available. We conduct experiments on Pythia with different parameter amounts, which provide public model weights and pre-training corpus. Due to the limited computation resource budget, open-source language models with unknown pre-training corpus like Mistral (Jiang et al., 2023) are not investigated in this work. However, the pre-training corpus distribution of open-weighted large language models can be roughly inferred by the BPE vocabulary (Hayase et al., 2024). It can re-construct a similar pre-training corpus to conduct replacing tokenizer experiments.

The 10B tokens of model tuning cost in replacing a tokenizer using UnifyVocab is another limitation, although it is only 3.33% of the 300B tokens pre-training corpus for Pythia. From the loss curve of UnifyVocab (Figure 4), we find that the start of full parameters tuning can be less than 5B tokens, which may result in a better balance.

7 CONCLUSION AND FUTURE WORK

In this paper, we introduce a method named UnifyVocab to replace the tokenizer of large language models from a token-token co-occurrence view. Extensive experiments demonstrate that UnifyVocab reserves the most performance of vanilla models (98.02% on average) using only 10B tokens, which enables deeper knowledge transfer between models like token-level distillation and cross-lingual knowledge transfer.

Beyond replacing the vocabulary of large language models, our method can be extended to replace the vocabulary of multi-modal models by aligning different modal tokens. The other direction is to develop a method with less training cost, e.g., incorporating meta-learning to replace the two-stage tuning method.

8 REPRODUCIBILITY STATEMENT

Codes and weights will be made public after review to advocate future research. Hyper-parameters are reported in the Appendix A. The weight of models with replaced vocabulary and source codes will be public after review to advocate future research.

REFERENCES

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. Llemma: An open language model for mathematics. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=4WnqRR915j.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155, 2003.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A suite for analyzing large language models across training and scaling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2397–2430. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/biderman23a.html.

Yonatan Bisk, Rowan Zellers, Ronan Le bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 7432–7439, Apr. 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2017. URL https://arxiv.org/abs/1607.04606.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. Parsing with multilingual BERT, a small corpus, and a small treebank. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1324–1334, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.118. URL https://aclanthology.org/2020.findings-emnlp.118.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL `https://aclanthology.org/D18-1269`.

Yiming Cui, Ziqing Yang, and Xin Yao. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*, 2023.

Konstantin Dobler and Gerard de Melo. FOCUS: Effective embedding initialization for monolingual specialization of multilingual models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13440–13454, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.829. URL `https://aclanthology.org/2023.emnlp-main.829`.

C.m. Downey, Terra Blevins, Nora Goldfine, and Shane Steinert-Threlkeld. Embedding structure matters: Comparing methods to adapt multilingual vocabularies to new languages. In Duygu Ataman (ed.), *Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL)*, pp. 268–281, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.mrl-1.20. URL `https://aclanthology.org/2023.mrl-1.20`.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL `https://arxiv.org/abs/2101.00027`.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 07 2024. URL `https://zenodo.org/records/12608602`.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024.

Jonathan Hayase, Alisa Liu, Yejin Choi, Sewoong Oh, and Noah A Smith. Data mixture inference: What do bpe tokenizers reveal about their training data? *arXiv preprint arXiv:2407.16607*, 2024.

S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022. URL `https://arxiv.org/abs/2203.15556`.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL `https://arxiv.org/abs/2001.08361`.

Denis Kocetkov, Raymond Li, Loubna Ben allal, Jia LI, Chenghao Mou, Yacine Jernite, Margaret Mitchell, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Dzmitry Bahdanau, Leandro Von Werra, and Harm de Vries. The stack: 3 TB of permissively licensed source code. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=pxpbTdUEpD`.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In Yansong Feng and Els Lefever (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 318–327, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-demo.28. URL `https://aclanthology.org/2023.emnlp-demo.28`.

Chong Li, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. Improving in-context learning of multilingual generative language models with cross-lingual alignment. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8058–8076, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.445. URL `https://aclanthology.org/2024.naacl-long.445`.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 9019–9052, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.616`.

Yihong Liu, Peiqin Lin, Mingyang Wang, and Hinrich Schuetze. OFA: A framework of initializing unseen subword embeddings for efficient large-scale multilingual continued pretraining. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 1067–1097, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.68. URL `https://aclanthology.org/2024.findings-naacl.68`.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Meta. Introducing meta llama 3: The most capable openly available llm to date. *Qwen blog*, 2024. URL `https://ai.meta.com/blog/meta-llama-3/`.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. In *International Conference on Learning Representations*, 2018.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1260. URL `https://aclanthology.org/D18-1260`.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*, 2013b.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3992–4006, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.293. URL https://aclanthology.org/2022.naacl-main.293.

Benjamin Minixhofer, Edoardo Maria Ponti, and Ivan Vulić. Zero-shot tokenizer transfer, 2024. URL https://arxiv.org/abs/2405.07883.

Luca Moschella, Valentino Maiorca, Marco Fumero, Antonio Norelli, Francesco Locatello, and Emanuele Rodolà. Relative representations enable zero-shot latent space communication. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=SrC-nwieGJ.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15991–16111, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.891. URL https://aclanthology.org/2023.acl-long.891.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. *arXiv preprint arXiv:2309.09400*, 2023.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. URL https://arxiv.org/abs/2303.08774.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https://aclanthology.org/D14-1162.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2362–2376, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.185. URL https://aclanthology.org/2020.emnlp-main.185.

Qwen. Introducing qwen1.5. *Qwen blog*, 2024. URL https://qwenlm.github.io/blog/qwen1.5/.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI blog*, 2018. URL https://openai.com/blog/language-unsupervised/.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019. URL https://openai.com/blog/better-language-models/.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pp. 3505–3506, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3406703. URL https://doi.org/10.1145/3394486.3406703.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8732–8740, Apr. 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In Katrin Erk and Noah A. Smith (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama. https://www.cerebras.net/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 16857–16867. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/c3a690be93aa602ee2dc0ccab5b7b67e-Paper.pdf.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Alexey Tikhonov and Max Ryabinin. It's All in the Heads: Using Attention Heads as a Baseline for Cross-Lingual Transfer in Commonsense Reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3534–3546, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.310. URL https://aclanthology.org/2021.findings-acl.310.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023a. URL https://arxiv.org/abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

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

Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. Aya model: An instruction finetuned open-access multilingual language model. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15894–15939, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.845.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. Extending multilingual BERT to low-resource languages. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 2649–2656, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.240. URL https://aclanthology.org/2020.findings-emnlp.240.

Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, et al. Skywork: A more open bilingual foundation model. *arXiv preprint arXiv:2310.19341*, 2023.

T Wolf. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

BigScience Workshop, :, 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, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2023. URL `https://arxiv.org/abs/2211.05100`.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. Bigtranslate: Augmenting large language models with multilingual translation capability over 100 languages. *arXiv preprint arXiv:2305.18098*, 2023.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL `https://aclanthology.org/P19-1472`.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024. URL `https://arxiv.org/abs/2401.02385`.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Extrapolating large language models to non-english by aligning languages. *arXiv preprint arXiv:2308.04948*, 2023.

A Hyper-parameters

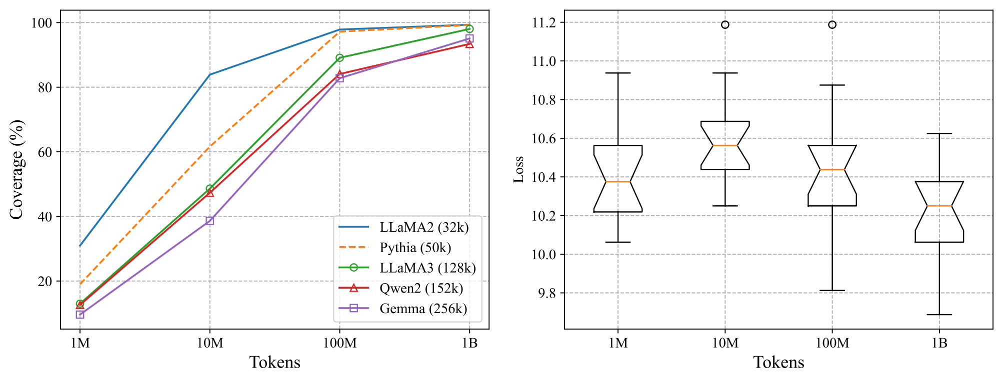
GloVe Training We empirically train GloVe vectors with 1B tokens, which covers most tokens from Gemma (95.10%), Qwen2 (93.40%), LLaMA2 (99.35%), and LLaMA3 (98.04%). The dimension size is set to 300. The max training iteration and the size of the slide window are 15.

Model Tuning The optimizer adopted in this work is AdamW (Loshchilov & Hutter, 2019), where $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate for baseline methods is set to 5e-5 to reduce the loss spike in Figure 5(b) and Figure 5(c). We adopt bf16 mixed precision training and ZeRO-1 to save GPU memory cost and speed up the training process (Micikevicius et al., 2018; Rasley et al., 2020). Following Biderman et al. (2023), the batch size is set to 2M tokens and the max sequence length is 2048.

B Additional Results

B.1 GloVe Vectors

We show the effects of different token amounts for the GloVe vectors training in Figure 6. It can be found that 1B tokens used in this work provide a high vocabulary coverage (>90%) and better initialization for $\text{Pythia}_{\text{1B}}$. Due to the limited computation budget, experiments with more than 1B tokens are not conducted.

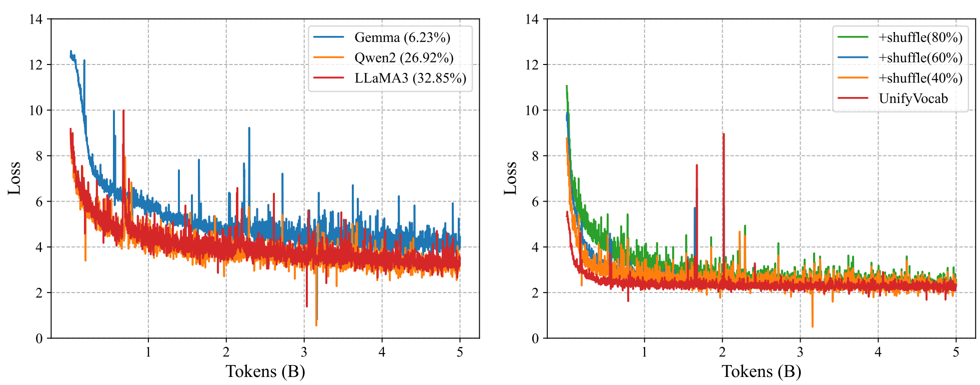


(a) Vocabulary coverage (b) Initial loss with Gemma tokenizer

Figure 6: The average vocabulary coverage (a) and initial training loss of $\text{Pythia}_{\text{1B}}$ (b) under different amount tokens to train the GloVe vector.

B.2 Convergence Analysis

To investigate the effect of overlapping rate between two tokenizers to the convergence of training, we plot Figure 7(a) for the random initialization baseline method. The convergence of Gemma tokenizer is slower than the other tokenizers and comes to worse results, which are similar to the case in 4(a). Moreover, we randomly shuffle the alignment matrix learned in UnifyVocab to imitate the case that other worse methods rather than cosine similarity to calculate the alignment matrix. Figure 7(b) shows that the higher percentage of randomly shuffle comes to higher initial training loss and slower convergence.



(a) Random Initialization (b) UnifyVocab with randomly shuffle for Qwen2

Figure 7: The training loss for random initialization to different tokenizers (a) and UnifyVocab for Qwen2 using $\text{Pythia}_{\text{1b}}$.

B.3 Vocabulary Adaptation Results with 2B Tokens

We further investigate a challenge condition that only 2B tokens are provided to adapt the target vocabulary. To meet the requirement, batch size is set to 1M tokens and training steps are reduced to 2k, correspondingly. Table 6 shows results of adapting to other 3 tokenizers using UnifyVocab. It can be found that 95.66% performance of vanilla model is recovered on average, which further demonstrates the effectiveness of our method.

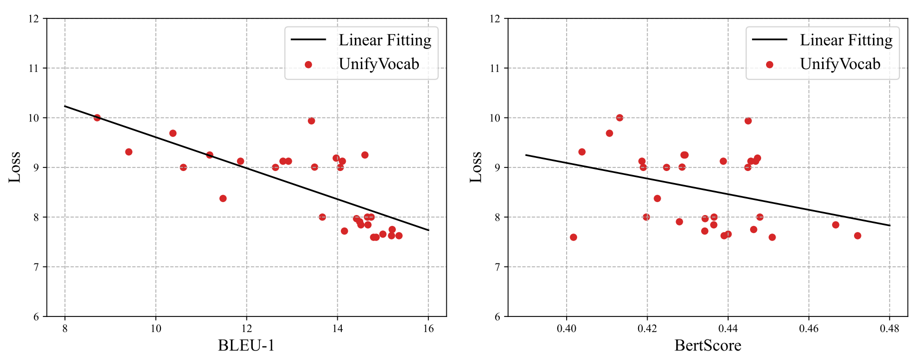
B.4 Additional Alignment Metrics

The BLEU-1 and BertScore can also be used to evaluate the performance of alignment matrix learned. The alignment evaluation process of BLEU-1 is same with the one of BLEU, which is the averaged of

Table 6: The main results of replacing the vocabulary of Pythia for UnifyVocab using 2B tokens from the Pile corpus.

Model	$\#\mathcal{V}$ (k)	ARC-E 0	ARC-E 5	BoolQ 0	BoolQ 5	HellaSwag 0	HellaSwag 5	OpenbookQA 0	OpenbookQA 5	PIQA 0	PIQA 5	WinoGrande 0	WinoGrande 5	Avg 0	Avg 5
Pythia$_{1B}$	50.3	56.82	58.71	60.43	57.37	37.68	37.66	18.80	19.00	70.40	71.49	53.20	52.01	49.55	49.37
$\rightarrow$ Gemma	256.0	51.09	52.44	53.12	52.35	35.00	35.05	20.20	18.60	64.80	65.83	53.12	51.62	46.22	45.98
$\rightarrow$ Qwen2	152.1	53.41	55.47	53.52	55.81	36.12	36.38	20.80	18.00	68.50	68.88	54.38	52.80	47.79	47.89
$\rightarrow$ LLaMA3	128.0	51.73	55.09	59.05	55.08	36.42	36.52	19.40	19.60	67.68	68.34	53.43	53.75	47.95	48.06

BLEU-1, BLEU-2, BLEU-3 and BLEU-4. As for BertScore, we first de-tokenized the target token ID corpus $\mathcal{C}_t^{'}$ using Tokenizer$_t$ into the text corpus $\mathcal{C}^{'}$, and evaluate the semantic similarity between $\mathcal{C}^{'}$ and the vanilla test corpus $\mathcal{C}$ using the sentence embedding model named "all-mpnet-base-v2" (Song et al., 2020). As shown in Figure 8, these metrics both show a clear negative relationship with the inital training loss.



(a) Initial training loss and BLEU-1($\mathcal{C}_t, \mathcal{C}_t^{'}$) (b) Initial training loss and BertScore($\mathcal{C}, \mathcal{C}^{'}$)

Figure 8: The relationship between initial training loss and BLEU-1 (a) or BertScore (b) for Pythia$_{1b}$.

B.5 Cross-lingual Transfer

Table 7 reports the 5-shot in-context learning results on 4 multilingual datasets. The average improvement over the baseline method Focus is 3.4% after 4B tokens tuning. We can find that the model initialized by UnifyVocab is comparable to the one of Focus after 4B tokens tuning.

Table 7: The 5-shot in-context learning results of cross-lingual transfer.

Model	#Tune(B)	XNLI en	XNLI de	XNLI zh	XNLI ar	XNLI th	XNLI vi	XNLI ur	XCOPA en	XCOPA th	XCOPA vi	XCOPA ta	XStoryCloze en	XStoryCloze zh	XStoryCloze ar	XStoryCloze te	XWinograd en	XWinograd zh	XWinograd ja	Avg
Pythia$_{1B}$	–	46.2	38.6	38.9	36.9	35.2	38.9	34.9	64.0	54.0	49.4	55.2	65.5	48.4	48.2	53.0	68.9	59.7	51.4	49.3
w/ Focus	0	32.8	32.2	33.6	33.6	33.5	32.0	32.8	49.4	51.2	48.4	54.4	46.0	47.7	**48.7**	46.5	49.7	47.2	50.3	42.8
	4	47.0	36.7	35.4	34.3	33.5	35.1	33.9	54.2	52.2	51.6	54.8	57.0	50.4	47.6	52.2	55.4	53.8	50.9	46.4
w/ UnifyVocab	0	**48.4**	35.9	33.4	33.1	31.8	32.5	33.8	54.6	52.0	47.4	**57.2**	58.6	46.5	46.7	51.0	54.4	50.2	50.5	45.4
	4	44.5	**37.5**	**38.3**	**35.6**	**35.0**	**37.7**	**35.5**	**63.4**	**54.4**	**52.0**	53.8	**65.0**	**51.2**	48.1	**53.3**	**65.8**	**58.7**	**53.3**	**49.1**
Pythia$_{6.9B}$	–	53.0	40.7	41.7	38.9	37.3	41.3	35.1	75.2	58.0	54.2	52.4	73.9	54.1	50.4	54.0	73.6	71.0	56.8	53.4
w/ Focus	0	31.5	31.3	33.0	32.6	33.4	32.2	32.6	46.4	52.4	49.0	**56.6**	44.6	47.3	48.2	47.4	48.3	46.8	51.1	42.5
	4	45.1	37.7	35.3	33.4	35.0	38.1	33.8	58.8	53.8	51.6	53.2	63.2	50.0	46.7	**54.5**	61.7	62.5	52.2	48.1
w/ UnifyVocab	0	**50.9**	37.6	34.3	34.6	33.7	33.1	33.7	60.2	52.6	48.0	55.8	63.1	47.1	47.0	50.3	59.6	48.6	51.4	46.8
	4	46.8	**39.1**	**37.3**	**37.7**	**38.0**	**42.5**	**34.9**	**73.2**	**55.6**	**54.6**	53.4	**73.1**	**53.9**	**49.2**	54.0	**74.0**	**63.3**	**56.7**	**52.1**

Case study of multilingual token alignment. Table 8 provides nine new tokens from three languages with their top 3 tokens in the source vocabulary. In most cases, a clear semantic relationship between two aligned tokens cannot be found. We argue that it may come from the following two reasons:

Table 8: The case study of new tokens from other languages in the target vocabulary with top-3 source tokens aligned. The language family of French, Chinese, and Korean are Indo-European, Sino-Tibetan, and Koreanic, respectively.

	French			Chinese			Korean		
Top-3	**dire(speak)**	**aller(go)**	**oui(are)**	**吃(eat)**	**科学(science)**	**智能(intelligence)**	**능(competence)**	**집(house)**	**왜(why)**
				Qwen2 (Target Tokenizer)					
1	ada	Ġsta	Ġsalv	allel	Ġantagon	_{[	Si	ĠBart	bst
2	ays	ĠÃ¨	Ġvas	Ġindicator	Ġign	liquid	uria	ĠPAT	rains
3	Ġ-	Ġdetermin	Ġexplos	Ġbasic	Ġcritic	Layer	ost	ĠEdgar	irc
				Gemma (Target Tokenizer)					
1	Ġj	Cor	Tools	kernel	ĠLed	Ġcommittee	Ġmang	Ġcru	Ġcholesterol
2	Ġdar	Ġequality	directed	sentence	COUNT	ĠUND	ial	Ġcal	Ġmolecule
3	ba	Lex	afx	messages	Ġglycine	Ġfactors	Ġrebut	Ġmalt	apor

- BPE algorithm (Sennrich et al., 2016) divides words into the sub-word units, also called tokens, from the statistical co-occurrence information. There may be less superficial semantic information in the tokens divided compared with words in the natural language.
- The GloVe vector for each token is obtained from the token-token co-occurrence information. These aligned tokens often appear together, e.g., 科学(science) and "Ġcritic", 왜(why) and "rains".

Therefore, it is better to choose a matric to quantify the performance of the alignment matrix learned, for example, the BLEU score in Section 2.2 or the perplexity of the initialized model.

C Language Codes

We provide details of languages involved in Table 9. Following Lai et al. (2023), languages are divided by the data ratios in CommomCrawl: High (>1%), Medium (>0.1%), and Low (>0.01%).

Table 9: Details of Language codes in this work.

ISO 639-1	Language	Family	ISO 639-1	Language	Family
AR	Arabic	Afro-Asiatic	TA	Tamil	Dravidian
BN	Bengali	Indo-European	TE	Telugu	Dravidian
DE	German	Indo-European	TH	Thai	Kra-Dai
EN	English	Indo-European	UR	Urdu	Indo-European
JA	Japanese	Japonic	VI	Vietnamese	Austroasiatic
KO	Korean	Koreanic	ZH	Chinese	Sino-Tibetan