MIGRATE: Cross-Lingual Adaptation of Domain-Specific LLMs through Code-Switching and Embedding Transfer

Seongtae Hong[1], Seungyoon Lee[1], Hyeonseok Moon[1], Heuiseok Lim[1,2‡]
[1]Department of Computer Science and Engineering, Korea University
[2]Human-inspired AI Research
[1,2]{ghdchlwls123,dltmddbs100,glee889,limhseok}@korea.ac.kr

Abstract

Large Language Models (LLMs) have rapidly advanced, with domain-specific expert models emerging to handle specialized tasks across various fields. However, the predominant focus on English-centric models demands extensive data, making it challenging to develop comparable models for middle and low-resource languages. To address this limitation, we introduce MIGRATE, a novel method that leverages open-source static embedding models and up to 3 million tokens of code-switching data to facilitate the seamless transfer of embeddings to target languages. MIGRATE enables effective cross-lingual adaptation without requiring large-scale domain-specific corpora in the target language, promoting the accessibility of expert LLMs to a diverse range of linguistic communities. Our experimental results demonstrate that MIGRATE significantly enhances model performance in target languages, outperforming baseline and existing cross-lingual transfer methods. This approach provides a practical and efficient solution for extending the capabilities of domain-specific expert models.

1 Introduction

Large Language Models (LLMs) have advanced natural language processing by demonstrating remarkable capabilities across various tasks and domains (OpenAI et al., 2024; Rozière et al., 2024). The development of domain-specific expert models has further expanded the potential of LLMs, enabling them to handle specialized terminology and complex concepts in fields such as science and mathematics (Azerbayev et al., 2024; Zhang et al., 2024; Taylor et al., 2022). However, these advancements have largely been centered around high-resource languages, particularly English, presenting significant challenges for middle and low-resource languages, which often lack the extensive domain-specific corpora required to train comparable models effectively (Nguyen et al., 2022). Consequently, speakers of these languages have limited access to powerful language technologies tailored to their linguistic and domain-specific needs.

Developing domain-specific expert models for underrepresented languages is difficult due to the scarcity of large-scale, high-quality datasets and the substantial computational resources required for training large models. Cross-lingual approaches, such as multilingual pre-training (Chi et al., 2021) or few-shot (Cahyawijaya et al., 2024), often do not capture the nuanced semantics and specialized terminology essential for expertise in specific domains, especially when dealing with languages with limited resources (Wu et al., 2022).

To address these challenges, we introduce MIGRATE, an effective method for migrating domain-specific expert models to target languages by enhancing monolingual static embeddings with code-switched data generated from the expert model's training corpus. Our approach facilitates embedding transfer without requiring large-scale target language corpora or significant computational resources, making it suitable for middle and low-resource languages. Specifically, we generate code-switched data by translating key nouns that often carry crucial domain-specific meanings and specialized terminology from the expert model's training corpus into the target language. By focusing on these nouns, we ensure that important vocabulary is represented in the target language, enriching the embedding space with cross-lingual lexical semantics. The code-switched data serves as a bridge between the source and target languages, and we transfer the enhanced embeddings to initialize the token embeddings of the target language in the expert model. This alignment of monolingual embeddings into a shared cross-lingual space enables the expert model to better understand and generate text

‡ Corresponding author

Proceedings of the 31st International Conference on Computational Linguistics, pages 9184–9193
January 19–24, 2025.

in the target language without extensive retraining or the need for large target language datasets.

We validate our method through extensive experiments in the science and mathematics domains, transferring expert models to Arabic, Bengali, German, Spanish, and Vietnamese. Our results demonstrate that MIGRATE significantly improves model performance in the target languages, outperforming baseline models and existing approaches. By integrating code-switched data during embedding enhancement, we enhance the model's multilingual capabilities and ensure that specialist terms and concepts are accurately represented in the target language. This targeted approach addresses the challenges of aligning domain-specific vocabulary across languages, enabling effective cross-lingual transfer.

In summary, MIGRATE provides a practical and resource-efficient solution for extending domain-specific expert models to underrepresented languages. By leveraging minimal code-switched data and existing resources, we promote inclusivity and enable the broader application of AI technologies across diverse linguistic landscapes. This work contributes to the democratization of AI, ensuring that advancements in language modeling are accessible to speakers of all languages, regardless of resource availability.

2 Related Work

Transferring language models to low-resource languages has been addressed through cross-lingual vocabulary transfer and embedding initialization strategies. Minixhofer et al. (2022) adapt pretrained models to target languages by replacing the tokenizer and initializing new token embeddings using multilingual static word embeddings, effectively transferring semantic knowledge without retaining source language capabilities. Dobler and de Melo (2023) build upon this by representing new target language tokens as combinations of overlapping source language tokens based on semantic similarity in an auxiliary embedding space, eliminating the need for bilingual dictionaries.

Remy et al. (2024) initialize target language embeddings using weighted averages of semantically similar source language embeddings, leveraging translation resources to adapt models to low-resource languages without extensive data. Additionally, Yamaguchi et al. (2024) explored vocabulary expansion with minimal target language text, emphasizing tailored strategies for low-resource settings, while Chirkova and Nikoulina (2024) investigated zero-shot cross-lingual transfer in instruction tuning, highlighting challenges such as reduced factual accuracy in target languages.

Our work introduces MIGRATE, which enhances monolingual embeddings with code-switched data generated from the expert model's corpus by translating key domain-specific nouns into the target language. This creates a code-switched corpus that accurately represents specialized terminology, facilitating effective cross-lingual transfer without large target language corpora or significant computational resources. Unlike previous methods, MIGRATE specifically addresses domain-specific vocabulary transfer, offering a practical solution for underrepresented languages in specialized domains.

3 Methods

To migrate a domain-specific expert model to the target language, we propose a simple and effective method, MIGRATE that leverages cross-lingual embedding transfer enhanced through code-switching. This approach involves two main stages: Enhancing static embeddings through code-switched data and Embedding transfer using cross-lingual static embedding. These stages aim to improve cross-lingual performance by systematically aligning monolingual embeddings into a shared cross-lingual space.

3.1 Enhancing Static Embeddings through Code-Switching

This stage comprises data preparation and model training processes to create enriched static embeddings for domain-specific cross-lingual transfer.

Data Preparation To enhance the cross-lingual performance of static embeddings, we generate a code-switched dataset from the training data used for the expert model. Let the training dataset be $\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$, where each document d_i is tokenized into sentences $S_i = \{s_{i1}, s_{i2}, \ldots, s_{iM_i}\}$, and M_i denotes the number of sentences in document d_i:

$$S_i = \text{sentence\_tokenize}(d_i), \quad \forall i \in [1, N] \quad (1)$$

We perform part-of-speech (POS) tagging on each sentence to identify all nouns:

$$T_{ij} = \text{pos\_tag}(s_{ij}), \quad \forall j \in [1, M_i] \quad (2)$$

Here, T_{ij} is the set of tuples (w, pos) representing words w and their POS tags in sentence s_{ij}. We extract all nouns[1] and denote the set of noun POS tags as NN. We then define N_{ij} as the set of nouns extracted from T_{ij}:

$$N_{ij} = \{n \mid (n, \text{pos}) \in T_{ij} \text{ and pos in NN}\} \quad (3)$$

We translate each noun $n \in N_{ij}$ into the target language and replace the original nouns with their translations to generate code-switched sentences:

$$s'_{ij} = s_{ij}[n \rightarrow \text{translate}(n) \mid n \in N_{ij}] \quad (4)$$

Our emphasis on nouns stems from their role as primary carriers of significant semantic information in domain-specific contexts. By focusing on nouns, which frequently embody the core terminology of a domain, we can facilitate effective cross-lingual alignment and preserve the semantic coherence of specialized vocabulary. For example, the English sentence "It is the question of uniqueness of empirical stratifications" transforms into the code-switched sentence "*It is the* ***câu hỏi*** *of* ***độc đáo*** *of empirical* ***phân tầng***", where the nouns have been replaced with their Vietnamese translations. The final code-switched dataset is:

$$\mathcal{D}' = \{s'_{ij} \mid i \in [1, N],\ j \in [1, M_i]\} \quad (5)$$

Continual Pre-Training Static Embedding We utilize fastText[2] to continual pre-train static embeddings from the code-switched data. fastText is capable of processing at the character n-gram level, allowing it to capture subword information and handle unseen words, which makes it suitable for processing code-switched text. We train the target language monolingual static embeddings $\mathbf{E}$ to obtain cross-lingual static embeddings $\tilde{\mathbf{E}}$ using the code-switched dataset $\mathcal{D}'$.

$$\tilde{\mathbf{E}} = \text{Train}(\mathbf{E}, \mathcal{D}') \quad (6)$$

This step allows the model to better capture the semantics of the domain-specific vocabulary present in the code-switched data, enhancing the quality of the cross-lingual embeddings.

[1]Including singular nouns ('NN'), plural nouns ('NNS'), proper nouns ('NNP'), and plural proper nouns ('NNPS').

[2]We use the pre-trained fastText models: https://fasttext.cc/docs/en/crawl-vectors.html. Each model is trained monolingually for the respective languages.

3.2 Embedding Transfer Using Cross-lingual Static Embedding

Drawing inspiration from FOCUS (Dobler and de Melo, 2023), we describe a robust methodology to migrate expert models to another language utilizing cross-lingual static embeddings. Our approach leverages the inherent structural properties of target language tokenizers and employs a rigorous process to initialize embeddings for target language tokens. The goal is to preserve linguistic information and achieve semantically meaningful embeddings.

First, we denote our source tokenizer as S and the target language tokenizer as T. The tokenizer S has a predefined vocabulary V_S, while the target tokenizer T operates with a vocabulary V_T. For tokens in V_T that overlap with V_S, we adopt a direct transfer strategy where their embeddings are copied from $\mathbf{E}_S$ to $\mathbf{E}_T$. This ensures consistency and preserves the linguistic knowledge embedded in the source embeddings.

For tokens in V_T that do not exist in V_S, a more nuanced approach is required to initialize new token embeddings. Specifically, the process of obtaining an initial embedding for each target token t_i absent from the source vocabulary is as follows:

$$\begin{aligned} \text{Weights} &= \text{Sparsemax}(\text{Similarity}(t_i, V_O)) \\ \mathbf{E}_T(t_i) &= \sum_{t_j \in V_O} \text{Weights}(t_j) \cdot \mathbf{E}_S(t_j) \end{aligned} \quad (7)$$

Here, $V_O = V_S \cap V_T$. $\text{Similarity}(t_i, V_O)$ represents the cosine similarity between the cross-lingual static embedding $\tilde{\mathbf{E}}(t_i)$ of the target token t_i and all embeddings $\tilde{\mathbf{E}}(V_O)$ in the overlapping vocabulary V_O. These cosine similarities are then processed through the SPARSEMAX (Martins and Astudillo, 2016) function, converting the similarity scores into a probability distribution while ensuring sparsity. This highlights the most relevant tokens in V_O. Finally, using the SPARSEMAX-derived weights, we compute the weighted mean of embeddings from V_O to initialize the embedding for t_i.

By leveraging cross-lingual static embeddings, we effectively address the challenges of aligning domain-specific vocabulary across multiple languages. Specialist terms and concepts often lack direct equivalents in other languages, leading to potential loss of nuance and precision. Our approach ensures better alignment by preserving these nuances through datasets trained on domain-specific

data, thus providing a more accurate cross-lingual transfer.

In summary, our methodology provides a systematic approach for transferring domain-specific expert models to the target language by leveraging cross-lingual embedding transfer enhanced through code-switching. This ensures that the migrated models retain their specialized performance and effectively adapt to new linguistic contexts.

4 Experimental Setup

In this section, we describe the details for migrating domain-specific expert models from English to target languages in our experiments. We focus on two domains: science and mathematics. To be specific, we transfer English-trained expert language models to the target languages: Arabic (AR), Bengali (BN), German (DE), Spanish (ES), and Vietnamese (VI).

4.1 Models

Domain Specific Models For the science domain, we utilize the galactica-1.3b model (Taylor et al., 2022). This model is trained on 106 billion tokens of open-access scientific text and data, encompassing a wide range of sources such as papers, textbooks, scientific websites, encyclopedias, reference material, and knowledge bases. For the mathematics domain, we employ the rho-math-7b-v0.1 model (Lin et al., 2024). The Rho-1 base models utilize Selective Language Modeling (SLM) for pretraining, selectively training on clean and relevant tokens that align with the desired distribution. The model is continually pre-trained on a 15 billion token mathematics corpus.

Neural Machine Translation To translate words into each target language, we use the nllb-200-distilled-1.3B (Team et al., 2022). This model is capable of single-sentence translation between 200 languages and is particularly effective for low-resource languages.

Target Language Tokenizer For tokenizing the target languages in our experiments, we employed language-specific tokenizers optimized for each language. Specifically, we used the tokenizer proposed by Cañete et al. (2020) for Spanish, the tokenizer developed by Nguyen Quang Duc (2024) for Vietnamese, and the tokenizers introduced by dbmdz (2021), Zehady (2024), and riotu lab (2024) for German, Bengali, and Arabic, respectively. Detailed descriptions and implementation specifics of each tokenizer are provided in the appendix C.

4.2 Dataset

Train Dataset The data used to train each expert language model is also utilized for continual pre-training of static embeddings. For the science domain experiments, we use the **scientific_papers** (Cohan et al., 2018). This dataset includes long and structured documents from ArXiv and PubMed OpenAccess repositories. For the mathematics domain, we use the **open-web-math** (Paster et al., 2023), which contains high-quality mathematical texts sourced from Common Crawl.

Test Dataset To evaluate the performance of each language and domain-specific model, we adopt the Eleuther AI Language Model Evaluation Harness framework and utilize a multilingual benchmark dataset from Lai et al. (2023). This benchmark dataset includes translations of ARC (Clark et al., 2018), TruthfulQA (Lin et al., 2022), and MMLU (Hendrycks et al., 2021) in 26 languages. The target languages (AR, BN, DE, ES, VI) are included in these translations. Specifically, ARC and TruthfulQA are used to evaluate the scientific domain models, while specific categories within MMLU are used to evaluate the mathematical models. Detailed categories of MMLU used for math model evaluation can be found in Appendix B.

5 Results

In this section, we present the experimental results of our proposed methods across various domains and languages. Our experiments evaluate the performance of our approach in the science and mathematics domains for languages such as Arabic (AR), Bengali (BN), German (DE), Spanish (ES), and Vietnamese (VI). We analyze the effectiveness of our methods including the impact of varying amounts of code-switched data in enhancing cross-lingual transfer and improving model performance. Additionally, we conduct ablation studies to examine the effects of different code-switching strategies and the inclusion of this data during training.

5.1 Performance Evaluation Across Domains by Transfer Methods

Tables 1 and 2 present the performance of our models on the science and mathematics datasets, respectively. The Baseline model, trained primarily on English data, exhibits limited accuracy across target languages due to insufficient cross-lingual generalization. The FOCUS method enhances language-

Methods	ARC						TruthfulQA					
	AR	BN	DE	ES	VI	*Avg*	AR	BN	DE	ES	VI	*Avg*
Baseline	0.2207	0.2352	0.2293	0.2333	0.2282	0.2293	0.2303	0.2087	0.2234	0.2395	0.2293	0.2262
FOCUS	0.2301	0.2558	0.2609	0.2590	0.2239	0.2459	0.2135	0.2215	0.2170	0.2294	0.2038	0.2170
MIGRATE$_{1M}$	0.2267	0.2583	**0.2626**	0.2709	0.2145	0.2466	0.2160	0.2254	**0.2195**	**0.2383**	**0.2089**	**0.2216**
- half	0.2258	0.2583	0.2601	0.2726	0.2179	0.2464	0.2160	0.2254	0.2170	0.2357	0.2000	0.2188
MIGRATE$_{3M}$	0.2284	**0.2660**	0.2618	0.2641	0.2137	0.2468	**0.2173**	0.2254	**0.2195**	0.2319	**0.2089**	0.2206
- half	**0.2344**	0.2592	0.2609	**0.2735**	**0.2282**	**0.2512**	0.2147	**0.2279**	0.2170	0.2345	0.2038	0.2196

Table 1: Performance evaluation of transfer methods across languages on the ARC and TruthfulQA in the science domain. Each value represents the accuracy for the respective language, and **bold** values indicate the best performance. Labeld with "half" denotes that only half of the extracted nouns are translated at random.

Methods	MMLU$_{MATH}$					
	AR	BN	DE	ES	VI	*Avg*
Baseline	0.2288	0.2000	0.2386	0.2464	0.2582	0.2344
FOCUS	0.2395	0.2335	0.2139	0.2617	0.2309	0.2359
MIGRATE$_{1M}$	0.2376	0.2294	0.2167	**0.2703**	**0.2658**	0.2440
- half	0.2395	**0.2406**	0.2148	0.2674	0.2611	**0.2447**
MIGRATE$_{3M}$	**0.2425**	0.2284	**0.2272**	0.2483	0.2498	0.2392
- half	**0.2425**	0.2274	0.2291	0.2464	0.2432	0.2377

Table 2: Performance evaluation of transfer methods across languages on the MMLU in the mathematics domain. Each value represents the accuracy for the respective language, and **bold** values indicate the best performance. Labeld with "half" denotes that only half of the extracted nouns are translated at random.

specific embeddings within the monolingual space, leading to notable improvements over the Baseline. For example, in ARC, Bengali accuracy increases from 0.2352 to 0.2558, and German from 0.2293 to 0.2609. This demonstrates that optimizing monolingual embeddings can positively impact performance even without explicit cross-lingual training. Introducing cross-lingual features through code-switched data, the MIGRATE$_{1M}$ method incorporates 1 million tokens into the static embedding training. This results in further performance enhancements across most languages. In ARC, German achieves an accuracy of 0.2626, the highest among methods utilizing 1 million tokens. The average accuracies in both ARC and TruthfulQA improve compared to FOCUS, indicating that code-switched data enables the model to better capture lexical semantics of target languages, thus enhancing cross-lingual transfer. By increasing the amount of code-switched data to 3 million tokens, the MIGRATE$_{3M}$ method evaluates the impact of larger datasets. Certain languages, such as Arabic and Vietnamese, exhibit significant gains; Vietnamese accuracy in ARC reaches 0.2282. While some languages show marginal improvements, the overall trend suggests that larger volumes of code-switched data contribute to better performance. In the mathematics domain (Table 2), a similar pattern emerges. The Baseline model's limited performance improves slightly with FOCUS, but our methods yield more substantial gains. Notably, Migrate$_{1M}$ achieves the highest average accuracy of 0.2440. These results across both domains confirm that incorporating code-switched data into static embedding training effectively enhances the model's cross-lingual lexical semantic understanding, leading to improved performance in target languages after embedding transfer.

Methods	AR	BN	VI	*avg*
ARC				
FOCUS	0.2524	0.2549	0.2325	0.2466
MIGRATE$_{1M}$	0.2566	0.2596	0.2385	0.2516
MIGRATE$_{3M}$	**0.2618**	**0.2601**	**0.2419**	**0.2546**
TruthfulQA				
FOCUS	0.2250	0.2305	0.2038	0.2198
MIGRATE$_{1M}$	0.2251	0.2330	0.2125	0.2235
MIGRATE$_{3M}$	**0.2354**	**0.2382**	**0.2280**	**0.2339**

Table 3: Performance evaluation of transfer methods followed by Language Adaptation Pre-Training (LAPT) using Wikipedia dataset (Foundation) 1 Billion tokens on FOCUS and half model. Each value represents the accuracy for the respective language (Arabic, Bengali, Vietnamese), and **bold** values indicate the best performance.

5.2 Impact of Language Adaptation Pre-Training

Table 3 presents the performance of our science domain expert models after applying Language Adaptation Pre-Training (LAPT) (Chau et al., 2020) in AR, BN and VI. LAPT serves to further align the initialized embeddings with the model's weights, enhancing the model's capacity for language acquisition and adaptation to the target languages.

In the ARC dataset, FOCUS achieves an average accuracy of 0.2466. With LAPT, the MIGRATE$_{1M}$ method increases this to 0.2516, and the MIGRATE$_{3M}$ method further elevates it to 0.2546. Similarly, in TruthfulQA, the average accuracy improves from 0.2198 with FOCUS to 0.2235 with MIGRATE$_{1M}$ and reaches 0.2339 with MIGRATE$_{3M}$, which represents an approximate 7% improvement over FOCUS.

These results reveal that the substantial performance gains from LAPT are closely tied to the quality of the initialized embeddings. The embeddings transferred with code-switched data provide a strong foundation that can be effectively enhanced by LAPT. This underscores the importance of both effective initialization and the additional adaptation phase provided by LAPT in maximizing the potential of multilingual language models. Thus, the improvements observed after applying LAPT highlight not just its own effectiveness, but also the critical role of proper initialization. A well-initialized model allows LAPT to better align the embeddings with the model's weights, significantly boosting language acquisition capabilities in low-resource languages. For detailed experimental procedures, please refer to Appendix D.

5.3 Ablation Study

To understand the contributions of different components in our approach, we conduct ablation studies focusing on the amount of code-switched words and the presence of code-switching in the training data.

Impact of Translated Noun Quantity We evaluate the impact of the quantity of translated nouns on model performance, as presented in Tables 1 and 2. Specifically, we compare two scenarios: one where all extracted nouns are translated and another where only half are translated at random, denoted as "half" in the tables.

Specifically, in the TruthfulQA, comparing MIGRATE$_{1M}$ with MIGRATE$_{1M}$- half, we observe that the performance is higher when all nouns are translated. For example, in German (DE), the accuracy of all translated nouns is 0.2195, whereas it drops to 0.2170 when using only half of the nouns. Similarly, ES and VI also show better performance when all nouns are translated.

A similar trend is observed in the mathematics domain. Comparing MIGRATE$_{1M}$ with MIGRATE$_{1M}$- half, we observe that the performance is higher when all nouns are translated. For instance, in BN, the MIGRATE$_{1M}$ achieves an accuracy of 0.2447, while it decreases to 0.2440 when only half of the nouns are translated.

Additionally, for the MIGRATE$_{3M}$, translating all nouns generally leads to improved performance across domains, although the improvement in the ARC dataset is less pronounced.

These results suggest that the number of code-switched words plays a crucial role in cross-lingual transfer performance. As the extent of translation increases, the model performance improves, indicating that translating all nouns is more effective. Therefore, translating the full set of extracted nouns is confirmed to be more beneficial for enhancing model performance.

Methods	MMLU$_{MATH}$					
	AR	BN	DE	ES	VI	*Avg*
FOCUS	0.2395	0.2335	0.2139	0.2617	0.2309	0.2359
MIGRATE$_{1M}$	0.2376	0.2294	0.2167	**0.2703**	**0.2658**	**0.2440**
w/o C.S	0.2356	0.2294	0.2196	0.2674	0.2554	0.2415
MIGRATE$_{3M}$	**0.2425**	0.2284	**0.2272**	0.2483	0.2498	0.2392
w/o C.S	0.2393	**0.2325**	0.2158	0.2455	0.2554	0.2377

Table 4: Performance comparison with and without code-switching on the MMLU across different languages in math domain. "w/o C.S" denotes models trained without code-switching. Each value represents the accuracy for the respective language, and **bold** values indicate the highest performance.

Advantages of Code Switching We investigate the role of code-switching in our approach by comparing static embeddings learned from code-switched data with those learned from the original English domain data without code-switching. Specifically, our experiments focus on the mathematics domain. Table 4 presents the performance results for models trained with and without code-switched data in the static embedding phase.

The results show that incorporating code-switched data for static embedding training significantly enhances model performance across most

languages. For the MIGRATE$_{1M}$, the average accuracy decreases from 0.2440 to 0.2415 when only the original English data is used. Similarly, the average accuracy of MIGRATE$_{3M}$ reduces from 0.2392 to 0.2377 without code-switching. Notably, the largest performance drops are observed in AR and VI, indicating that code-switching plays a vital role in improving performance.

Additionally, training with domain-related English data alone provides moderate performance improvements, even without incorporating code-switching. These findings underscore that integrating code-switched data during static embedding training is essential for maximizing the model's multilingual capabilities. While training with domain-related English data provides some benefits over the baseline, the models trained with code-switched data consistently outperform those without it. This highlights the crucial role of code-switching in enhancing cross-lingual lexical alignment and embedding transfer.

6 Conclusion

We address the challenge of migrating English-centric expert models to other languages without the need for domain-specific corpora in those languages. We enhance monolingual static embeddings through further training with code-switching data to create cross-lingual embeddings. These enriched embeddings are then used to initialize the token embeddings of the target languages, facilitating effective embedding transfer.

Furthermore, we highlight the critical importance of proper embedding initialization. By using these enriched cross-lingual embeddings, we establish a strong foundation that, when combined with LAPT, leads to significant performance gains. This underscores the necessity of both effective initialization and the adaptation phase provided by LAPT in maximizing the potential of expert models.

Our experiments across various languages and domains represent simple and effective strategies for enhancing cross-lingual transfer in English-centric expert models. Future work could explore extending this approach to other expert domains and languages.

Limitations

This study presents several limitations. We conduct experiments on a limited set of languages, including Arabic, Bengali, German, Spanish, and Vietnamese. While these languages provide a diverse range of linguistic characteristics for evaluation, there are many other languages with unique scripts and linguistic features that we do not test. Due to constraints in computational resources and data availability, we are unable to experiment with all possible languages. Therefore, we cannot assert that our method will exhibit similar performance improvements for other low-resource languages or those with distinct scripts, and further investigation is needed to substantiate the generalizability of our approach.

Additionally, limitations in GPU resources lead us to employ Parameter-Efficient Fine-Tuning (PEFT) techniques during LAPT phase. While PEFT allows for efficient model adaptation, it may not fully capture all the potential benefits that could be achieved through full fine-tuning. This may restrict the optimal performance of our method, and future work should explore the effects of full fine-tuning using more extensive computational resources to fully realize the advantages of LAPT.

Finally, our experiments are confined to two specific domains: science and mathematics. Although we demonstrate the effectiveness of our approach within these domains, it remains uncertain whether similar benefits would be observed in other domains. Each domain possesses unique linguistic patterns and specialized terminologies that could affect the performance of our method differently. Additional research is necessary to assess the applicability and effectiveness of our approach across a broader range of domains.

Ethical Considerations

In this study, we utilize code-switched data to transfer English-centric expert models to target languages, aiming to enhance access to expert knowledge in fields like science and mathematics for speakers of low-resource languages. Our research makes use of publicly available data, ensuring that all data are used under appropriate licenses.

We recognize that biases present in the original English data may be transferred to the target languages during the model transfer process. Such biases could lead to unequal performance or unintended consequences for certain languages or groups. Additionally, the code-switching process may not fully capture cultural contexts or linguistic nuances of the target languages, potentially resulting in inappropriate expressions or misunderstand-

ings among target language users. It is important for users of our models to exercise caution and consider local cultural and linguistic characteristics when applying them.

Acknowledgments

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (RS-2024-00398115, Research on the reliability and coherence of outcomes produced by Generative AI). This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) under the Leading Generative AI Human Resources Development(IITP-2024-R2408111) grant funded by the Korea government(MSIT). This work was supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. RS-2022-II220369, 2022-0-00369 (Part 4) Development of AI Technology to support Expert Decision-making that can Explain the Reasons/Grounds for Judgment Results based on Expert Knowledge)

References

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2024. Llemma: An open language model for mathematics. *Preprint*, arXiv:2310.10631.

Samuel Cahyawijaya, Holy Lovenia, and Pascale Fung. 2024. Llms are few-shot in-context low-resource language learners. *Preprint*, arXiv:2403.16512.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. Parsing with multilingual BERT, a small corpus, and a small treebank. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. Association for Computational Linguistics.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Nadezhda Chirkova and Vassilina Nikoulina. 2024. Zero-shot cross-lingual transfer in instruction tuning of large language models. *Preprint*, arXiv:2402.14778.

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. *Preprint*, arXiv:1803.05457.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *arXiv e-prints*, pages arXiv–2307.

dbmdz. 2021. dbmdz/german-gpt2.

Konstantin Dobler and Gerard de Melo. 2023. Focus: Effective embedding initialization for monolingual specialization of multilingual models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, page 13440–13454. Association for Computational Linguistics.

Wikimedia Foundation. Wikimedia downloads.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Preprint*, arXiv:2009.03300.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

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

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *Preprint*, arXiv:2307.16039.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. *Preprint*, arXiv:2109.07958.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. 2024. Rho-1: Not all tokens are what you need. *Preprint*, arXiv:2404.07965.

André F. T. Martins and Ramón Fernandez Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. *Preprint*, arXiv:1602.02068.

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

Duong Nguyen, Nam Cao, Son Nguyen, Son Ta, and Cuong Dinh. 2022. Mfinbert: Multilingual pretrained language model for financial domain. In *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–6.

Nguyen Duc Nhan Nguyen Dich Nhat Minh Le Thanh Huong Dinh Viet Sang Nguyen Quang Duc, Le Hai Son. 2024. Towards comprehensive vietnamese retrieval-augmented generation and large language models. *arXiv preprint arXiv:2403.01616*.

OpenAI et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. 2023. Openwebmath: An open dataset of high-quality mathematical web text. *Preprint*, arXiv:2310.06786.

François Remy, Pieter Delobelle, Hayastan Avetisyan, Alfiya Khabibullina, Miryam de Lhoneux, and Thomas Demeester. 2024. Trans-tokenization and cross-lingual vocabulary transfers: Language adaptation of llms for low-resource nlp. *Preprint*, arXiv:2408.04303.

riotu lab. 2024. riotu-lab/arabiangpt-03b.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2024. Code llama: Open foundation models for code. *Preprint*, arXiv:2308.12950.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *Preprint*, arXiv:2211.09085.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Shijie Wu, Benjamin Van Durme, and Mark Dredze. 2022. Zero-shot cross-lingual transfer is underspecified optimization. In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 236–248, Dublin, Ireland. Association for Computational Linguistics.

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

Abdullah Khan Zehady. 2024. Banglallm/bangla-llama-7b-base-v0.1.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, Dongzhan Zhou, Shufei Zhang, Mao Su, Han-Sen Zhong, and Yuqiang Li. 2024. Chemllm: A chemical large language model. *Preprint*, arXiv:2402.06852.

A Static Embedding Training

Pre-trained FastText word vectors for 157 languages are trained on Common Crawl and Wikipedia using CBOW with position-weights. The embeddings are 300-dimensional with character n-grams of length 5, a window size of 5, and 10 negative samples. For our purposes, we continual pre-trained these embeddings on the code-switched data for 1 epoch with a $\alpha = 0.025$. All experiments are performed on a dual-socket Intel Xeon Gold 6230R CPU @ 2.10GHz system with 104 cores and 376GB of RAM.

B Benchmark Dataset in Math

To evaluate the model's performance in the mathematics domain across five languages, we use the mathematics-related subjects from the Multilingual MMLU Dataset (Dac Lai et al., 2023). The specific subjects included are *college_mathematics*, *high_school_mathematics*, *elementary_mathematics*, *abstract_algebra*, and

high_school_statistics. This dataset encompasses a wide range of mathematical concepts and problem-solving skills, from elementary to college-level mathematics, including areas like abstract algebra and statistics. It provides a comprehensive benchmark to assess the model's mathematical understanding and performance in each language.

C Tokenizer Details

In our experiments, we carefully select tokenizers and appropriate vocabulary sizes to effectively capture the linguistic characteristics of each target language, grouping common features to minimize redundancy.

For both Arabic (riotu lab, 2024) and German (dbmdz, 2021), Byte Pair Encoding (BPE) tokenizers tailored to these languages are used. The Arabic tokenizer is modified to accommodate the rich morphology of the Arabic script, resulting in a vocabulary size of 64,002 tokens. This adjustment ensures better tokenization of inflected forms and improves overall language representation. Similarly, a German tokenizer trained on a German corpus is used, employing byte-level BPE with a vocabulary size of 50,265 tokens. This tokenizer effectively captures German orthography and compound word formation, which are characteristic of the language.

For Bengali (Zehady, 2024) and Vietnamese (Nguyen Quang Duc, 2024), SentencePiece (Kudo and Richardson, 2018) is adopted, which is effective for languages with complex scripts and lack of clear word boundaries. The Bengali tokenizer enhances the base vocabulary by adding 16,000 Bengali-specific tokens, increasing the total vocabulary size to 50,437 tokens. This expansion improves the representation of Bengali script and phonetics, allowing for more accurate tokenization of native words. In the case of Vietnamese, a SentencePiece without prior word segmentation is used to create 20,000 Vietnamese tokens. By merging this with the original vocabulary and removing duplicate tokens, a total vocabulary size of 46,303 tokens is achieved. This significantly improved tokenization efficiency for Vietnamese text, reducing the number of tokens required compared to previous versions.

For Spanish (Cañete et al., 2020), a SentencePiece employing BPE subwords, trained on a large Spanish corpus, is utilized. The vocabulary consists of approximately 31,002 tokens, effectively handling Spanish morphology and syntax, including accented characters and conjugations, which are prevalent in the language.

D Experimental Setting in LAPT

After completing the embedding transfer, we perform Language Adaptation Pre-Training (LAPT) on the models to further adapt them to the target languages. Given computational resource constraints, we employ Parameter-Efficient Fine-Tuning (PEFT) techniques, specifically using Low-Rank Adaptation of Large Language Models(LoRA) (Hu et al., 2021). This approach allows us to efficiently fine-tune the models without updating all of the parameters, thereby reducing computational demands while maintaining model performance. For training data, we utilize the Wikipedia dataset for each target language, which provides a substantial amount of monolingual text suitable for language adaptation[3].

LoRA Configuration We configure LoRA with a rank (r) of 64 for the low-rank adaptation matrices and set the scaling factor (α) to 16. A dropout probability of 0.1 is applied to the LoRA layers to prevent overfitting. No bias terms are included in the LoRA layers, simplifying the adaptation process. LoRA is applied to the transformer modules associated with the attention mechanisms, specifically the query, key, value, and output projections.

Hyperparameters We set the batch size to 32. The models are trained for 1 epoch using the AdamW optimizer with a learning rate of 1e-4. The maximum sequence length is set to 2048 tokens, and we apply a warmup ratio of 8% using the WarmupDecayLR scheduler. Mixed-precision training using the bfloat16 format is enabled to reduce memory usage and accelerate computation.

Hardware We use 4 NVIDIA A6000 GPUs, each with 48GB of memory capacity, along with AMD EPYC 7513 processors featuring 32 cores, to train the LLMs. For inference, we employ a single accelerator.

[3] https://huggingface.co/datasets/wikimedia/wikipedia