Investigating and Scaling up Code-Switching for Multilingual Language Model Pre-Training

Zhijun Wang[1,2], Jiahuan Li[2], Hao Zhou[1], Rongxiang Weng[2], Jingang Wang[2], Xin Huang[3], Xue Han[3], Junlan Feng[3], Chao Deng[3], Shujian Huang[1]*

[1]National Key Laboratory for Novel Software Technology, Nanjing University, China
[2]Meituan Inc. [3]China Mobile Research Beijing, China
{wangzj,zhouh}@smail.nju.edu.cn,{huangsj}@nju.edu.cn
{lijiahuan04,wengrongxiang,wangjingang02}@meituan.com
{huangxinyjy,hanxueai,fengjunlan,dengchao}@chinamobile.com

Abstract

Large language models (LLMs) exhibit remarkable multilingual capabilities despite the extreme language imbalance in the pre-training data. In this paper, we closely examine the reasons behind this phenomenon, focusing on the pre-training corpus. We find that the existence of code-switching, alternating between different languages within a context, is key to multilingual capabilities. We conduct an analysis to investigate code-switching in the pre-training corpus, examining its presence and categorizing it into four types within two quadrants. We then assess its impact on multilingual performance. These types of code-switching data are unbalanced in proportions and demonstrate different effects on facilitating language transfer. To better explore the power of code-switching for language alignment during pre-training, we investigate the strategy of synthetic code-switching. We continuously scale up the synthetic code-switching data and observe remarkable improvements in both benchmarks and representation space. Extensive experiments indicate that incorporating synthetic code-switching data enables better language alignment and generalizes well to high, medium, and low-resource languages with pre-training corpora of varying qualities. Code and scripts are freely available at `https://github.com/NJUNLP/SynCS`.

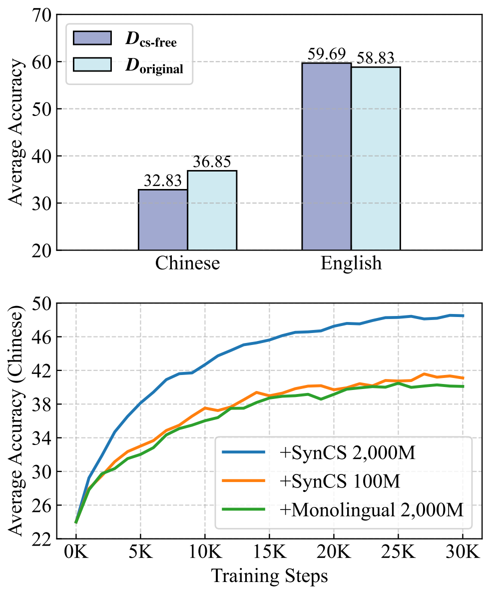

Figure 1: Performance of models pre-trained on language-imbalance data (60B En and 600M Zh, 100:1). $\mathcal{D}_{\text{cs-free}}$ in the upper sub-graph means the natural code-switching is removed. "+SynCS" and "+Monolingual" in the lower sub-graph denote adding synthetic code-switching data and monolingual data, respectively. The numbers represent newly added Zh tokens.

1 Introduction

Large Language Models (LLMs) such as ChatGPT (OpenAI, 2023), GPT-4 (Achiam et al., 2023), Llama2 (Touvron et al., 2023), Llama3 (Dubey et al., 2024), and Qwen2.5 (Yang et al., 2024) have demonstrated remarkable performance across various tasks, including multiple-choice question-answering (Robinson and Wingate, 2023), summarization (Pu et al., 2023), and reasoning (Yu et al., 2023). Meanwhile, LLMs also demonstrate excellent multilingual capabilities. Among them, some models are pre-trained on corpora not specifically designed for multilingual use (Touvron et al., 2023), while others are pre-trained on corpora containing only a small fraction of multilingual data (Dubey et al., 2024). Despite the extreme language imbalance in the pre-training corpus (Ranta and Goutte, 2021), LLMs demonstrate impressive cross-lingual transfer to some extend (Pires et al., 2019; Kargaran et al., 2024). This raises the question: where do these cross-lingual transfers come from?

Code-switching, also known as code-mixing or language alternation, is the process of alternating between two or more languages in a single conversation (Thara and Poornachandran, 2018). This type of data puts concepts from different languages within the same context, creating favorable condi-

*The Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2025, pages 11032–11046
July 27 - August 1, 2025

Category	Example
Sent-Annt.	Now depending on where you shop in China, sometimes you need to bargain for what you are buying. Mike, the fruits stand is just ahead, let's buy some fruit OK? (麦克，前面有一个水果摊，我们买点儿水果吧。)
Sent-Repl.	Can you name some traditional Chinese festivals? Do you like them? Why? 这道题的目的是要求考生陈述出来传统文化的重要性。 Traditional cultures should be protected. because first...... [The Chinese sentence means "The purpose of this question is to require candidates to state the importance of traditional culture."]
Token-Annt.	The customs of the spring festival: 1. Putting up Spring Couplet (贴春联) and Burning Firecrackers (放鞭炮).
Token-Repl.	You can use the above picture and add some related words, such as 剃须刀、字典、镜子、毛巾、冰箱、微波炉、电脑 and 书橱. Classify these words and fill in the table. [These Chinese words mean "razor", "dictionary", "mirror", "towel", "refrigerator", "microwave", "computer", and "bookcase", respectively.]

Table 1: Examples of code-switching types in FineWeb-Edu. For annotation types, annotations are typically placed in parentheses, with the annotated text underlined. For replacement types, code-switching occurs within the original text, and explanations are appended in brackets after the example.

tions for potential language transfer learning in LLMs. Many works attempt to leverage code-switching on multilingual tasks. Yoo et al. (2024a); Li et al. (2024b) reveal the effects of synthetic code-switching data in cross-lingual transfers. Briakou et al. (2023) investigate the incidental bilingualism in the unreasonable translation capabilities of LLMs. However, there is a lack of detailed analysis of code-switching in multilingual pre-training.

To investigate the effects of code-switching, we pre-train a 1.5B model on 60B tokens with extreme language imbalance (100:1). Taking English and Chinese as the high and low-resource language examples, we initially explore the natural code-switching phenomenon of two high-quality pre-training corpora. We conduct a model-based method to analyze and categorize four common code-switching types. Subsequently, we conduct experiments to assess the impact of various code-switching on cross-lingual transfer.

Building on this analysis, we propose to enhance the advantages of code-switching by incorporating synthetic code-switching data during pre-training, valued for its controllability and flexibility. Through a series of scaling experiments, synthetic code-switching (SynCS) significantly improves cross-lingual transfer, outperforming the addition of 20 times the amount of monolingual data with natural code-switching. Further analysis shows that models trained on SynCS data obtain improved multilingual alignment in the representation space. Finally, we expand our approach to multilingual settings, encompassing high, medium, and low-resource languages, showcasing the generalization of SynCS across languages.

In summary, our findings are:

- Natural Code-Switching in Pre-Training Data: In FineWeb-Edu (Penedo et al., 2024), 0.4% of documents contain English-Chinese code-switching, compared to 51.6% in Chinese-FineWeb-Edu-v2 (Yu et al., 2025). These instances, categorized into four types, enhance multilingual transfer despite their imbalance.

- Role of Natural Code-Switching: Natural code-switching plays a crucial role in facilitating cross-lingual transfer. As illustrated in Figure 1, models trained without it experience a notable performance drop.

- We introduce SynCS, a flexible framework for synthesizing code-switching with precise control over density and magnitude. Models trained with SynCS exhibit superior multilingual alignment, surpassing the performance achieved by adding 20x monolingual data, as shown in Figure 1.

2 Measuring Code-Switching

2.1 Categorizing Code-Switching

Based on our empirical analysis, code-switching segments are categorized into **Sentence-Level** and **Token-Level**, each further divided into **Annotation** and **Replacement**. Considering languages A and B, the code-switching types are defined as follows:

- **Sentence-Level-Annotation** (denoted as **Sent-Annt.**): In a continuous sequence of sentences in the context of language A, some sentences are annotated by their translation in language B, commonly appearing in parentheses. The semantics represented by these sentences appear in both languages A and B.

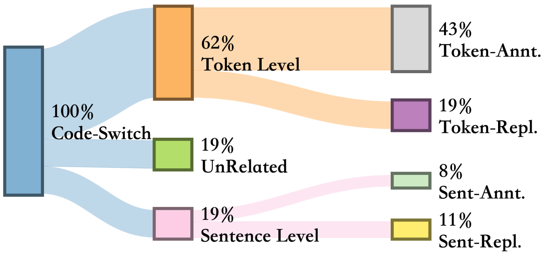

Figure 2: Distribution of different types of En->Zh code-switching in FineWeb-Edu.

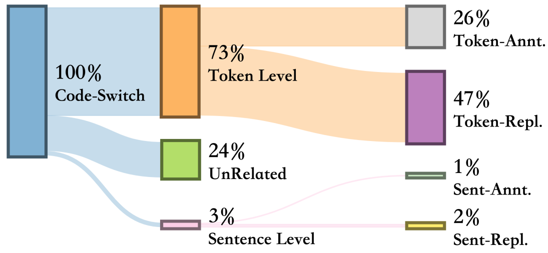

Figure 3: Distribution of different types of Zh->En code-switching in Chinese-FineWeb-Edu-v2.

- **Sentence-Level-Replacement** (denoted as **Sent-Repl.**): In a continuous sequence of sentences in the context of language A, some sentences are replaced by their translation in language B. The semantics represented by these sentences appear only in language B.
- **Token-Level-Annotation** (denoted as **Token-Annt.**): In a sentence of language A, some tokens are annotated by their translation in language B, commonly appearing in parentheses. The concepts represented by these tokens appear in both languages A and B.
- **Token-Level-Replacement** (denoted as **Token-Repl.**): In a sentence of language A, some tokens are replaced by their translation in language B. The concepts represented by these tokens appear only in language B.

Table 1 presents examples for each type of Chinese code-switching in English data. In our following discussions, "Code-Switching in A" refers to containing text of B in the context of A.

2.2 Detecting Code-Switching Segments

To investigate the characteristics of natural code-switching, we need first detect all code-switching segments. Code-switching in both high-resource and low-resource languages can enhance cross-lingual transfer, so we begin by identifying code-switching documents in each language dataset. These documents are segmented into sentences, with each sentence tagged by its language. Sentences entirely in one language, differing from the document's language, are classified as sentence-level code-switching, while sentences incorporating both languages are considered token-level code-switching.

The strategy for classifying segments into Annt. and Repl. differs between sentence-level and token-level code-switching. For sentence level, this process is indeed identifying translation pairs. We employ a cross-lingual encoder to find semantically aligned sentence pairs, following Briakou et al. (2023). For token level, we leverage SOTA LLMs to classify. Additionally, we use LLMs to detect unrelated code-switching segments, which may result from nonsensical content or language recognition errors (such as text of Japanese).

To simplify our analysis, we choose to explore the Chinese and English code-switching data which has completely different scripts. We choose two high-quality corpora: FineWeb-Edu (Penedo et al., 2024) and Chinese-FineWeb-Edu-v2 (Yu et al., 2025). More details are illustrated in section A.1.

2.3 Counting Code-Switching Segments

We calculate the ratio of different code-switching types at the segment granularity.

In FineWeb-Edu, 0.4% of documents contain Chinese-English code-switching. Figure 2 shows the distribution of different types. 19% code-switching segments fall under unrelated, most of which are segments containing characters of Japanese or nonsense text. In the remaining 81% code-switching documents, the main type is token-level (62%), among which Annt. accounts the most (43%). For sentence-level code-switching, the proportion of Annt. and Repl. are similar. Examples of each type are illustrated in Table 1 and section B.

In Chinese-FineWeb-Edu-v2, 51.2% of documents contain Chinese-English code-switching. Figure 3 demonstrates the distribution. 24% are unrelated. The proportion of sentence-level code-switching is very small, approximately 3%, with 1% being Annt. and the rest 2% being Repl. In contrast to FineWeb-Edu, the Token-Repl. code-switching accounts more than the Token-Annt. code-switching. This is caused by the frequent use of proper noun, such as "Microsoft", "CAR-T" (Chimeric Antigen Receptor T-Cell) and so on. Examples of each type are illustrated in section B.

Data	En		Zh		MEXA
	PPL	Acc. Avg.	PPL	Acc. Avg.	
$\mathcal{D}_{\text{original}}$	11.3	58.8	41.2	36.9	0.66
$\mathcal{D}_{\text{control}}$	11.3	59.1	40.5	37.9	0.66
$\mathcal{D}_{\text{cs-free}}$	11.4	59.7	66.0	32.8	0.43

Table 2: Comparison of Chinese performance of models trained on different data. "Acc. Avg." is the average accuracy on Hellaswag and ARC-Easy.

3 Analyzing the Impact of Code-Switching

Based on our analysis of natural code-switching, we investigate its impact on cross-lingual transfer.

3.1 Experiment Setup

Pre-Training Recipes We sample 60B English tokens from FineWeb-Edu and 600M Chinese tokens from Chinese-FineWeb-Edu-v2 to simulate the language imbalance (100:1) pre-training[1]. A 1.5B model is trained from scratch on this sampled data to explore the cross-lingual transfer during pre-training. The hyper-parameters for pre-training are detailed in section D.

Evaluation Recipes We use the perplexity on Wikipedia (Foundation), and the accuracy on Hellaswag (Zellers et al., 2019) and ARC-Easy (Clark et al., 2018) to evaluate the performance in each language. Besides, we present MEXA (Kargaran et al., 2024) scores, which assess alignment between English and non-English languages using parallel sentences to evaluate language transfer. More evaluation details are illustrated in Section D.

3.2 Ablating All Code-Switching

We employ a document-substitute-based ablating method. Let $\mathcal{M}$ denote the documents used for pre-training and $\mathcal{P}$ denote the homologous holdout documents, where the partitions are defined as:

$$\mathcal{M} = \mathcal{M}_{\text{wcs}} \cup \mathcal{M}_{\text{wocs}}, \quad \mathcal{P} = \mathcal{P}_{\text{wcs}} \cup \mathcal{P}_{\text{wocs}}$$

where "wcs" and "wocs" refer to documents with and without code-switching respectively.

To investigate the overall impact of code-switching, we construct the code-switching-free dataset $\mathcal{D}_{\text{cs-free}}$ through document substitution:

$$\mathcal{D}_{\text{cs-free}} = \mathcal{M}_{\text{wocs}} \cup \mathcal{S}$$

[1]We follow Li et al. (2024b)'s language imbalance pre-training settings.

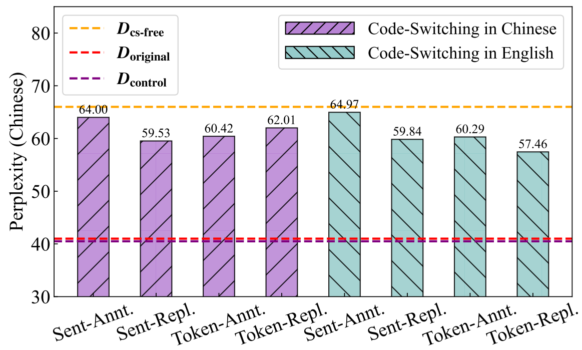

Figure 4: Impact of different types of natural code-switching on the cross-lingual transfer.

where $\mathcal{S}$ is a randomly sampled subset from $\mathcal{P}_{\text{wocs}}$ satisfying $\mathcal{S} \subseteq \mathcal{P}_{\text{wocs}}$ and $|\mathcal{S}| = |\mathcal{M}_{\text{wcs}}|$ to maintain equivalent corpus size.

To control for potential confounding factors from newly introduced documents $\mathcal{S}$, we further build a control dataset $\mathcal{D}_{\text{control}}$:

$$\mathcal{D}_{\text{control}} = (\mathcal{M} \setminus \mathcal{T}) \cup \mathcal{S}$$

where $\mathcal{T}$ is a randomly sampled subset from $\mathcal{M}_{\text{wocs}}$ with $|\mathcal{T}| = |\mathcal{S}|$ to maintain equivalent corpus size.

Natural Code-Switching Plays a Crucial Role in Cross-Lingual Transfer In Table 2, the perplexity of the model trained on $\mathcal{D}_{\text{cs-free}}$ shows a significant increase compared to that of $\mathcal{D}_{\text{control}}$ (40.5 to 66.0), and the benchmark performance also decreases by about 5 points. Without natural code-switching, the MEXA alignment score of the model drops significantly (0.66 to 0.43), indicating a worse multilingual alignment in hidden states. These results reveal the importance of natural code-switching in cross-lingual transfer.

3.3 Ablating Individual Type

To further investigate the impact of code-switching in various formats, we conduct experiments trained on data containing only one type. Since the ablation for each type shows an imperceptible difference in benchmarks, we mainly report the perplexity. Figure 4 demonstrates the results.

Less Tokens but Better Transfer For Repl. code-switching in Chinese, the number of tokens in Chinese is actually decreasing from the original 600M since some tokens are replaced by its translation. However, leveraging Repl. code-switching can still reduce the perplexity, indicating the potential cross-lingual transfer. Sent-Repl. presents the best effects on cross-lingual transfer, even though it only accounts for 2%.

Repl. Contributes More than Annt. For code-switching in English, Repl. demonstrates better effects than Annt., as shown in Figure 4. We suppose that while the concepts represented by code-switched tokens appear twice in both languages in Annt., the model may pay less attention to the Chinese tokens during training. This process may degrade the potential transfer learning.

Translation Fails in Enhancing Multilingual Transfer It is worth noting that Sent-Annt. in both English and Chinese, show the worst effects compared to other types. This suggests that while parallel sentences in the pre-training corpus are crucial for the model's translation capabilities (Briakou et al., 2023), they may not significantly enhance multilingual transfer.

4 Scaling up Code-Switching

Despite the effectiveness evidenced in the experiment of previous section, the natural code-switching phenomenon is rare and usually restricted to specific domains. In this section, we explore improving multilingual pre-training by synthesizing large-scale documents with code-switching. This method is more flexible and controllable, allowing us to inject code-switching into any document at any density and in any format.

4.1 Code-Switching Synthesis Pipeline

Given a collection of documents, we first split them into sentences and randomly select sentences to apply different types of code-switching.

Synthesizing Sentence-level Code-switching For sentence-level code-switching, we use TowerInstruct (Colombo et al., 2024) to translate each selected sentence. When conducting Sent-Repl., the source sentence is directly replaced with its translation. When conducting Sent-Annt., the source sentence is preserved with its translation following behind in parentheses, which is the most frequent pattern for natural Sent-Annt.

Synthesizing Token-level Code-Switching Currently, there is a lack of flexible and low-cost methods for synthesizing high-quality token-level code-switching. Li et al. (2024b) conduct rule-based method using a bilingual dictionary. However, it suffers from the one-to-many problem of word alignment and fails to select suitable tokens to replace or annotate. Yoo et al. (2024a) leverages GPT-4o and parallel sentences to synthesize high quality Token-Repl. code-switching data. However, it is expensive when scaling up and can not be used on monolingual documents. Empirically, we also find that SOTA LLMs struggle to generate token-level code-switching content given only monolingual text.

To synthesize high-quality token-level code-switching without requiring parallel sentences at a low-cost, we introduce a data-based distillation method. Initially, inspired by Yoo et al. (2024a), we leverage GPT-4o-mini to generate high-quality Token-Annt. and Token-Repl. code-switching data based on parallel sentences. Then we construct Supervised Fine-Tuning (SFT) data by only preserving the sentence of one language in the instruction, resulting in a multilingual dataset. A small language model is then fine-tuned on this dataset, learning to synthesize token-level code-switching. Practically, we select Qwen2.5-3B-Instruct as the base model, taking both speed and effect into consideration. The resultant model can rapidly generate diverse and high-quality code-switching data at a low cost. The prompts for generating SFT data and fine-tuning are illustrated in section C.1.

4.2 Scaling up Code-Switching in Chinese

To assess whether scaling on the low-resource language enhances cross-lingual transfer, we modify the 600M Chinese documents to include English code-switching segments. In Figure 5, we increase the number of newly added English tokens from 0M to 500M by adjusting the ratio of modified sentences.

Improved Cross-Lingual Transfer with Code-Switching scaling in Chinese As we modify more sentences from the 600M Chinese documents, the performance in Chinese continues to improve. Adding 300M new English tokens results in significant improvements (39.99 vs 36.85). This demonstrates that SynCS in the Chinese effectively enhances cross-lingual transfer.

The Importance of Chinese Monolingual Data Beyond 300M, all four types of code-switching in Chinese exhibit a notable performance drop. This decline is due to excessive alterations of the original Chinese documents as we modify over 60% of the sentences. This highlights the importance

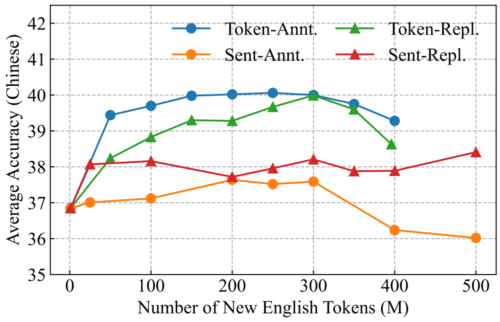

Figure 5: Scaling code-switching in Chinese: Average accuracy on Hellaswag and ARC-Easy in Chinese.

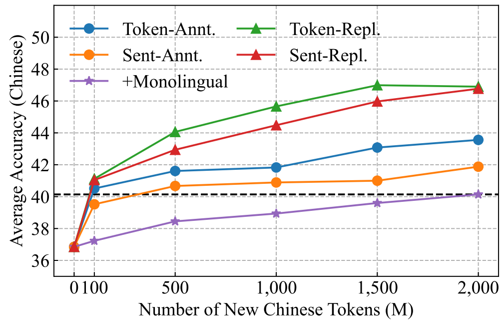

Figure 6: Scaling code-switching in English: Average accuracy on Hellaswag and ARC-Easy in Chinese.

of retaining the low-resource language monolingual data. Notably, even with 100% modification, Token-Annt. still presents substantial improvements (+2.43).

Token-Level Code-Switching Exceeds Sentence-Level In Figure 5, Token-Annt. and Token-Repl. consistently exceeds Sent-Annt. and Sent-Repl., with a maximum gap of 1.58 points. The scalability of sentence-level code-switching in Chinese appears to be limited, suggesting that token-level code-switching is more suitable for the low-resource language.

4.3 Scaling up Code-Switching in English

Since code-switching in English increases the token count of Chinese, we explore whether it exhibits better scalability. We modify only 20% of the documents (12B) to ensure stable English learning. In Figure 6, we increase the number of newly added Chinese tokens from 0M to 2,000M by adjusting the ratio of modified sentences.

Greater Efficiency of Code-Switching in English The results show the advantages of code-switching in English compared to Chinese. By adding 100M new tokens, the performance of code-switching in English exceeds that of Chinese by 1.42 points. This gap increases with over 100M tokens, reaching a maximum of 6.93 points. As English dominates during pre-training, it allows for extensive code-switching scaling without reducing low-resource langauge tokens.

Superior Scalability of Code-Switching in English By scaling the newly added Chinese tokens from 0M to 2,000M, SynCS demonstrates improvements from 0 to 10.14. This showcases its superior scalability. In experiments comparing the addition of an equivalent amount of Chinese monolingual tokens from holdout documents, SynCS consistently demonstrates superior performance. At 100 M, SynCS matches or surpasses the performance achieved by adding 20x monolingual data at 2,000M, highlighting its remarkable efficiency.

Replacement Transfers Better than Annotation Figure 6 shows that Sent-Repl. and Token-Repl. outperform Sent-Annt. and Token-Annt. with faster performance improvements. This is consistent with the ablation study of natural code-switching in section 3.3, which indicates that Repl. in English enhances multilingual performance more than Annt. Figure 7 presents the t-SNE visualizations (Van der Maaten and Hinton, 2008) of parallel sentences' middle-layer hidden states for models trained on SynCS data of different types. Notably, only Token-Repl. and Sent-Repl. exhibit significant changes, suggesting a more comprehensive cross-lingual transfer process through evenly mixed representations of parallel sentences.

4.4 Bring All Together

To investigate potential mutual promotion effects between different code-switching types and identify the optimal mixing strategy, we merge all types in both English and Chinese. For simplicity, code-switching of type X in language L is denoted as "L-X". "En-T-Repl." refers to conduct Token-Level-Replacement code-switching in the English data. Under the 500M and 2,000M budgets explored in the scaling experiments, we implement the following heuristic mixing strategies:

- Equal: In each language, four types of code-switching are evenly mixed.
- Extreme: In each language, the most powerful type of code-switching is used at its optimal

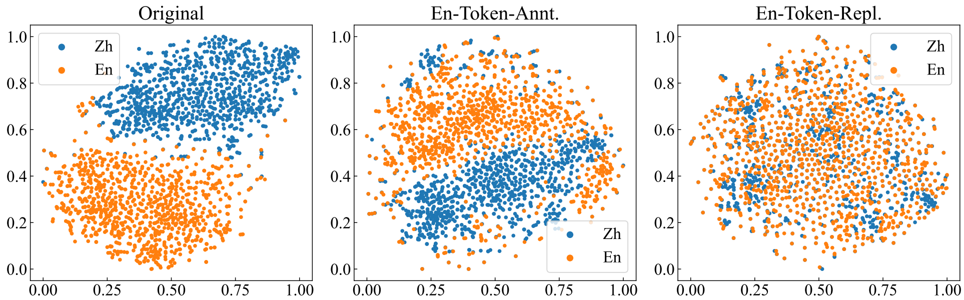

Figure 7: T-SNE visualization of parallel sentences' middle-layer hidden states shows significant changes only in En-Token-Repl. and En-Sent-Repl, as illustrated in Figures 9. We take En-Token-Annt. and En-Token-Repl. as examples here.

Data	# New Tokens	English				Chinese				
		PPL ↓	ARC-E	Hellaswag	Acc. Avg.	PPL ↓	GK.	NLU	Reasoning	Acc. Avg.
1.5B Pre-Trained Model (60B En + 600M Zh)										
Original Data	0M	11.3	66.9	50.7	58.8	41.2	29.8	52.8	41.6	41.4
+Monolingual	2,000M	**11.2**	**68.5**	50.0	59.3	**29.0**	31.0	54.8	43.2	43.0
+SynCS										
En-Token-Repl.	100M	11.3	67.9	**50.8**	59.3	38.5	30.8	55.4	43.0	43.1 (+0.0)
En-Token-Repl.	2,000M	11.4	68.1	50.2	59.1	35.0	31.5	55.4	47.6	44.9 (+1.9)
Equal	2,000M	11.8	68.2	49.9	59.1	40.5	30.6	56.1	46.9	44.5 (+1.5)
Extreme	2,000M	11.6	67.9	50.3	59.1	36.4	30.7	56.2	47.4	44.7 (+1.7)
En-Repl. Equal	2,000M	11.4	68.4	50.3	**59.4**	34.1	**31.7**	**57.4**	**47.9**	**45.7 (+2.7)**
7B Pre-Trained Model (120B En + 12B Zh)										
+Monolingual	2,000M	**5.8**	72.4	**59.3**	65.9	**16.3**	36.3	60.9	50.8	49.3
+SynCS	2,000M	5.9	**72.9**	58.9	**65.9**	18.2	**37.1**	**62.7**	**53.4**	**51.0**

Table 3: Evaluation results of different code-switching mixing strategies. "En-Token-Repl." represents Token-Level-Replacement code-switching in English, which performs the best in the scaling experiments.

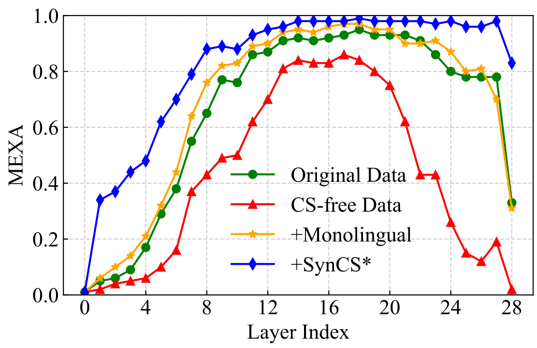

Figure 8: The MEXA alignment score comparison.

scale (En-Token-Repl. at 2,000M, and Zh-T-Annt. at 200M).

- En-Repl. Equal: En-Token-Repl. and En-Sent-Repl. are evenly mixed with each at the 1,000M scale, derived from their superior performance in the scaling experiments.

We expand our evaluation to three dimensions: General Knowledge (GK.), Natural Language Understanding (NLU), and Reasoning with each containing 4 benchmarks (Kydlíček et al.). Details are illustrated in section D. Table 3 presents the results.

SynCS Achieves 20x the Efficiency of Monolingual Data SynCS-Equal leads to a significant improvement (+3.16) and substantially outperforms adding an equal amount of monolingual data with natural code-switching (+1.52). Using the best En-Token-Repl. type at the 100M scale even demonstrates comparable performance to adding 20x monolingual data (43.1 vs 43.0).

Mixing SynCS in Both Languages Brings No Improvement Results show that SynCS-Equal and SynCS-Extreme demonstrate a slight decrease compared to En-Token-Repl., indicating that mixing SynCS in both languages does not yield significant mutual promotion effects.

The Most Two Powerful Types Promote Each Other En-Repl. Equal showcases substantial improvements over other mixing strategies. Its performance outperforms each of its composition types at the same scale, indicating the potential mutual promotion effects. We use this strategy as our final method in the following experiments, denoted as

Data	# New Tokens	English				Chinese			
		PPL ↓	Hellaswag	ARC-E	Acc. Avg.	PPL ↓	Hellaswag	ARC-E	Acc. Avg.
Original Data	0M	**13.6**	**48.4**	**67.8**	**58.1**	60.0	33.9	49.1	41.5
+Monolingual	3,000M	13.7	48.2	66.4	57.3	**50.1**	34.6	52.2	43.4
+SynCS*	150M	13.8	48.4	66.5	57.4	58.6	35.1	52.5	43.8
+SynCS*	3,000M	14.2	46.8	65.3	56.1	56.1	**37.2**	**56.3**	**46.7**
Data	**# New Tokens**	**Romanian**				**Bengali**			
		PPL ↓	**Hellaswag**	**ARC-E**	**Acc. Avg.**	**PPL ↓**	**Hellaswag**	**ARC-E**	**Acc. Avg.**
Original Data	0M	9.8	30.9	33.9	32.4	9.7	27.0	28.9	28.0
+Monolingual	3,000M	**8.6**	32.0	35.6	33.8	**7.9**	27.6	31.5	29.6
+SynCS*	150M	9.2	30.9	37.1	34.0	8.6	27.8	30.1	29.0
+SynCS*	3,000M	8.7	**32.5**	**40.7**	**36.6**	8.2	**28.1**	**32.6**	**30.3**

Table 4: Evaluation results in the multilingual setting.

SynCS*. Figure 8 shows the MEXA alignment scores. SynCS* significantly enhances MEXA alignment across all layers, particularly in shallow and deep layers, whereas monolingual data exhibits a slower, natural alignment process.

Larger Scale Still Works To investigate the effectiveness of SynCS at a larger scale, we conduct additional experiments by pre-training a 7B LLM on 132B tokens. We increase the Chinese token count to 12B, resulting in a language data ratio of 10:1. As shown in Table 3, SynCS demonstrates robust performance at this larger scale, yielding a significant improvement of 1.7 points. These results further validate the effectiveness and scalability of our approach.

5 Extend to Multilingual and DownStream Cross-Lingual Tasks

5.1 SynCS Generalizes to Other Languages

To assess SynCS's effectiveness in multilingual settings, we select Chinese, Romanian, and Bengali as representatives of high, medium, and low-resource languages. Details of the synthesis setup are in section C. The pre-training setup follows section D, except that the tokenizer is changed to DeepSeek-V3 (Liu et al., 2024) for improved tokenization of Romanian and Bengali. Due to the lack of benchmarks for Bengali and Romanian, we evaluate only on perplexity, Hellaswag, and ARC-Easy.

We first choose the same sentences at the 2,000M setting in our scaling experiments and evenly allocate them to these languages. Notably, the total number of new low-resource language tokens becomes 3,000M beacause of the different tokenization for languages. Table 4 presents that SynCS significantly outperforms the addition of an equivalent amount of monolingual documents across all three languages. Meanwhile, the 20x efficiency ratio still holds true on Romanian. For Bengali, SynCS presents comparable performance to its 20x monolingual data. This demonstrates the robust language generalization capabilities of SynCS.

Model	Flores (En->Zh)		ZS-CLT	
	BLEU	COMET	En	Zh
Mono	18.37	73.19	**80.46**	67.31
SynCS	**21.81**	**76.87**	80.38	**70.78**

Table 5: Evaluation results on translation task and ZS-CLT. "Mono" and "SynCS" refer to the models finetuned from "+Monolingual 2000M" and "En-Repl. Equal 2000M" in Table 3 respectively.

5.2 Investigation for DownStream Cross-Lingual Tasks

To assess whether pre-training on SynCS data enhances cross-lingual transfer capabilities in downstream tasks, we conducted experiments on translation and Zero-shot Cross-lingual Transfer (ZS-CLT).

For the translation task, the models are further fine-tuned using OPUS En->Zh data and evaluated on the Flores En->Zh translation task. We measure performance using sacreBLEU (Post, 2018) and COMET (Rei et al., 2022) [2]. For the ZS-CLT task, we chose XNLI (Conneau et al., 2018) as the training dataset, where pre-trained models were exclusively fine-tuned on the English train set of XNLI and subsequently evaluated on both English and Chinese test sets.

As illustrated in Table 5, our SynCS model delivers considerable improvements in translation and ZS-CLT tasks over models trained on monolingual data, indicating that pre-training with SynCS data augments the base model's cross-lingual transfer capabilities.

[2] We employed the wmt22-comet-da version.

6 Related Work

6.1 Cross-Lingual Transfer

Due to the imbalance of languages in the pre-training corpora, LLMs' multilingual abilities still show significant disparities (Bai et al., 2023; Dubey et al., 2024). Since addressing this language data imbalance is challenging (Ranta and Goutte, 2021), many efforts have been made to explore cross-lingual transfer in LLMs, which aim to transfer knowledge or reasoning capabilities from high-resource languages to low-resource languages. In the post-training stage, She et al. (2024) utilize response consistency between low- and high-resource languages to optimize and enhance LLMs' multilingual reasoning using DPO or PPO. Zhou et al. (2024) propose to prevent high-resource languages' catastrophic forgetting during continual pre-training for better low-resource language adaptation. In the pre-training stage, Dufter and Schütze (2020) identify shared parameters, subwords, and position embeddings as keys to transformer's multilingualism. Li et al. (2024b) argue that aligning multilingual representations before large-scale pre-training, followed by input-only code-switching, enhances multilingual capabilities.

6.2 Code-Switching

Code-switching, or language alternation, is a linguistic phenomenon where multilingual speakers use multiple languages within a conversation (Poplack, 1978). While LLMs exhibit strong multilingual capabilities, they struggle with code-switching tasks. Yoo et al. (2024b) show that code-switching attack prompts increase success rates. Code-switching aids multilingual alignment, as demonstrated by Li et al. (2024b), who use input-only code-switching during pre-training. Yoo et al. (2024a) introduce CSCL, a curriculum learning method using synthetic code-switching data to enhance multilingual alignment. Yoo et al. (2024a) is the most similar work to us. However, we focus on the pre-training stage, analyzing how natural code-switching enhances LLMs' multilingual capabilities and proposing a more flexible and less expensive code-switching synthesis approach.

7 Conclusion

This study explores the impact of code-switching on cross-lingual transfer during pre-training. We find that natural code-switching significantly enhances the multilingual capabilities of LLMs under extreme language imbalance. To address the scarcity of natural code-switching, we introduce a synthetic framework requiring only a small set of high-quality parallel sentences. Through extensive experiments and analysis, we demonstrate that this framework outperform those trained on equivalent monolingual data, improving performance across languages of varying resources.

8 Limitations

Due to the resource limit, our models fall under a 1.5B small language model trained on 60B tokens, which lacks generation abilities. Whether the findings in the paper hold on larger settings remains to be explored. Table 4 demonstrates that the improvement achieved on the low-resource language is not substantial because of the low-quality of the pre-training and synthetic code-switching data. How to generate high-quality code-switching data for these languages is a problem. Additionally, models trained with SynCS demonstrates worse performance on the Wiki-ppl compared to monolingual data, which may be handled by continue training on monolingual data or using the input-only code-switching (Li et al., 2024b). We leave these limitations for further work.

9 Acknowledgement

We would like to thank the anonymous reviewers for their insightful comments. Shujian Huang is the corresponding author. This work is supported by National Science Foundation of China (No. 62376116, 62176120), research project of Nanjing University-China Mobile Joint Institute, the Fundamental Research Funds for the Central Universities (No. 2024300507).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. arXiv preprint arXiv:2309.16609.

Eleftheria Briakou, Colin Cherry, and George Foster. 2023. Searching for needles in a haystack: On the role of incidental bilingualism in PaLM's translation capability. In Proceedings of the 61st Annual

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 9432–9452, Toronto, Canada. Association for Computational Linguistics.

Michael Chen, Mike D'Arcy, Alisa Liu, Jared Fernandez, and Doug Downey. 2019. CODAH: An adversarially-authored question answering dataset for common sense. In Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP, pages 63–69, Minneapolis, USA. Association for Computational Linguistics.

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

Pierre Colombo, Duarte Alves, José Pombal, Nuno Guerreiro, Pedro Martins, Joao Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, et al. 2024. Tower: An open multilingual large language model for translation-related tasks.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. arXiv preprint arXiv:2207.04672.

Tri Dao. 2024. FlashAttention-2: Faster attention with better parallelism and work partitioning. In International Conference on Learning Representations (ICLR).

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. arXiv preprint arXiv:2407.21783.

Philipp Dufter and Hinrich Schütze. 2020. Identifying elements essential for BERT's multilinguality. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4423–4437, Online. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Wikimedia Foundation. Wikimedia downloads.

Hai Hu, Kyle Richardson, Liang Xu, Lu Li, Sandra Kuebler, and Larry Moss. 2020. Ocnli: Original chinese natural language inference. In Findings of EMNLP.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In Advances in Neural Information Processing Systems.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 427–431. Association for Computational Linguistics.

Amir Hossein Kargaran, Ali Modarressi, Nafiseh Nikeghbal, Jana Diesner, François Yvon, and Hinrich Schütze. 2024. Mexa: Multilingual evaluation of english-centric llms via cross-lingual alignment. arXiv preprint arXiv:2410.05873.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles.

Hynek Kydlíček, Guilherme Penedo, Clémentine Fourier, Nathan Habib, and Thomas Wolf. Finetasks: Finding signal in a haystack of 200+ multilingual tasks.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 318–327, Singapore. Association for Computational Linguistics.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2024a. CMMLU: Measuring massive multitask language understanding in Chinese. In Findings of the Association for Computational Linguistics: ACL 2024, pages 11260–11285, Bangkok, Thailand. Association for Computational Linguistics.

Jiahuan Li, Shujian Huang, Aarron Ching, Xinyu Dai, and Jiajun Chen. 2024b. PreAlign: Boosting cross-lingual transfer by early establishment of multilingual alignment. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10246–10257, Miami, Florida, USA. Association for Computational Linguistics.

Bill Yuchen Lin, Seyeon Lee, Xiaoyang Qiao, and Xiang Ren. 2021. Common sense beyond English: Evaluating and improving multilingual language models for commonsense reasoning. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1274–1287, Online. Association for Computational Linguistics.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024. Deepseek-v3 technical report. arXiv preprint arXiv:2412.19437.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James F Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics, pages 46–51. Association for Computational Linguistics.

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. 2023. Crosslingual generalization through multitask finetuning. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

OpenAI. 2023. Chatgpt (mar 23 version) [large language model].

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. In Advances in Neural Information Processing Systems, volume 37, pages 30811–30849. Curran Associates, Inc.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 2362–2376, Online. Association for Computational Linguistics.

Shana Poplack. 1978. Syntactic structure and social function of code switching. Latino Discourse and Communicative Behavior/Ablex Publishing.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In Proceedings of the Third Conference on Machine Translation: Research Papers, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Xiao Pu, Mingqi Gao, and Xiaojun Wan. 2023. Summarization is (almost) dead. arXiv preprint arXiv:2309.09558.

Aarne Ranta and Cyril Goutte. 2021. Linguistic diversity in natural language processing. Traitement Automatique des Langues, 62(3):7–11.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In Proceedings of the Seventh Conference on Machine Translation (WMT), pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Joshua Robinson and David Wingate. 2023. Leveraging large language models for multiple choice question answering. In The Eleventh International Conference on Learning Representations.

Shuaijie She, Wei Zou, Shujian Huang, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. 2024. MAPO: Advancing multilingual reasoning through multilingual-alignment-as-preference optimization. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 10015–10027, Bangkok, Thailand. Association for Computational Linguistics.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. arXiv preprint arXiv:1909.08053.

S Thara and Prabaharan Poornachandran. 2018. Code-mixing: A brief survey. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pages 2382–2388.

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

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. Journal of machine learning research, 9(11).

Haoran Xu, Kenton Murray, Philipp Koehn, Hieu Hoang, Akiko Eriguchi, and Huda Khayrallah. 2024. X-alma: Plug & play modules and adaptive rejection for quality translation at scale. arXiv preprint arXiv:2410.03115.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. arXiv preprint arXiv:2412.15115.

Haneul Yoo, Cheonbok Park, Sangdoo Yun, Alice Oh, and Hwaran Lee. 2024a. Code-switching curriculum learning for multilingual transfer in llms. arXiv preprint arXiv:2411.02460.

Haneul Yoo, Yongjin Yang, and Hwaran Lee. 2024b. Code-switching red-teaming: Llm evaluation for safety and multilingual understanding. arXiv preprint arXiv:2406.15481.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. arXiv preprint arXiv:2309.12284.

Yijiong Yu, Ziyun Dai, Zekun Wang, Wei Wang, Ran Chen, and Ji Pei. 2025. Opencsg chinese corpus: A series of high-quality chinese datasets for llm training. arXiv preprint arXiv:2501.08197.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Wenxuan Zhang, Sharifah Mahani Aljunied, Chang Gao, Yew Ken Chia, and Lidong Bing. 2023. M3exam: A multilingual, multimodal, multilevel benchmark for examining large language models.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2024. AGIEval: A human-centric benchmark for evaluating foundation models. In Findings of the Association for Computational Linguistics: NAACL 2024, pages 2299–2314, Mexico City, Mexico. Association for Computational Linguistics.

Hao Zhou, Zhijun Wang, Shujian Huang, Xin Huang, Xue Han, Junlan Feng, Chao Deng, Weihua Luo, and Jiajun Chen. 2024. Moe-lpr: Multilingual extension of large language models through mixture-of-experts with language priors routing. arXiv preprint arXiv:2408.11396.

A Code-Switching Data Detecting

A.1 Detecting Details

We first apply a character-based filtering to obtain documents that contain English and Chinese. Then we use fasttext (Joulin et al., 2017) to classify each sentence as monolingual or bilingual, corresponding to sentence-level and token-level code-switching, respectively. We prompt Qwen2.5-72B-Instruct (Yang et al., 2024) to filter out the unrelated code-switching sentences. Each segment is then categorized as either Annt. or Repl..

For sentence level, classifying into Annt. and Repl. is indeed detecting the translation pairs. We employ LABSE (Feng et al., 2022) cross-lingual encoder to find semantic-align sentence pairs in two languages, following Briakou et al. (2023).

For token level, we use an LLM-based detection strategy to categorize. We prompt Qwen2.5-72B-Instruct with the instructions as following and ask for classification.

Prompts for Annotation and Replacement classification

Code-switching can be classified more finely according to different characteristics and uses. Here are some common types:

1. Annotation: In this case, another language is used to explain or define a noun before or after it. For example: During the festival, we watched a dragon dance (舞龙). In this sentence, the word "舞龙" serves as an annotation for "dragon dance".
2. Replacement: A specific word is replaced by a foreign word. For example: During the festival, we watched a 舞龙. In this sentence, the word "舞龙" replaces the English word "dragon dance".

Given an English sentence containing Chinese code-switching, please classify the sentence according to the above two types.
Examples:
[English Sentence]: During the festival, we watched a dragon dance (舞龙), which is a traditional Chinese performance.
[Answer]: "舞龙" appears after "dragon dance", which explains this English word in Chinese and is its annotation. Formatting result: \\box(1)

[English Sentence]: We enjoyed some delicious food at a nearby 茶馆.
[Answer]: The word "茶馆" is directly used as part of the sentence. It can be assumed that the original word is "teahouse", but it is directly replaced by "茶馆". Formatting result: \\box(2)

The following is your task. You can do a brief analysis, but please be sure to output it in the format of the example at the end.
[English Sentence]:
[Answer]:

B Examples for Various Natural Code-Switching Segments

English-Side Code-Switching

Unrelated:
1. ◇お客、こちらのブラウスですと、いまお召しのスッにもよく合いますが。
2. there are also the phrases いつ(about when?
3. 2 Polypodiaceae Tac ke 家Me.

Chinese-Side Code-Switching

Unrelated:
1. zxx520llc发表于: 2个月前#9
2. X$Gx170 水利图书F' q A t8̂t2G [Garbled]

T-Annt.:
1.比如盐酸(HCL)、硝酸。[Explanation in English: For example, hydrochloric acid (HCL) and nitric acid.]

T-Repl.:
1. Microsoft 商店很可能误解了你尝试下载或安装的应用程序。[Explanation in English: It's possible that the Microsoft Store misunderstood the app you were trying to download or install.]

S-Annt.:
1. 任何人都不太可能真正了解它的全部。These are the basic materials that go into a pencil, graphite, cedar, metal,and rubber。这些就是构成铅笔的基本材料，石墨、雪松、金属、橡胶。

S-Repl.:
1. 我只想引述GPT-4官方新闻的一句话：As a result, our GPT-4 training run was (for us at least!) unprecedentedly stable. [Explanation in English: I just want to quote a sentence from the official GPT-4 news: As a result, our GPT-4 training run was (for us at least!) unprecedentedly stable.]

C Code-Switching Data Synthesis

Synthesis Model Training Details We utilize 4 A100 GPUs and conduct multilingual and multi-task supervised fine-tuning on Qwen2.5-3B-Instruct. The model is fine-tuned for 3 epochs, using a context length of 2048 tokens, a warmup ratio of 0.1, and a peak of learning rate at 5e-5 with cosine decaying to 0. We utilize bf16 mixed precision and flash attention (Dao, 2024) to speed up the training process. We assign the temperature as 0 when generating code-switching data and translating sentences (i.e. greedy decoding). vLLM (Kwon et al., 2023) is used to accelerate the generation.

The source data for generating code-switching supervised fine-tuning data includes X-ALMA (Xu et al., 2024) and flores200 (Costa-jussà et al., 2022). While TowerInstruct doesn't support Bengali, we use NLLB (Costa-jussà et al., 2022) as the translator. As the data of Xu et al. (2024) doesn't contain Bengali, we directly use the dev and devtest set of the flores200 (Costa-jussà et al., 2022) dataset. Table 6 shows the number of parallel sentences in each language when generating the SFT data. We use the same data for the Annotation and Replacement types in both languages, resulting in a total of 62000 multilingual and multi-task SFT data. We directly reuse the prompts above except only the source language sentence is given.

C.1 Synthesis Prompts

When generating the token-level code-switching SFT data using GPT4o-mini, we follow and slightly modify the prompt of Yoo et al. (2024a) for better instruction-following.

Language Pairs	# of Parallel Sentences
English-Chinese	6906
English-Romanian	4987
English-Bengali	3604
Total	15500

Table 6: Number of parallel sentences used for generating token-level code-switching SFT data.

Prompts of Code-Switching Generation

Annotation (Target-Side as example):
Given a pair of {*Source Language*}-English parallel sentence, generate an English-annotated {*Source Language*} sentence. Annotation is the use of words from another language to explain certain words in a sentence.
[{*Source Language*} Sentence]:

Replacement:
Given a pair of {*Source Language*}-English sentence, generate a {*Source Language*} and English code-switching sentence. Code-switching is the use of more than one linguistic variety in a manner consistent with the syntax and phonology of each variety.
[{*Source Language*} Sentence]:

D Experiment Settings

Pre-Training Recipes We sample 60B English tokens from FineWeb-Edu and 600M Chinese tokens from Chinese-FineWeb-Edu-v2 to simulate the language-imbalance (100:1) pre-training. A 1.5B Qwen2.5 model (Yang et al., 2024) is trained on this sampled data to explore the cross-lingual transfer during pre-training. All models are trained for 30,000 steps with a batch size of 2M tokens. We group training documents with the length of 2048 and pre-training with global batch size of 1024. The learning rate performs cosine decay from 2e-4 to 5e-6 with 1% warmup. Experiments are conducted on the Megatron-LM (Shoeybi et al., 2019) framework. We use flash-attn (Dao, 2024) to accelerate training. Each experiment is trained on 128 A100s for 9 hours.

Evaluation Recipes We use the perplexity on Wikipedia (Foundation) and the finetasks (Kydlíček et al.) to evaluate our models. In finetasks, we choose the 12 tasks belonging to 3 dimensions:

- General Knowledge: AGI-Eval (Zhong et al., 2024), C-EVAL (Huang et al., 2023), CMMLU (Li et al., 2024a), M3Exams (Zhang et al., 2023).
- Natural Langauge Understanding: M-Hellaswag (Lai et al., 2023), Ocnli (Hu et al., 2020), X-winigrad (Muennighoff et al., 2023), Xstory-cloze (Mostafazadeh et al., 2017).
- Reasoning: Xcodah (Chen et al., 2019), XCOPA (Ponti et al., 2020), XCSPA (Lin et al., 2021), ARC-Easy (Clark et al., 2018).

The multilingual translated version of Hellaswag (Lai et al., 2023) is used. Since there is no multilingual version of ARC-Easy, we translate the original English version to Chinese, Romanian, and Bengali using GPT-4o-mini, following Lai et al. (2023). We also present MEXA (Kargaran et al., 2024) scores, which assess alignment between English and non-English languages using parallel sentences, flores200 (Costa-jussà et al., 2022), to evaluate language transfer. When we explore the natural code-switching and scaling up the synthetic code-switching, since the differences on these benchmarks are insignificant at a small scale, only perplexity, Hellaswag, and ARC-Easy are reported. Besides, in our multilingual settings, there are lack of evaluation benchmarks for Bengali and Romanian. We also only report these three results.

E T-SNE Visualization

Figure 9 demonstrates the T-SNE visualization of parallel sentences' middle layer hidden states for models trained on Chinese and English-side SynCS respectively. Only En-Token-Repl. and En-S-Repl. showcase obvious differences for mixing the representation space in two languages.

E.1 Detailed Evaluations

Table 7, 8, and 9 presents the detailed evaluations on each Chinese benchmarks mentioned at Table 3.

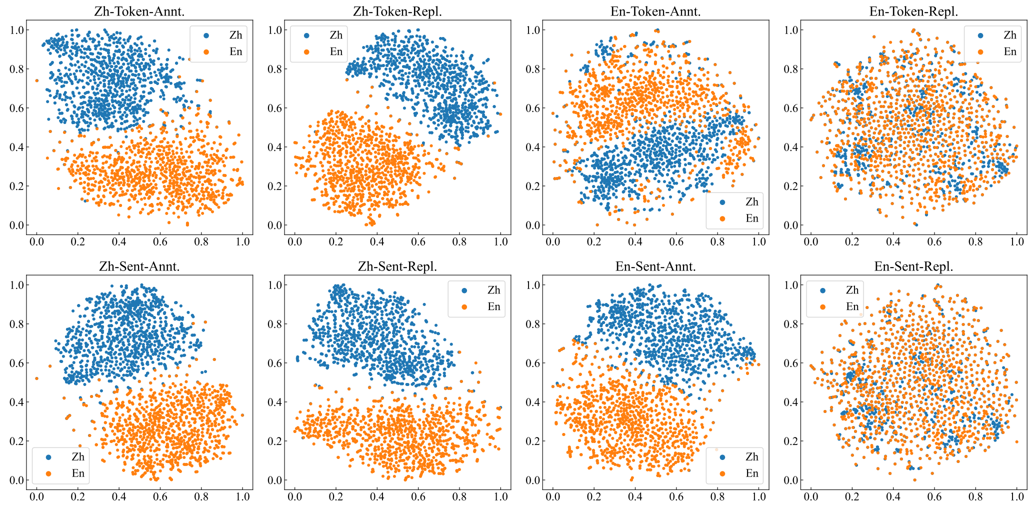

Figure 9: T-SNE visualization of parallel sentences' middle layer hidden states for models trained on Chinese-side and English-side SynCS.

Data	# New Tokens	AGI-Eval	CEVAL	CMMLU	M3Exams	Avg.
Original Data	0M	28.8	28.3	30.1	31.9	29.8
+Monolingual	2,000M	29.5	31.0	31.6	32.0	31.0
+SynCS						
En-Token-Repl.	100M	30.5	30.2	30.8	31.9	30.8
En-Token-Repl.	2,000M	30.7	31.3	31.8	32.3	31.5
Equal	2,000M	29.7	29.5	30.6	32.8	30.6
Extreme	2,000M	29.2	30.9	31.1	31.4	30.7
En-Repl. Equal	2,000M	30.5	29.9	31.5	35.1	31.7

Table 7: Chinese evaluation results on the General Knowledge (GK.) evaluation set.

Data	# New Tokens	AGI-Eval	CEVAL	CMMLU	M3Exams	Avg.
Original Data	0M	33.8	54.3	65.5	57.8	52.8
+Monolingual	2,000M	35.3	56.8	66.9	60.3	54.8
+SynCS						
En-Token-Repl.	100M	35.8	59.9	67.7	58.3	55.4
En-Token-Repl.	2,000M	39.7	55.2	68.9	57.7	55.4
Equal	2,000M	38.5	60.3	66.7	59.1	56.1
Extreme	2,000M	39.2	58.3	66.9	60.3	56.2
En-Repl. Equal	2,000M	40.1	62.4	66.9	60.2	57.4

Table 8: Chinese evaluation results on the Natural Language Understanding (NLU) evaluation set.

Data	# New Tokens	XCodah	XCOPA	XCSQA	ARC-Easy	Avg.
Original Data	0M	33.0	57.4	35.9	39.9	41.6
+Monolingual	2,000M	33.0	58.6	36.3	45.0	43.2
+SynCS						
En-Token-Repl.	100M	33.7	56.6	35.4	46.5	43.0
En-Token-Repl.	2,000M	35.7	61.8	39.0	54.1	47.6
Equal	2,000M	32.7	62.4	38.3	54.0	46.9
Extreme	2,000M	34.3	60.0	40.0	55.2	47.4
En-Repl. Equal	2,000M	33.3	61.4	40.0	56.8	47.9

Table 9: Chinese evaluation results on the Reasoning evaluation set.