# SLAM: Towards Efficient Multilingual Reasoning via Selective Language Alignment

**Yuchun Fan[1], Yongyu Mu[1], Yilin Wang[1], Lei Huang[3], Junhao Ruan[1], Bei Li[4], Tong Xiao[1,2*], Shujian Huang[5], Xiaocheng Feng[3], and Jingbo Zhu[1,2]**
[1]NLP Lab, School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2]NiuTrans Research, Shenyang, China
[3]Harbin Institute of Technology, Harbin, China [4]Meituan Inc.
[5]National Key Laboratory for Novel Software Technology, Nanjing University
yuchunfan_neu@outlook.com {xiaotong,zhujingbo}@mail.neu.edu.cn

## Abstract

Despite the significant improvements achieved by large language models (LLMs) in English reasoning tasks, these models continue to struggle with multilingual reasoning. Recent studies leverage a full-parameter and two-stage training paradigm to teach models to first understand non-English questions and then reason. However, this method suffers from both substantial computational resource computing and catastrophic forgetting. The fundamental cause is that, with the primary goal of enhancing multilingual comprehension, an excessive number of irrelevant layers and parameters are tuned during the first stage. Given our findings that the representation learning of languages is merely conducted in lower-level layers, we propose an efficient multilingual reasoning alignment approach that precisely identifies and fine-tunes the layers responsible for handling multilingualism. Experimental results show that our method, SLAM, only tunes 6 layers' feed-forward sub-layers including $6.5-8\%$ of all parameters within 7B and 13B LLMs, achieving superior average performance than all strong baselines across 10 languages. Meanwhile, SLAM only involves one training stage, reducing training time by $4.1-11.9\times$ compared to the two-stage method[1].

## 1 Introduction

Large language models (LLMs) (Touvron et al., 2023; OpenAI, 2023) have demonstrated significant advancements in reasoning abilities (Huang and Chang, 2023). However, these improvements are primarily focused on English, leading to inferior performance in non-English scenarios, especially in low-resource languages (Chen et al., 2023b). As the demand for deploying LLMs in multilingual environments increases (Qin et al., 2024), recent years have witnessed a growing interest in multilingual reasoning alignment, which aims to bridge the gap between the non-English reasoning abilities of LLMs and those in English (Chen et al., 2023b; Zhu et al., 2024; She et al., 2024).

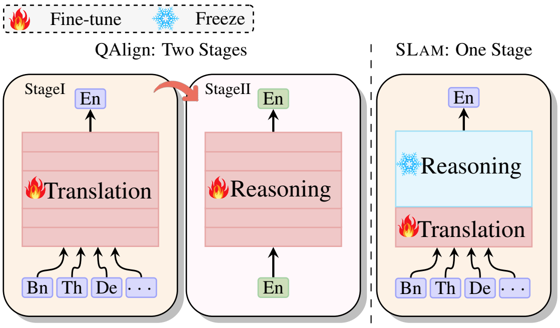


Figure 1: The comparison of QAlign with SLAM (Ours). 🟪 and 🟩 denote the translation data and reasoning data, respectively. Unlike the traditional two-stage training approach that fully trains all parameters, SLAM selectively trains only the lower-level layers responsible for multilingual comprehension in one stage.

Early work (Chen et al., 2023b) along this line of research directly fine-tunes models on multilingual mathematical question-answer pairs synthesized via machine translation. However, the reasoning answers are too complex for accurate translation, potentially compromising the performance of fine-tuned models. Building upon this, Zhu et al. (2024) introduce a two-stage learning strategy. It first teaches models to comprehend multilingual inputs by translating non-English questions into English counterparts and then trains them with English-only reasoning datasets to awaken their multilingual reasoning abilities.

Despite its effectiveness, notable limitations persist with the two-stage framework. On the one hand, during its training process, all parameters of LLMs are tuned to facilitate multilingual reasoning alignment, which consumes substantial computa-

[*] Corresponding author.
[1]The project will be available at: https://github.com/fmm170/SLAM

*Proceedings of the 31st International Conference on Computational Linguistics*, pages 9499–9515
January 19–24, 2025. ©2025 Association for Computational Linguistics

tional resources and hinders applying this method in resource-constrained scenarios. Moreover, conducting full-parameter training in the first training stage can lead to catastrophic forgetting, destroying the inherent reasoning abilities within LLMs. Thus this method depends on the second training stage to awaken the model's reasoning abilities, seeming necessary but inefficient.

When considering multilingual models, one might think of capturing language-specific knowledge in certain parts of these models. Tang et al. (2024) has verified that only a small number of parameters in LLMs are language-dependent. In this work, we further confirm this viewpoint by examining neuron activations across different layers of LLMs. We find that lower-level layers are more involved in learning language-specific representations, which are then transformed into universal representations in higher-level layers. This suggests that it might be worth separating the learning of language-specific and universal representations in developing multilingual reasoning LLMs.

This work is motivated by a perspective of parameter-efficient fine-tuning. In response to our findings, we precisely identify and fine-tune the *multilingualism-handling layers* with translation-only data to achieve multilingual reasoning alignment in one stage. Our method, SLAM, first calculates the mean squared deviation (MSD) of the numbers of neurons activated by different languages and selects the layers with higher MSD scores. Next, recognizing that feed-forward networks (FFN) store most of the multilingual knowledge (Geva et al., 2021), SLAM achieves further efficiency by only training FFN sub-layers within the selected layers. Compared to the two-stage method, SLAM significantly improves the efficiency in terms of training data and computational resources. Moreover, SLAM merely trains models one time since freezing irrelevant parameters effectively prevents the inherent reasoning abilities from being destroyed, as illustrated by Figure 1.

Experimental results on two multilingual mathematical reasoning benchmarks, MGSM (Shi et al., 2023) and MSVAMP (Chen et al., 2023b), show that SLAM outperforms strong baselines in both in-domain and out-of-domain settings, with only 8% and 6.5% of the parameters tuned in 7B and 13B models, respectively. Furthermore, SLAM reduces the training time by $4.1\times$ and $11.9\times$ compared to the two-stage method. Moreover, SLAM can also be generalized to multilingual common sense reasoning (Lin et al., 2021) and can leverage models with advanced reasoning abilities to consistently enhance multilingual reasoning performance.

## 2 Background

**Point-wise feed-forward network.** Multilingual reasoning requires LLMs to integrate both multilingual comprehension and reasoning abilities. Our approach builds on the findings that factual knowledge is stored in the model's FFN sub-layers (Dai et al., 2022; Meng et al., 2022). By directly injecting language-specific knowledge into these FFN sub-layers, we can enhance the multilingual comprehension abilities of LLMs. Typically, each layer of Transformer-based LLMs is predominantly composed of a multi-head self-attention (MHA) sub-layer followed by an FFN sub-layer. Concretely, let $l$ denote the length of the input sentence. Given the output $\boldsymbol{x}^i \in \mathbb{R}^{l\times d_{\text{model}}}$ from the MHA sub-layer at the $i$-th layer, the computation within the FFN sub-layer can be formulated as follows:

$$\text{FFN}(\boldsymbol{x}^i) = f(\boldsymbol{x}^i \cdot \boldsymbol{W}^i_{\text{up}}) \cdot \boldsymbol{W}^i_{\text{down}}, \qquad (1)$$

where $\boldsymbol{W}^i_{\text{up}} \in \mathbb{R}^{d_{\text{model}}\times d_{\text{inter}}}$, and $\boldsymbol{W}^i_{\text{down}} \in \mathbb{R}^{d_{\text{inter}}\times d_{\text{model}}}$ are the mapping matrices. The $d_{\text{model}}$ denotes the input dimension, $d_{\text{inter}}$ refers to the intermediate hidden dimension, and $f(\cdot)$ represents a non-linear activation function.

**Neuron activation.** Recent work (Dai et al., 2022; Mu et al., 2024) reveals that language-specific neurons in the FFN sub-layers significantly influence how LLMs process multilingual languages. As a new variant of the activation function, SwiGLU (Shazeer, 2020) is widely used in current LLMs (Touvron et al., 2023). Thus, Equation (1) can further be decomposed as follows:

$$\text{FFN}(\boldsymbol{x}^i) = \left[f(\boldsymbol{W}^i_{\text{gate}}(\boldsymbol{x}^i)) \otimes \boldsymbol{W}^i_{\text{up}}(\boldsymbol{x}^i)\right] \cdot \boldsymbol{W}^i_{\text{down}}, \qquad (2)$$

where $\boldsymbol{W}^i_{\text{gate}} \in \mathbb{R}^{d_{\text{model}}\times d_{\text{inter}}}$ is the mapping metric introduced by SwiGLU. A neuron is defined as a single column in $\boldsymbol{W}^i_{\text{up}}$, thereby an FFN sub-layer within one layer consists of $d_{\text{inter}}$ neurons. In our work, a neuron in the $i$-th FFN sub-layer is considered activated if the value of the element in $f(\boldsymbol{W}^i_{\text{gate}}(\boldsymbol{x}^i))$ exceeds zero (Tang et al., 2024).

## 3 Methodology

In this section, we start with a preliminary study of examining neuron activations across different

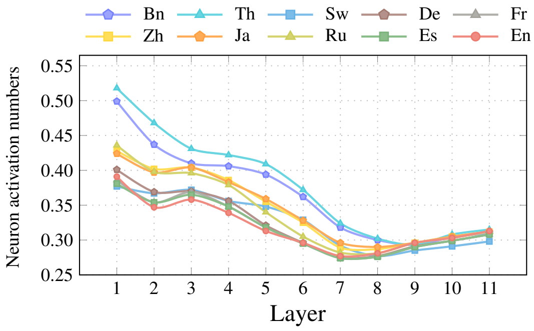


Figure 2: The number of activated neurons across all languages. The count of activated neurons is normalized by the total neurons within the FFN sub-layer.

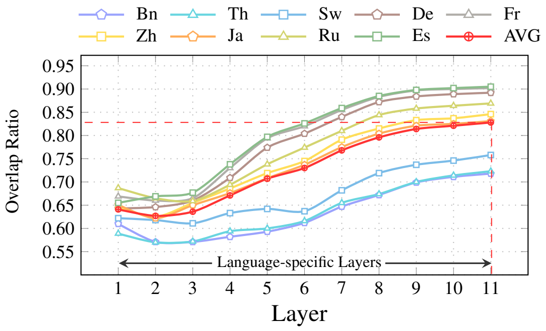


Figure 3: The overlap between non-English and English activated neurons. "AVG" denotes the average overlap ratio among all non-English languages.

layers during the multilingual reasoning process, which motivates our approach to achieving efficient multilingual reasoning alignment.

### 3.1 Preliminary Study

We examine neuron activations across different layers from two perspectives: (1) the number of neurons activated by different languages, and (2) the overlap ratio of activated neurons between non-English languages and English. We first sample $n$ questions, each expressed in 10 different languages. Let $lang$ denote the language of the input sentence. Subsequently, we calculate the count of activated neurons for all samples of a language $lang$ in the $i$-th layer, as specified by the following equation:

$$A^i_{lang} = \mathbb{I}[f(\boldsymbol{W}^i_{\text{gate}}(x^i)) > 0], \qquad (3)$$

where $\mathbb{I}$ is the indicator function. To provide a more intuitive understanding of neuronal activation, we normalize the count of activated neurons, as defined by the following equation:

$$R^i_{lang} = \frac{A^i_{lang}}{d_{\text{inter}}}. \qquad (4)$$

Then we compute the overlap ratio of activated neurons between non-English languages and English.

The results presented in Figures 2, 3 show that while the number of activated neurons across all languages initially decreases and then stabilizes, the neurons activated by non-English languages and English progressively overlap and finally reach a stable level. This indicates the existence of language-specific layers, which are more involved in learning language-specific representations. The findings also align with the phenomenon found in Zhao et al. (2024). Specifically, we define all layers preceding the point at which the average overlap ratio among all non-languages reaches its maximum as language-specific layers, as shown in Figure 3. Full visualization results of all layers are presented in Figure 11 and Figure 12, respectively.

### 3.2 SLAM

In response to our findings, we formally introduce a training method to efficiently achieve multilingual reasoning alignment of LLMs in one stage. The method consists of two steps: (1) Selecting multilingualism-handling layers, and (2) Selectively supervised fine-tuning.

**Selecting multilingualism-handling layers.** Directly fine-tuning the language-specific layers will inevitably impair the model's inherent reasoning abilities, since reasoning abilities also persist in these layers (Chen et al., 2023a). Therefore, we design a layer selection algorithm to identify layers that are more involved in multilingual comprehension, thereby effectively balancing understanding and reasoning abilities. Consequently, we introduce the mean squared deviation (MSD) to precisely measure the stabilization of neuron activation across different languages. For each layer $i$ within the language-specific layers denoted by $\mathcal{K}$, $MSD^i$ is defined by the following equations:

$$\mu^i = \frac{1}{|L|} \sum_{lang \in L} R^i_{lang}, \qquad (5)$$

$$MSD^i = \frac{1}{|L|} \sum_{lang \in L} (R^i_{lang} - \mu^i)^2, \qquad (6)$$

where $L$ denotes all languages and $R^i_{lang}$ is calculated using Equation (4). A higher $MSD^i$ indicates the destabilization of neurons activated in different languages, suggesting that the layer is more actively engaged in multilingual comprehension. The average engagement in multilingual comprehension across language-specific layers is quantified

| Task | Dataset | Usage | Lang | Number |
|---|---|---|---|---|
| **Multilingual Mathematical Reasoning** | MGSM-I (Question) | Training | 10 | 57,817 |
| | MGSM-I (Answer) | Training | 10 | 65,968 |
| | MGSM | In-Domain Test | 10 | 2,500 |
| | MSVAMP | Out-of-Domain Test | 10 | 10,000 |
| **Common Sense Reasoning** | XCSQA-TEST | Training | 16 | 16,110 |
| | Flores-200-DEV | Training | 16 | 14,955 |
| | XCSQA-DEV | Test | 16 | 16,000 |

Table 1: Statistics of the involved datasets. "Lang" denotes the number of languages covered, and "Number" denotes the total samples within each dataset. "MGSM-I" stands for the MGSM8KINSTRUCT dataset.

by the following equation:

$$\theta = \frac{1}{|\mathcal{K}|} \sum_{i \in \mathcal{K}} MSD^i. \tag{7}$$

We select the layers with $MSD^i$ exceeding threshold $\theta$ for subsequent fine-tuning, as these layers contribute most to multilingual comprehension while minimally affecting reasoning abilities.

**Selectively supervised fine-tuning.** Since the FFN sub-layers store the majority of the knowledge (Geva et al., 2021; Dai et al., 2022), we achieve further efficiency by only training the FFN sub-layers within the multilingualism-handling layers utilizing X-English translation data. Given non-English inputs $I_{lang}$ and their English counterparts $I_{eng}$, the optimization objective can be formulated as:

$$\arg\min_{\varphi} \sum_{lang \in L \setminus \{\text{English}\}} -\log p_{\varphi}(I_{eng}|I_{lang}), \tag{8}$$

where $\varphi$ denotes the FFN of the selected layers.

# 4 Experiment Settings

## 4.1 Baseline Models

For a fair comparison, we compare SLAM with the following strong baselines, all trained based on Llama 2 (Touvron et al., 2023). Detailed statistics of the training data for all baselines can be found in the Appendix A.

**MAmmoTH:** Yue et al. (2024) collected diverse instruction datasets on MATH and directly fine-tuned the model on these datasets.

**WizardMath:** Luo et al. (2023) leveraged reinforcement learning to train the model on two mathematical reasoning datasets GSM8K and MATH.

**MetaMath:** Yu et al. (2023) first employed question bootstrapping to create high-quality English reasoning dataset METAMATHQA and then fine-tuned the model on the dataset.

**MathOctopus:** Chen et al. (2023b) employed supervised fine-tuning using MSGM8KINSTRUCT, a multilingual reasoning dataset constructed by directly translating the data from GSM8K.

**LangBridge:** Yoon et al. (2024) introduced an extra multilingual encoder and trained a linear layer connecting this encoder to the LLM using English data to enhance multilingual comprehension.

**QAlign:** Zhu et al. (2024) employed a two-stage training strategy, where the model first learns to translate non-English questions into English and then unlocks multilingual reasoning abilities using the English reasoning data METAMATHQA.

## 4.2 Experimental Details

We constructed X-English translation data from MGSM8KINSTRUCT (Chen et al., 2023b) as training data. For MGSM (Shi et al., 2023) and MSVAMP (Chen et al., 2023b), we trained only the FFN within the first six layers of the model. Considering that QAlign and LangBridge are either trained on METAMATHQA or built upon MetaMath, we implemented SLAM on the MetaMath model to ensure a fair comparison. During inference, we adopted the settings from Yu et al. (2023). For more details, please refer to Appendix A.

## 4.3 Evaluation Datasets

We utilized MGSM and MSVAMP as in-domain and out-of-domain test sets to evaluate the multilingual mathematical reasoning abilities of LLMs. Data statistics are reported in Table 1.

## 4.4 Evaluation Metrics

Following Zhu et al. (2024), our evaluation primarily focuses on two key dimensions: Accuracy and Prediction Consistency Ratio.

**Accuracy.** Following Yu et al. (2023), accuracy is calculated by comparing the last numerical value in the response to the gold answer. Higher accuracy indicates stronger reasoning ability.

**Prediction Consistency Ratio (PCR).** Assuming $M$ and $N$ are sets of questions correctly answered in English and non-English languages respectively. PCR is calculated as: $\frac{|M \cap N|}{|M|}$. Higher PCR denotes the model's non-English reasoning ability is closer to its English reasoning ability.

| Model | Training Cost | Trained Param. | Bn | Th | Sw | Ja | Zh | De | Fr | Ru | Es | En | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | *7B Models* | | | | | | |
| MAmmoTH[†] | 2.7× | 100.0% | 3.6 | 4.8 | 2.4 | 10.8 | 17.2 | 33.2 | 32.8 | 26.0 | 32.4 | 49.6 | 21.3 |
| WizardMath[†] | 0.9× | 100.0% | 2.0 | 4.0 | 3.4 | 24.0 | 22.4 | 30.4 | 30.4 | 30.8 | 34.8 | 47.6 | 23.0 |
| MetaMath[†] | 3.5× | 100.0% | 7.6 | 5.6 | 5.2 | 34.0 | 45.2 | 54.0 | **56.8** | 51.6 | 58.8 | 65.5 | 38.4 |
| MathOctopus[†] | 0.8× | 100.0% | 28.8 | 34.4 | 39.2 | 36.0 | 38.4 | 44.8 | 43.6 | 39.6 | 42.4 | 52.4 | 40.0 |
| QAlign[†] | 4.1× | 100.0% | **32.4** | 39.6 | **40.4** | 44.0 | **48.4** | **54.8** | **56.8** | 52.4 | **59.6** | **68.0** | **49.6** |
| +LoRA (r=256) | 2.6× | 8.7% | 6.0 | 6.8 | 6.0 | 22.0 | 22.0 | 28.0 | 32.8 | 30.4 | 29.6 | 38.8 | 22.2 |
| **SLAM** | 1.0× | 8.0% | 32.0 | **44.0** | 40.0 | **46.0** | **48.4** | 54.0 | 55.2 | **54.8** | 56.8 | 64.8 | **49.6** |
| | | | | | | | *13B Models* | | | | | | |
| MAmmoTH[†] | 7.3× | 100.0% | 3.6 | 5.2 | 1.6 | 19.2 | 31.2 | 45.6 | 39.6 | 36.8 | 50.0 | 56.4 | 28.9 |
| WizardMath[†] | 2.5× | 100.0% | 6.4 | 5.6 | 5.6 | 22.0 | 28.0 | 40.4 | 42.0 | 34.4 | 45.6 | 52.8 | 28.3 |
| MetaMath[†] | 10.0× | 100.0% | 12.4 | 11.2 | 6.4 | 42.0 | 46.0 | **64.0** | 62.4 | 61.6 | 64.8 | 68.4 | 43.9 |
| MathOctopus[†] | 2.0× | 100.0% | 35.2 | 46.8 | 42.8 | 43.2 | 48.8 | 44.4 | 48.4 | 47.6 | 48.0 | 53.2 | 45.8 |
| QAlign[†] | 11.9× | 100.0% | 38.4 | **49.6** | 46.0 | 52.4 | **59.2** | 62.0 | 62.4 | **64.4** | 67.2 | 69.2 | 57.1 |
| +LoRA (r=256) | 3.1× | 7.1% | 12.8 | 15.2 | 10.8 | 35.6 | 34.4 | 46.0 | 44.4 | 40.8 | 47.2 | 54.8 | 34.2 |
| **SLAM** | 1.0× | 6.5% | **45.6** | 47.6 | **46.4** | **54.0** | 58.8 | 62.8 | **65.2** | **64.4** | **67.6** | **71.2** | **58.3** |

Table 2: The accuracy (%) on the in-domain MGSM test sets. "Avg." denotes the average multilingual performance and the highest score among systems of the same size are highlighted in **bold**. "Training Cost" denotes the time required for training models. "Trained Param." indicates the proportion of trainable parameters to the model's total parameters. Results marked with † come from Zhu et al. (2024).

## 5 Experimental Results

### 5.1 Main Results

**SLAM effectively achieves multilingual reasoning alignment.** We present the results on MGSM in Table 2, which demonstrates that SLAM outperforms all strong baselines in in-domain settings. Specifically, SLAM achieves comparable performance with QAlign in 7B models and surpasses all baselines in 13B models, with an average accuracy improvement of 2.1%. Notably, compared to MetaMath, SLAM exhibits substantial improvements, achieving increases of 29.2% and 32.8% in the 7B and 13B models, by selectively fine-tuning multilingualism-handling layers in MetaMath using translation data. Furthermore, as shown in Figure 7 (a), SLAM demonstrates higher answer consistency. This highlights that directly enhancing multilingual comprehension at specific lower-level layers can effectively improve the multilingual reasoning abilities of LLMs.

| Models | Extra Param. | Training Cost | MGSM | MSVAMP |
|---|---|---|---|---|
| MetaMath-7B | - | 10.0× | 38.4 | 46.2 |
| **+LB[†]-9B** | 2B | 0.3× | 48.8 | 52.0 |
| **+SLAM** | 0B | 1.0× | **49.6** | **60.5** |
| MetaMath-13B | - | 10.0× | 43.9 | 51.8 |
| **+LB[†]-20B** | 7B | 0.3× | 55.8 | 57.9 |
| **+SLAM** | 0B | 1.0× | **58.3** | **62.7** |

Table 3: The average accuracy of LangBridge 9B/20B models on the MGSM and MSVAMP test sets. Results marked with † come from Yoon et al. (2024).

**SLAM shows significant out-of-domain generalization.** To further validate the generalization ability of SLAM, we evaluate it on the out-of-domain test sets, MSVAMP. As shown in Table 4, SLAM demonstrates robust generalization compared with all baselines, exhibiting an average accuracy improvement of 5.8% and 0.2% in the 7B and 13B models, respectively. Notably, SLAM-7B shows substantial performance gains across all 10 languages. Moreover, as shown in Figure 7 (b), SLAM also enhances answer consistency across all non-English languages. This superior generalization is attributed to the modest number of trainable parameters in SLAM. Unlike the baselines, which undergo fine-tuning all parameters on the in-domain dataset and potentially suffer from overfitting, SLAM only fine-tunes partial parameters, significantly reducing the impact of domain shifts.

**SLAM demonstrates superior efficiency.** Rather than fine-tuning all layers, SLAM selectively trains layers responsible for multilingual comprehension. As shown in Table 2 and Table 4, SLAM fine-tunes only 8% and 6.5% of the

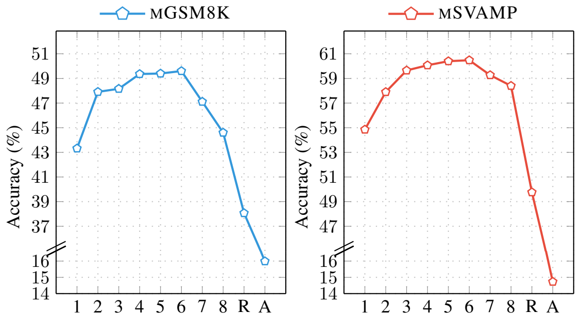
| Model | Training Cost | Trained Param. | Bn | Th | Sw | Ja | Zh | De | Fr | Ru | Es | En | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | *7B Models* | | | | | | | |
| MAmmoTH† | 2.7× | 100.0% | 4.3 | 6.3 | 4.2 | 26.7 | 26.8 | 39.6 | 39.9 | 33.7 | 42.9 | 45.1 | 26.3 |
| WizardMath† | 0.9× | 100.0% | 16.1 | 17.0 | 10.3 | 37.9 | 36.3 | 39.2 | 37.7 | 37.4 | 44.8 | 48.5 | 32.5 |
| MetaMath† | 3.5× | 100.0% | 15.0 | 17.1 | 15.4 | 51.9 | 54.4 | 60.9 | 62.2 | 59.3 | 63.3 | **65.5** | 46.2 |
| MathOctopus† | 0.8× | 100.0% | 31.8 | 39.3 | 43.4 | 41.1 | 42.6 | 48.4 | 50.6 | 46.9 | 49.4 | 50.7 | 44.1 |
| QAlign† | 4.1× | 100.0% | 41.7 | 47.7 | 54.8 | 58.0 | 55.7 | 62.8 | 63.2 | 61.1 | 63.3 | 65.3 | 57.2 |
| +LoRA (r=256) | 2.6× | 8.7% | 19.3 | 20.1 | 15.1 | 33.0 | 32.7 | 46.2 | 47.3 | 41.9 | 47.2 | 51.4 | 35.4 |
| **SLAM** | 1.0× | 8.0% | **49.1** | **50.6** | **55.4** | **60.6** | **63.1** | **64.6** | **65.4** | **64.1** | **66.6** | 65.3 | **60.5** |
| | | | | | | *13B Models* | | | | | | | |
| MAmmoTH† | 7.3× | 100.0% | 5.0 | 13.7 | 12.9 | 42.2 | 47.7 | 52.3 | 53.8 | 50.7 | 53.9 | 53.4 | 38.6 |
| WizardMath† | 2.5× | 100.0% | 13.7 | 16.3 | 12.5 | 29.5 | 37.0 | 48.7 | 49.4 | 43.8 | 49.4 | 56.3 | 35.7 |
| MetaMath† | 10.0× | 100.0% | 20.6 | 20.5 | 19.1 | 57.0 | 58.8 | 68.4 | 68.1 | **67.5** | 68.9 | 68.9 | 51.8 |
| MathOctopus† | 2.0× | 100.0% | 35.2 | 41.2 | 46.8 | 39.2 | 52.0 | 47.2 | 48.0 | 45.6 | 53.2 | 56.4 | 46.5 |
| QAlign† | 11.9× | 100.0% | 49.2 | **55.5** | 55.2 | **64.3** | **63.8** | **69.5** | 68.1 | 66.4 | **66.4** | 67.6 | 62.6 |
| +LoRA (r=256) | 3.1× | 7.1% | 30.2 | 34.8 | 24.2 | 49.3 | 54.6 | 59.7 | 60.4 | 57.0 | 59.5 | 63.0 | 49.3 |
| **SLAM** | 1.0× | 6.5% | **52.5** | 53.5 | **58.2** | 62.5 | 61.7 | 68.8 | **69.9** | 64.5 | 66.1 | **69.5** | **62.7** |

Table 4: The accuracy (%) on the out-of-domain MSVAMP test sets.

parameters in the 7B and 13B models, respectively. Notably, compared with QAlign, SLAM reduces training time by 4.1 × and 11.9 × in the 7B and 13B models. These results suggest that SLAM not only achieves effective performance in multilingual reasoning but also exhibits superior efficiency. Additionally, we also compare SLAM with LangBridge (Yoon et al., 2024), which only fine-tunes the linear layer that aligns the multilingual encoder to the LLMs. As indicated in Table 3, despite LangBridge having slightly less training time, it underperforms in both in-domain and out-of-domain multilingual reasoning performance compared with SLAM. Furthermore, the additional multilingual encoder in LangBridge increases its parameters, thereby reducing its efficiency during inference and increasing deployment costs.

### 5.2 Ablation Study

**Training different layers.** To evaluate the necessity of the layer selection training strategy, we conduct ablation studies by training different layers. As shown in Figure 4, the layers selected by SLAM achieve the best performance both in-domain and out-of-domain performance. Selecting an insufficient number of layers may lead to inadequate multilingual comprehension, while excessive layer selection will impair the model's reasoning abilities. This suggests that while the ability to handle multilingualism is concentrated in lower-level layers, precisely selecting layers that are more actively engaged in multilingual comprehension is crucial for effective multilingual reasoning alignment.



(a) End Training Layer (b) End Training Layer

Figure 4: The average accuracy of training different layers. The $x$-axis, "End Training Layer" signifies that model training encompasses all FFN sub-layers from the first through to the specified layer. "R" denotes randomly selected layers, and "A" denotes all layers.

**Training different sub-layers.** To explore the role of different sub-layers within the multilingualism-handling layers, we conduct ablation studies by training only the Attention sub-layers, and both the Attention and FFN sub-layers, utilizing X-English translation data. As shown in Figure 5, training only the FFN sub-layers results in the highest average accuracy for both in-domain and out-of-domain tests (For the results of MSVAMP, refer to Appendix B). Specifically, training only the FFN sub-layers leads to notable

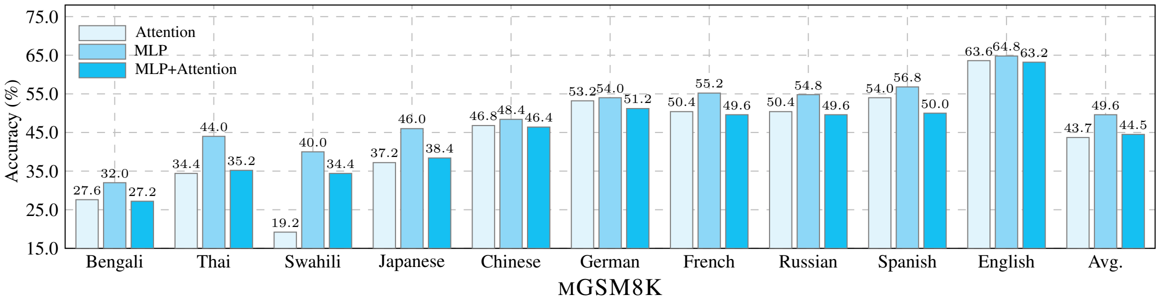


Figure 5: The accuracy of training different sub-layers on MGSM8K test sets.

| Model | Sw | Hi | Ar | Zh | De | En | Avg. |
|---|---|---|---|---|---|---|---|
| Llama2-mono | 22.1 | 31.9 | 31.7 | 52.3 | 58.4 | 77.6 | 45.6 |
| +SLAM | 27.1 | 36.1 | 36.3 | 54.6 | 59.9 | 77.7 | 48.6 |
| Improvements | **+5.0** | **+4.2** | **+4.6** | **+2.3** | **+1.5** | **+0.1** | **+3.0** |

Table 5: The accuracy (%) on the XCSQA test sets.

| Model | Training Cost | Trained Param. | MGSM8K |  | MSVAMP |  |
|---|---|---|---|---|---|---|
|  |  |  | Non-En | En | Non-En | En |
| RFT-7B | 0.8× | 100.0% | 33.6 | **67.6** | 38.0 | **59.6** |
| +SLAM | 1.0× | 6.7% | **44.8** | 65.6 | **50.7** | 59.5 |
| RFT-13B | 2.6× | 100.0% | 38.0 | **75.2** | 45.5 | **66.3** |
| +SLAM | 1.0× | 5.4% | **54.5** | 72.0 | **56.0** | 65.5 |

Table 6: The accuracy (%) of RFT-MuggleMath 7B and 13B models on the MGSM and MSVAMP test sets. For accuracy of all languages, refer to Appendix D.

improvements in low-resource languages, such as Bengali, Thai, and Swahili. This indicates that models can utilize the multilingual knowledge within the FFN sub-layers to enhance the comprehension of multilingual questions.

## 6 Analysis

### 6.1 Scalability of SLAM in Multilingual Common Sense Reasoning

To evaluate the scalability of SLAM, we extend our method to Multilingual Common Sense Reasoning (XCSQA) (Lin et al., 2021). More experimental details and results can be found in Appendix C. We observe that SLAM improves accuracy across all languages. Notably, as shown in Table 5, SLAM achieves significant improvements for low-resource languages such as Swahili, Hindi, and Arabic, with increases of 22.6%, 13.2%, and 14.5%, respectively. The results demonstrate that our method can be effectively adapted to other multilingual reasoning tasks. This also suggests that the multilingual reasoning process can be decomposed into comprehend-then-reason patterns across layers.

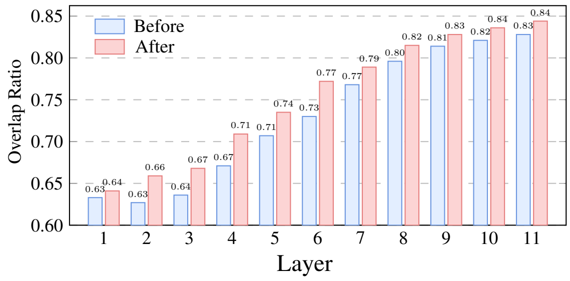


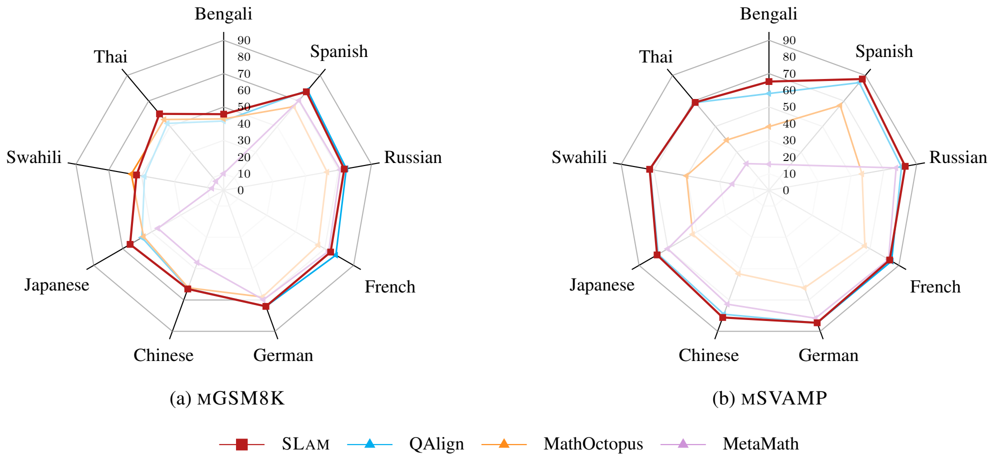
Figure 6: The comparison of the overlap ratio between non-English languages and English activated neurons in the MetaMath-7B model, before and after training.

### 6.2 Extending SLAM to Other Strong Reasoning Models.

We extend our method to the RFT-MuggleMath models (Li et al., 2024), which possess stronger reasoning abilities. As shown in Table 6, SLAM achieves significant in-domain and out-of-domain improvements, with increases of 33.3% in the average accuracy for non-English languages in both the 7B and 13B models, while tuning only 6.7% and 5.4% of parameters, respectively. This suggests that when the models possess stronger English reasoning abilities, it provides an advantageous starting point for SLAM, which leads to significant improvements in multilingual reasoning alignment.

### 6.3 Comparison of Neuron Activation Before and After Training

To facilitate comparisons, we compute the average overlap ratio across all non-English languages to represent the overlap at that layer. As shown in Figure 6, the overlap ratio increases significantly after training (The comparison results of MetaMath-13B, refer to Appendix E). This demonstrates that enhancing the overlap of activated neurons between non-English languages and English at lower-level layers can improve the model's comprehension of multilingual questions, thereby achieving better



(a) MGSM8K

(b) MSVAMP

Figure 7: PCR results across various systems on the MGSM8K and MSVAMP test sets.

multilingual reasoning alignment. We also visualize the representations of the final token of multilingual questions, as it is crucial for the model's subsequent reasoning (Wendler et al., 2024). As shown in Figure 10, the semantic space becomes more unified after training, thereby facilitating the sharing of abilities across languages.

## 7 Related Work

### 7.1 Multilingual Mathematical Reasoning

Significant performance discrepancies in LLM reasoning between high-resource and low-resource languages have spurred research aimed at aligning their multilingual reasoning abilities. Early efforts (Chen et al., 2023b; Lai and Nissim, 2024) directly fine-tune models on multilingual mathematical reasoning data generated via machine translation. Another line of research concentrates on leveraging additional components during training. These works either utilize a multilingual encoder (Yoon et al., 2024; Huang et al., 2024b) to facilitate cross-lingual transfer or employ an additional translation model to construct preference signals for preference optimization (She et al., 2024). Recent work (Zhu et al., 2024) proposes a two-stage approach where the model is first learned to translate non-English questions into English for multilingual comprehension and then trained on English reasoning data to enhance multilingual reasoning abilities. In contrast, our study particularly focuses on achieving efficient multilingual reasoning alignment, a perspective that remains under-explored. We precisely fine-tune lower-level layers that are responsible for learning language-specific representations and leverage the model's inherent reasoning ability to facilitate multilingual reasoning alignment. This ensures superior efficiency in one stage without the need for two-stage full parameters training or introducing additional components.

### 7.2 Mechanism of Multilingual Language Processing in LLMs

Recently, multilingual LLMs have garnered significant attention. Numerous studies attempt to explore the mechanism of LLMs in processing multiple languages. Recent research (Chen et al., 2023a; Zhao et al., 2024) reveals that both lower and upper layers of multilingual LLMs are language-dependent. The lower layers are designed to convert inputs from various languages into a high-resource language (*e.g.*, English), while the upper layers perform the reverse function. Additionally, further studies (Tang et al., 2024; Mu et al., 2024) highlight that the proficiency of LLMs in comprehending a particular language is significantly influenced by a small subset of language-specific neurons. Despite their limited number, these neurons play a crucial role in bolstering the multilingual understanding abilities in LLMs. Aligning with this line of research, SLAM further reveals that specific layers are dedicated to handling multilingualism, as evidenced by neuron activation patterns. This advancement significantly deepens the understanding of the multilingual mechanisms in LLMs.

## 8 Conclusion

In this paper, we propose SLAM for efficiently achieving multilingual reasoning alignment in LLMs. Inspired by neuron activations in language abilities, we develop an approach to precisely identify the layers mostly engaging in multilingual comprehension during multilingual reasoning. After that, we fine-tune the FFN sub-layers within the selected layer to enhance the multilingual understanding abilities of LLMs. This enables achieving multilingual reasoning alignment in one stage without compromising LLMs' inherent reasoning abilities. The experimental results on multilingual mathematical reasoning demonstrate the effectiveness and superior efficiency of SLAM. Further analysis reveals that SLAM exhibits significant out-of-domain generalization and can be effectively adapted to other multilingual reasoning tasks.

## Limitations

Our work presents several limitations worth noting. First, to ensure a fair comparison with baseline models, our method primarily conducts experiments using the Llama2 series models. Future work will involve extending our experiments to additional series models to more comprehensively evaluate the generalizability of our method across diverse baseline models. Second, while our method achieves substantial advantages in average accuracy on both in-domain and out-of-domain test sets, the degrees of alignment across different languages result in performance trade-offs. We hypothesize that this issue may be due to the imbalanced data among languages in the X-English translation dataset. In the future, we will conduct an in-depth analysis of this phenomenon.

## Acknowledgements

This work was supported in part by the National Science Foundation of China (Nos. 62276056 and U24A20334), the Natural Science Foundation of Liaoning Province of China (2022-KF-16-01), the Fundamental Research Funds for the Central Universities (Nos. N2216016 and N2316002), the Yunnan Fundamental Research Projects (No. 202401BC070021), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No.B16009).

## References

Nuo Chen, Ning Wu, Shining Liang, Ming Gong, Linjun Shou, Dongmei Zhang, and Jia Li. 2023a. Is bigger and deeper always better? probing llama across scales and layers. *CoRR*, abs/2312.04333.

Nuo Chen, Zinan Zheng, Ning Wu, Linjun Shou, Ming Gong, Yangqiu Song, Dongmei Zhang, and Jia Li. 2023b. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. *CoRR*, abs/2310.20246.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8493–8502. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5484–5495. Association for Computational Linguistics.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2024. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1049–1065. Association for Computational Linguistics.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024a. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.* Just Accepted.

Zixian Huang, Wenhao Zhu, Gong Cheng, Lei Li, and Fei Yuan. 2024b. Mindmerger: Efficient boosting LLM reasoning in non-english languages. *CoRR*, abs/2405.17386.

Huiyuan Lai and Malvina Nissim. 2024. mcot: Multilingual instruction tuning for reasoning consistency in language models. *CoRR*, abs/2406.02301.

Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. 2024. Mugglemath: Assessing the impact of query and response augmentation on math reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 10230–10258. 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual*

*Event, August 1-6, 2021*, pages 1274–1287. Association for Computational Linguistics.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *CoRR*, abs/2308.09583.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Yongyu Mu, Peinan Feng, Zhiquan Cao, Yuzhang Wu, Bei Li, Chenglong Wang, Tong Xiao, Kai Song, Tongran Liu, Chunliang Zhang, and Jingbo Zhu. 2024. Large language models are parallel multilingual learners. *CoRR*, abs/2403.09073.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Libo Qin, Qiguang Chen, Yuhang Zhou, Zhi Chen, Yinghui Li, Lizi Liao, Min Li, Wanxiang Che, and Philip S. Yu. 2024. Multilingual large language model: A survey of resources, taxonomy and frontiers. *CoRR*, abs/2404.04925.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2020, Virtual Event / Atlanta, Georgia, USA, November 9-19, 2020*, page 20. IEEE/ACM.

Noam Shazeer. 2020. GLU variants improve transformer. *CoRR*, abs/2002.05202.

Shuaijie She, Shujian Huang, Wei Zou, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. 2024. MAPO: advancing multilingual reasoning through multilingual alignment-as-preference optimization. *CoRR*, abs/2401.06838.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. *CoRR*, abs/2402.16438.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *CoRR*, abs/2402.10588.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *CoRR*, abs/2309.11674.

Dongkeun Yoon, Joel Jang, Sungdong Kim, Seungone Kim, Sheikh Shafayat, and Minjoon Seo. 2024. Langbridge: Multilingual reasoning without multilingual supervision. *CoRR*, abs/2401.10695.

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. *CoRR*, abs/2309.12284.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. 2024. Mammoth: Building math generalist models through hybrid instruction tuning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? *CoRR*, abs/2402.18815.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question translation training for better multilingual reasoning. *CoRR*, abs/2401.07817.

# A Experimental Details of Multilingual Mathematical Reasoning Task

## A.1 Training Details

We utilize LLaMA-Factory[2] as our training framework. The training is conducted on the MGSM8KINSTRUCT training dataset. Due to the complexity of mathematical reasoning questions (Guo et al., 2024), hallucinations (Huang et al., 2024a) may arise during the translation process, leading to errors in the training data, such as repeated translations. To address this issue, we conduct data quality filtering. This process yields 57,817 question translations and 65,968 answer translations. All models are trained using NVIDIA A800 GPUs. All models are trained for 4 epochs with a batch size of 512, and the learning rate is set to 2e-5. We set the maximum token length to be 1024. We use Deepspeed stage 2 (Rajbhandari et al., 2020) to conduct multi-GPU distributed training, with training precision Bfloat16 enabled.

## A.2 Training Prompts

As shown in Table 7, we adopt the training prompt from Xu et al. (2023). In the template, {*source_lang*} can be replaced with any of the following languages: Bengali, Thai, Swahili, Japanese, Chinese, German, French, Russian, and Spanish. The placeholder {*source sentence*} is substituted with the multilingual mathematical reasoning question (or answer), and {*English sentence*} is replaced with the corresponding English question (or answer) that conveys the same meaning.

## A.3 Evaluation Details

During inference, we use greedy decoding to ensure the determinism of the outputs and set the maximum generation length to 512. We adopt the inference prompt from Yu et al. (2023). The evaluation prompt for both the MGSM and MSVAMP test sets is shown in Table 8.

## A.4 The Training Data of All Baselines

The number of training samples used for all baselines is presented in Table 13.

# B The results of training different sub-layers on MSVAMP test sets

The accuracy of training only the Attention sub-layers, and both the Attention and FFN sub-layers on MSVAMP is shown in Figure 8.

# C Experimental Details of Multilingual Common Sense Reasoning Task

## C.1 Training Details

Initially, we fine-tune the Llama2-7B base model using the English instruction dataset XCSQA-TRAIN to equip the model with fundamental English common sense reasoning abilities. The resulting model is named Llama2-mono. Following Zhu et al. (2024), we use the XCSQA-TEST datasets to construct the X-English translation data for fine-tuning. During the training process of SLAM, we use the same layer selection approach to select the first four layers of the model as multilingualism-handling layers in XCSQA. Subsequently, SLAM trains only the FFN sub-layers within the selected layers of the model to achieve multilingual reasoning alignment. All models are trained for 3 epochs using NVIDIA A800 GPUs. The learning rate is set to 2e-5, with a total batch size of 512 and a maximum input length of 512.

## C.2 Training Prompts

We use the training prompt shown in Table 7. In the template, {*source_lang*} can be replaced with any of the following languages: Arabic, German, Spanish, French, Hindi, Italian, Japanese, Dutch, Polish, Portuguese, Russian, Swahili, Urdu, Vietnamese, and Chinese. The placeholder {*source sentence*} is substituted with the multilingual common sense reasoning question, and {*English sentence*} is replaced with the corresponding English question that conveys the same meaning.

## C.3 Evaluation Details

Following Zhu et al. (2024), we employ XCSQA-DEV for evaluation. During the evaluation, we calculate accuracy by comparing the last label within brackets in the LLM-generated response with the gold answer. Specifically, we use greedy decoding to ensure the determinism of the outputs and set the maximum generation length to 512. The evaluation prompt is shown in Table 9.

## C.4 Languages Included in XCSQA

The XCSQA test sets include 15 non-English languages: Arabic, German, Spanish, French, Hindi, Italian, Japanese, Dutch, Polish, Portuguese, Russian, Swahili, Urdu, Vietnamese, and Chinese. The abbreviations for these languages are as follows: ar, de, es, fr, hi, it, ja, nl, pl, pt, ru, sw, ur, vi, and zh.

[2] https://github.com/hiyouga/LLaMA-Factory

| | |
|---|---|
| **Training Prompt** | Translate this from [{*source_lang*}] to [English]:\n[{*source_lang*}]: {*source sentence*}\n[English]: {*English sentence*} |

Table 7: The prompt used to train the FFN sub-layers within the multilingualism-handling layers.

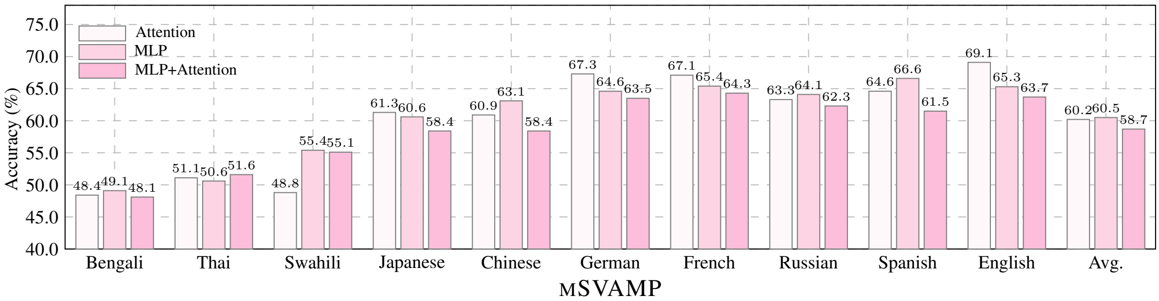


Figure 8: The accuracy (%) of training different sub-layers on MSVAMP test sets.

### C.5 Scores for All Languages

Table 10 presents the scores for all 16 languages on the XCSQA test sets.

## D Scores Across All Languages for the RFT-MuggleMath Model

Consistent with using MetaMath as the base model, we apply the identical layer selection approach and settings for the RFT-MuggleMath model. For more details on the overlap between non-English and English activated neurons, and the number of activated neurons during the layer selection process, we present the results of all layers in Figure 13 and Figure 14, respectively. Specifically, we select the first five layers of the RFT-MuggleMath models as multilingualism-handling layers. Tables 11 and Tables 12 present the accuracy for all languages on MGSM8K and MSVAMP test sets, respectively.

## E Comparison of neuron activation before and after training of SLAM-13B

We present the comparison results in Figure 9.

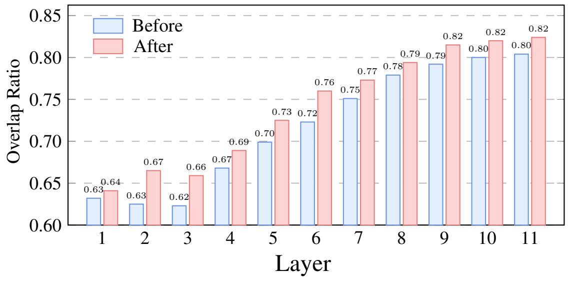


Figure 9: The comparison of the overlap ratio in the MetaMath-13B model, before and after training.

| Model | Lang | Number |
|---|---|---|
| **MAmmoTH** (Yue et al., 2024) | 1 | 262,039 |
| **WizardMath** (Luo et al., 2023) | 1 | 96,000 |
| **MetaMath** (Yu et al., 2023) | 1 | 395,000 |
| **MathOctopus** (Chen et al., 2023b) | 10 | 73,559 |
| **QAlign** (Zhu et al., 2024) | 10 | 468,559 |

Table 13: “Lang” denotes the number of languages covered, and “Number” denotes the total samples used to train the models.

| | |
|---|---|
| **Inference Prompt** | Below is an instruction that describes a task.\n Write a response that appropriately completes the request.\n\n### Instruction:\n{*query*}\n\n### Response: Let's think step by step. |

Table 8: Prompt utilized to evaluate the model on the MGSM and MSVAMP test sets.

| | |
|---|---|
| **Inference Prompt** | Below is an instruction that describes a task.\n Write a response that appropriately completes the request.\n\n### Instruction:\n{*query*}\n\n### Response: |

Table 9: Prompt utilized to evaluate the model on the XCSQA test sets.

| **Model** | **Ar** | **De** | **Es** | **Fr** | **Hi** | **It** | **Ja** | **Nl** | **Pl** | **Pt** | **Ru** | **Sw** | **Ur** | **Vi** | **Zh** | **En** | **Avg.** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Llama2-mono | 31.7 | 58.4 | 63.6 | 58.3 | 31.9 | 58.7 | 49.8 | 55.4 | 53.4 | 60.7 | 55.7 | 22.1 | 25.4 | 47.4 | 52.3 | 77.6 | 50.1 |
| +SLAM | 36.3 | 59.9 | 63.8 | 59.4 | 36.1 | 59.4 | 52.3 | 57.7 | 55.2 | 61.4 | 56.6 | 27.1 | 27.8 | 49.8 | 54.6 | 77.7 | 52.1 |
| Improvements | **+4.6** | **+1.5** | **+0.2** | **+1.1** | **+4.2** | **+0.7** | **+2.5** | **+2.3** | **+1.8** | **+0.7** | **+0.9** | **+5.0** | **+2.4** | **+2.4** | **+2.3** | **+0.1** | **2.0** |

Table 10: The accuracy (%) for all languages on the XCSQA test sets.

| **Model** | **Training Cost** | **Trained Param.** | **Bn** | **Th** | **Sw** | **Ja** | **Zh** | **De** | **Fr** | **Ru** | **Es** | **En** | **Avg.** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | *7B Models* | | | | | | | |
| RFT | 0.8× | 100.0% | 4.4 | 6.0 | 4.0 | 34.4 | 38.4 | 56.0 | **56.0** | 48.4 | 54.8 | **67.6** | 37.0 |
| RFT+SLAM | 1.0× | 6.7% | **27.6** | **30.0** | **33.6** | **41.6** | **42.0** | **60.4** | 55.2 | **53.2** | **60.0** | 65.6 | **46.9** |
| | | | | | | *13B Models* | | | | | | | |
| RFT | 2.6× | 100.0% | 11.2 | 5.2 | 7.2 | 48.8 | 50.8 | **63.2** | 66.8 | **64.8** | 67.6 | **75.2** | 46.0 |
| RFT+SLAM | 1.0× | 5.4% | **42.0** | **41.6** | **42.4** | **54.0** | **51.6** | 61.2 | **68.0** | 61.2 | **68.8** | 72.0 | **56.3** |

Table 11: The accuracy (%) on the in-domain MGSM8K test sets of RFT-MuggleMath models.

| **Model** | **Training Cost** | **Trained Param.** | **Bn** | **Th** | **Sw** | **Ja** | **Zh** | **De** | **Fr** | **Ru** | **Es** | **En** | **Avg.** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | *7B Models* | | | | | | | |
| RFT | 0.8× | 100.0% | 14.0 | 12.5 | 9.9 | 44.2 | 46.4 | 53.5 | 56.1 | 46.7 | 59.1 | **59.6** | 40.2 |
| RFT+SLAM | 1.0× | 6.7% | **40.6** | **44.2** | **44.7** | **51.4** | **50.2** | **56.8** | **58.3** | **50.3** | **59.6** | 59.5 | **51.6** |
| | | | | | | *13B Models* | | | | | | | |
| RFT | 2.6× | 100.0% | 20.2 | 20.7 | 14.8 | 55.7 | 52.8 | **62.3** | **62.7** | 59.2 | 61.8 | **66.3** | 47.6 |
| RFT+SLAM | 1.0× | 5.4% | **45.3** | **46.9** | **50.3** | **58.5** | **56.1** | 61.7 | 61.8 | **60.7** | **62.5** | 65.5 | **56.9** |

Table 12: The accuracy (%) on the out-of-domain MSVAMP test sets of RFT-MuggleMath models.

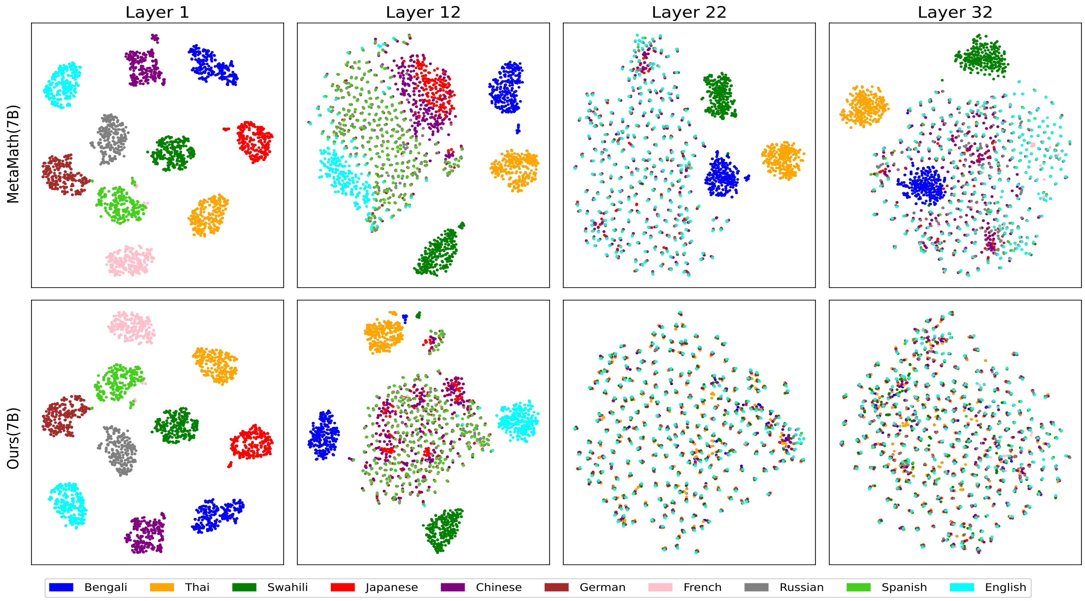


Figure 10: The visualization of the final token representations from multilingual questions is performed using T-SNE for dimension reduction. The distributions in MetaMath-7B at the 1th, 12th, 22th, and 32th layers are compared before and after training. Different colors represent different languages.

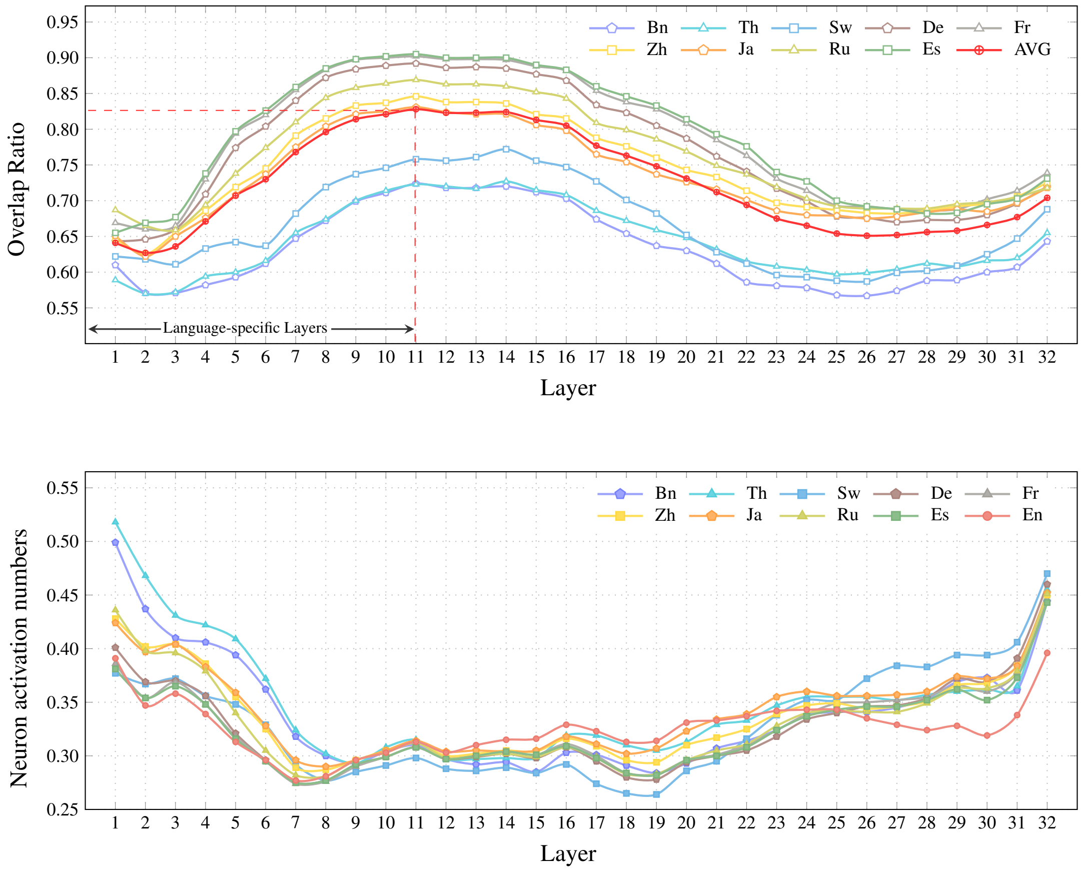
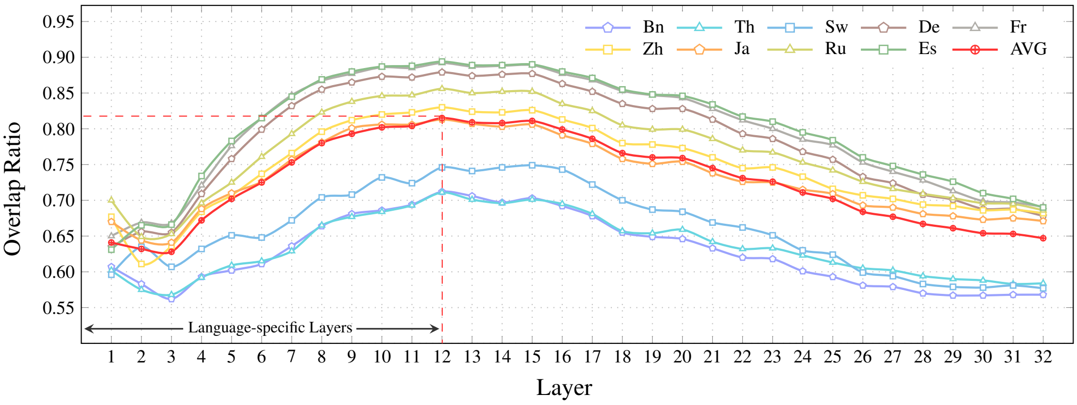


Figure 11: The overlap between non-English and English activated neurons and the normalized number of activated neurons across all languages in the MetaMath-7B model.



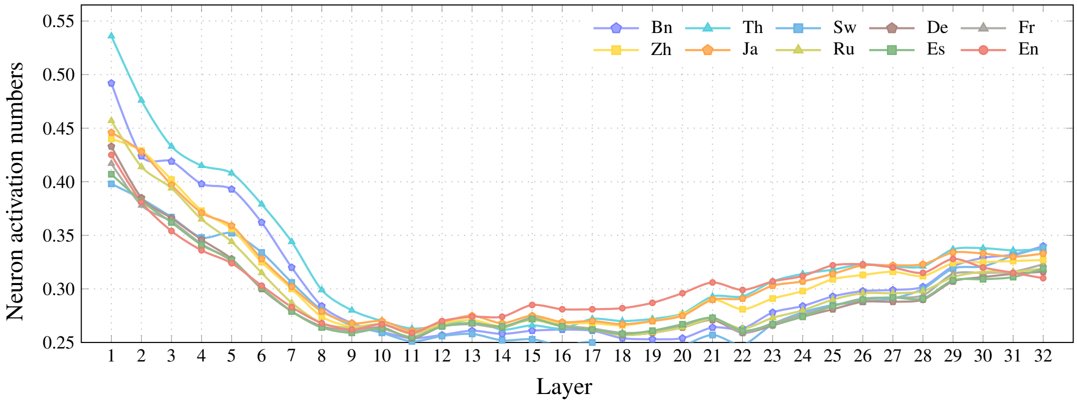
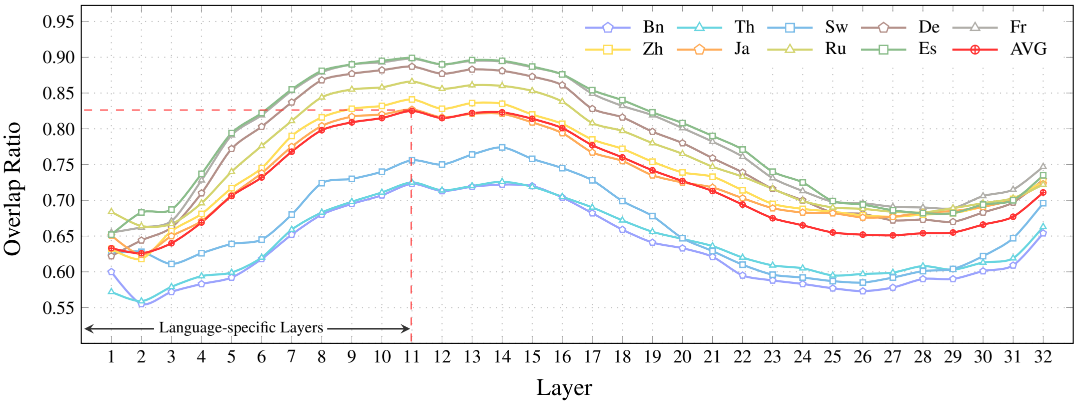


Figure 12: The overlap between non-English and English activated neurons and the normalized number of activated neurons across all languages in the MetaMath-13B model.



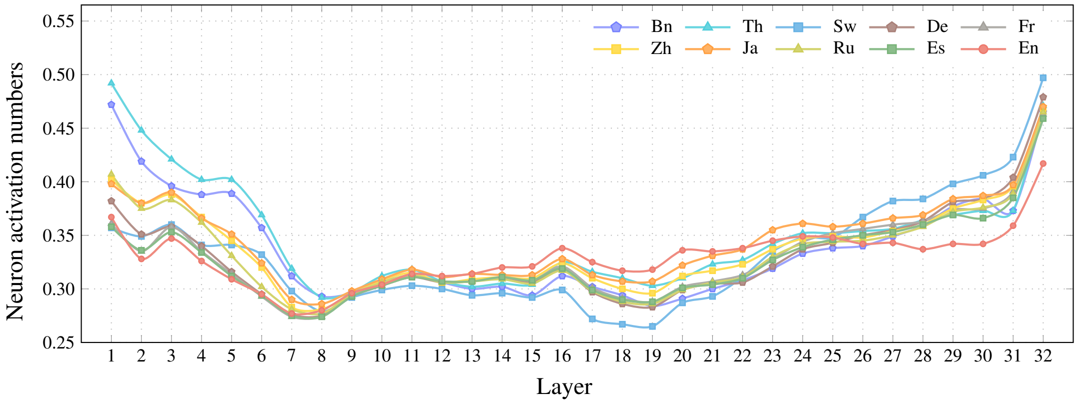


Figure 13: The overlap between non-English and English activated neurons and the normalized number of activated neurons across all languages in the RFT-MuggleMath-7B model.

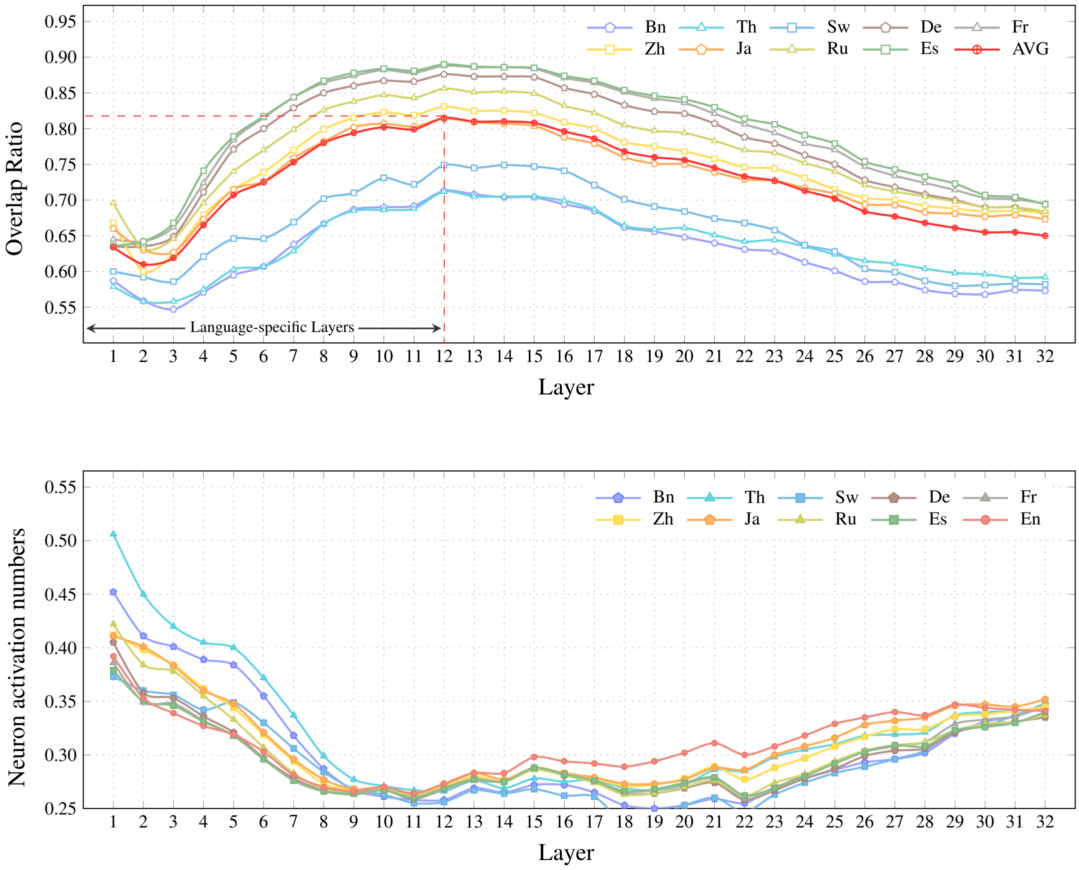


Figure 14: The overlap between non-English and English activated neurons and the normalized number of activated neurons across all languages in the RFT-MuggleMath-13B model.