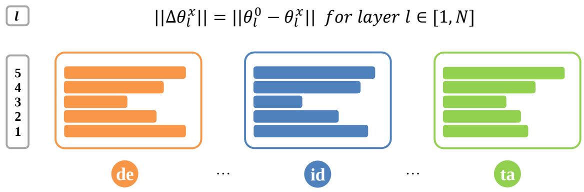
Group then Scale: Dynamic Mixture-of-Experts Multilingual Language Model

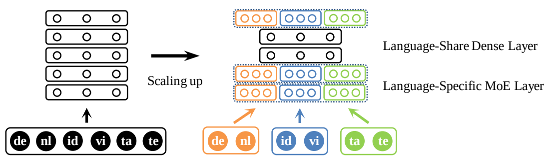
Chong Li, Yingzhuo Deng, Jiajun Zhang, Chengqing Zong*
State Key Laboratory of Multimodal Artificial Intelligence Systems,
Institute of Automation, CAS, Beijing, China
School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
{lichong2021, dengyingzhuo2024}@ia.ac.cn, {jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

The curse of multilinguality phenomenon is a fundamental problem of multilingual Large Language Models (LLMs), where the competition between massive languages results in inferior performance. It mainly comes from limited capacity and negative transfer between dissimilar languages. To address this issue, we propose a method to dynamically group and scale up the parameters of multilingual LLM while boosting positive transfer among similar languages. Specifically, the model is first tuned on monolingual corpus to determine the parameter deviation in each layer and quantify the similarity between languages. Layers with more deviations are extended to mixture-of-experts layers to reduce competition between languages, where one expert module serves one group of similar languages. Experimental results on 18 to 128 languages show that our method reduces the negative transfer between languages and significantly boosts multilingual performance with fewer parameters. Such language group specialization on experts benefits the new language adaptation and reduces the inference on the previous multilingual knowledge learned. [1]



(a) Calculate the parameter deviation for each language.



(b) Dynamically scale up multilingual LLM.

Figure 1: (a) We first statisticize layer-wise parameter deviation of the multilingual language model for each language, (b) then dynamically scale up layers with more deviations into mixture-of-experts layers for language groups divided by language similarity.

1 Introduction

After training on the massive multilingual corpus, large language models obtain impressive multilingual abilities, e.g., cross-lingual natural language understanding (Xue et al., 2021) and in-context learning (Lin et al., 2022; Scao et al., 2023; Wei et al., 2023b; Anil et al., 2023; Üstün et al., 2024). However, their performance in medium- to low-resource languages, still lags behind that of high-resource languages (Lai et al., 2023; Asai et al., 2024; Li et al., 2024), and is hindered by the *curse of multilinguality* phenomenon (Aharoni et al., 2019; Wu and Dredze, 2020). It is found that the limited capacity and negative language transfer mainly contribute to the curse of multilinguality phenomenon (Chang et al., 2024). Thus, our key research problem lies on: *How to flexibly augment the capacity of LLM for massive languages?*

To address the research problem above, Pfeiffer et al. (2022) fine-tuned a module for each new language to augment parameters. The additional language identification process hinders its general application and affects the inference performance if misclassification. Blevins et al. (2024) trained models for new languages using the multilingual base model as initialization, and assembled them with vanilla models during inference. It largely increases the inference cost and the amount of model parameters which linearly grows with the number of languages involved.

In contrast, we introduce language specialization to the mixture-of-experts structure to scale up the parameters of the model. In particular, monolingual corpus from each language x is first adopted to tune the model and obtain the layer-wise parameter deviation $\Delta\theta_l^x$ like Figure 1(a). Layers near the

*Corresponding author.
[1] Our code and model weights are available at https://github.com/ZNLP/DMoE

Findings of the Association for Computational Linguistics: ACL 2025, pages 1730–1754
July 27 - August 1, 2025

input and output of LLM are often found with more derivation than the others (refer to Figure 10 in Appendix B for more details). We argue that layers with more deviation require more capacity to contain language-specific knowledge, while the other layers can be shared with all languages, like the "concept space" in the multilingual LLM (Wendler et al., 2024). Thus, the former is extended to the mixture-of-experts layer, and the parameter of each expert is tuned by a group of similar languages like Figure 1(b). It aims to precisely exploit parameters during scaling up and keep a similar inference cost for each token. Such designation is also beneficial for extending new languages while reducing the effect on the previously learned languages. Given a new language to adapt, we first determine its similarity between existing language groups, then copy and fine-tune the expert for the most similar language group to achieve a better transferring performance and alleviate catastrophic forgetting. The experimental results on 18 to 128 languages show that our method significantly improves multilingual performance and mitigates the curse of multilinguality phenomenon. The improvement in perplexity reaches 11.4% over the continual pre-training method and even surpasses X-ELM (Blevins et al., 2024) 9.6% with 3.6x fewer parameters on average. In summary, our contributions lie in the following:

- We propose a mixture-of-experts training framework to flexibly group languages and dynamically augment the capacity of multilingual large language models.

- We formalize language grouping into a max-imin optimization problem and introduce a token-level language classification loss to specialize mixture-of-experts layers.

- Extensive experiments on 18 to 128 languages demonstrate the effectiveness of our method which largely mitigates the curse of multilinguality phenomenon.

2 Related Works

2.1 Quantify Language Similarity

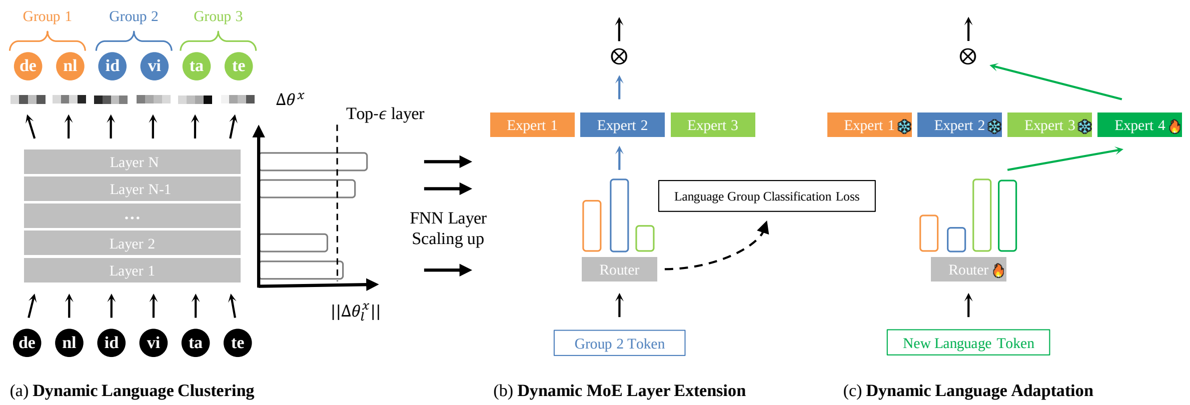
The LANG2VEC method (Littell et al., 2017) represents languages as typological, geographical, and phylogenetic vectors to calculate the similarity between them and has been widely adopted (Blevins et al., 2024; Chang et al., 2024). However, they rely exclusively on language- or data-intrinsic features, ignoring the characteristics of the downstream models. To address this limitation, prior works have proposed model-specific representations, such as learnable language vectors (Tsvetkov et al., 2016; Östling and Tiedemann, 2017; Johnson et al., 2017) and leveraging hidden states of the model (Malaviya et al., 2017) or gradients of the loss function (Wang and Zhang, 2022) to derive language representations. These model-specific approaches often require training from scratch or incur high computational costs by recalculating similarity during training. In contrast, our method utilizes parameter deviations as language representations, enabling stable similarity estimation through fine-tuning the downstream model on a small dataset in the preparatory phase.

2.2 Mixture of Experts

Since the concept of mixture-of-experts proposed (Jacobs et al., 1991; Jordan and Jacobs, 1994), it has been widely applied to SVM (Collobert et al., 2001), Gaussian process (Tresp, 2000), Dirichlet process (Shahbaba and Neal, 2009), LSTM (Theis and Bethge, 2015; Shazeer et al., 2017), and Transformer (Lepikhin et al., 2021; Roller et al., 2021; Fedus et al., 2022; Dai et al., 2022; MistralAI, 2023; Dai et al., 2024; Zhou et al., 2025). Adding more experts scales up the total capacity of the model while keeping similar inference costs on each token. Previous studies mainly focus on designing a better load-balancing routing strategy (Roller et al., 2021; Fedus et al., 2022; Zhou et al., 2022) and a training method (Sukhbaatar et al., 2024). Our work is similar to X-MOD (Pfeiffer et al., 2022), which trains an adapter module for each language in all layers. The main differences lie in 1) grouping similar languages in each expert to boost cross-lingual transfer rather than allocating one adapter for each language. 2) Text for inference can be directly input to our model without specifying languages which is inflexible and required for X-MOD.

2.3 Multilingual Large Language Model

The pre-training methods of multilingual large language models (Conneau et al., 2020; Lin et al., 2022; Scao et al., 2023; Yang et al., 2023; Wei et al., 2023b; Üstün et al., 2024; Dang et al., 2024; Ming et al., 2024; Ji et al., 2025) are mainly extended from the one for the monolingual corpus (Radford et al., 2018; Devlin et al., 2019; Lewis et al., 2020; Raffel et al., 2020), and relied on a



(a) **Dynamic Language Clustering** (b) **Dynamic MoE Layer Extension** (c) **Dynamic Language Adaptation**

Figure 2: The overview of our method to group and scale up the multilingual LLM. (a) Given pre-training languages, we first determine their parameter deviation $\Delta\theta^x$ on the model, then group similar languages by the similarity of $\Delta\theta^x$. (b) These layers with higher $||\Delta\theta_l^x||$ are extended to MoE layers, where each expert is tuned with tokens from the corresponding language group to specialize. (c) To adapt to the new language, we copy the multilingual expert from the most similar language group, then only fine-tune the router and expert added.

balanced sampling method to mitigate the performance gap between languages.

To mitigate the curse of multilinguality, Blevins et al. (2024) applied the Branch-Train-Merge method (Li et al., 2022) on the training of multilingual language model, where one model serves for a group of languages, and assembled output of top-m models after language identification during inference. In contrast, our method is motivated by the distribution of parameter deviation during the training of multilingual large language models and strives to scale up the parameter on the language-specific layers. It keeps a similar cost without additional language classification while augmenting the capacity of the multilingual language model during inference.

3 Method

As shown in Figure 2, to train a **D**ynamic **M**ixture-**o**f-**E**xperts model (**DMoE**), we first fine-tune the model on the monolingual corpus to obtain the parameter deviation for each language. Then the parameter deviation is used to cluster similar languages (Section 3.1) and determine layers to extend parameters (Section 3.2). Besides, new languages for adaption are also dynamically assigned to the most similar language cluster to mitigate catastrophic forgetting (Section 3.3).

3.1 Dynamic Language Clustering

The quality of the clustering method is primarily influenced by the choice of similarity metric, making the determination of an appropriate metric central to its effectiveness. We first obtain the parameter deviation of the model by fine-tuning only ten steps, investigated in Appendix B, and take it as a representation of distinctive characteristics for each language. Given the high-dimensional nature of the parameter deviation, we employ cosine similarity as the metric to measure the similarity between languages. To satisfy the clustering process, we define the intra-group language similarity as follows:

$$\mathrm{Sim}(\theta, G_k) = \min_{x,y \in G_k} \frac{\Delta\theta^x \cdot \Delta\theta^y}{||\Delta\theta^x|| ||\Delta\theta^y||} \tag{1}$$

where G_k is the k-th group of languages, $\Delta\theta^x$ and $\Delta\theta^y$ are the parameter deviation of language x and y respectively on the parameter θ, and $\Delta\theta^x = [\Delta\theta_1^x, \Delta\theta_2^x, \cdots, \Delta\theta_N^x]$ is the concatenation of the parameter deviation from all N layers after fine-tuning on language x.

A higher intra-group similarity indicates that the languages within the group are more similar, resulting in less conflict between them. This reduces the potential for gradient conflicts during the continuous pre-training on different languages, making it more appropriate to share parameters with the same expert. Therefore, we can perform clustering by maximizing the similarity within each group, which can be formalized as follows:

$$\max_{G_1, G_2, \ldots, G_K} \sum_{k=1}^{K} \mathrm{Sim}(\theta, G_k) \tag{2}$$

However, obtaining the global optimal solution to this problem is NP-Hard. Additionally, the number of languages in each group needs to be balanced to enhance the utilization of experts. To address these challenges, we employ a greedy algorithm. The pseudo-code is provided in Algorithm 1.

Algorithm 1 Balanced Language Clustering

Input: Parameter deviations for different languages $\Delta\Theta = \{\Delta\theta^1, \Delta\theta^2, \dots, \Delta\theta^x\}$, Number of groups K

Output: Language clustering result *Groups*

```
Initialize Groups = {}
while ΔΘ is not empty do
    Compute the cosine similarity between languages (i, j) for all Δθ^i, Δθ^j ∈ ΔΘ
    Find the most similar pair of languages (i*, j*)
    Merge languages i* and j* to form a group: G = {i*, j*}
    Remove i* and j* from ΔΘ
    while |G| < |ΔΘ|/K do
        Compute the intra-group similarity (Eq. 1) of G ∪ {m} for all Δθ^m ∈ ΔΘ
        Find the group G ∪ {m*} that maximizes the intra-group similarity
        Add m* to G
        Remove m* from ΔΘ
    end while
    Add group G to Groups
end while
Return: Groups
```

3.2 Dynamic MoE Layer Extension

We assume that the layers with large parameter deviations are important and language-specific during fine-tuning, requiring additional capacities to mitigate the conflicts between languages. Thus the top-ϵ of dense layers and extended to the mixture-of-experts layers with g experts, where $\epsilon \in [0, 1]$ and $g \in \mathbb{N}^+$ are hyper-parameters. Each expert is initialized from the parameters of the original dense layer, while the ones of router are randomly initialized. Corpora from the same language group are organized in the same batch and used to fine-tune the parameters of the corresponding expert. We also train the router with the following language group classification loss:

$$\mathcal{L}_{RC}(\theta) = -\sum_{x}\sum_{i=1}^{M}\left[\log\left(\mathrm{P}_i(l|x;\theta)\right)\right] \quad (3)$$

where x is a token from the language group l, and $\mathrm{P}_i(\cdot)$ is the probability estimated by the router at the i-th MoE layer. Thus the training loss comes to the weighted sum of **C**ausal **L**anguage **M**odeling (**CLM**) loss and the above language group classification loss:

$$\mathcal{L}(\theta) = \mathcal{L}_{CLM}(\theta) + \alpha\mathcal{L}_{RC}(\theta) \quad (4)$$

where $\alpha \in \mathbb{R}^+$ is a hyper-parameter.

3.3 Dynamic Language Adaptation

Given new languages to adapt, we introduce a method to augment their capacity while reducing the inference to other languages learned. Specifically, samples from the new language are first input to experts through a hard routing strategy. The multilingual expert with the lowest perplexity is considered the most similar one, which is copied and only fine-tuned for fast adaptation. It is noted that the other part of parameters, like the shared dense layers and the other experts, are frozen to avoid catastrophic forgetting during the new language learning (Winata et al., 2023).

4 Experiments

4.1 Experiments Settings

Large Language Models We adopt the multilingual Bloom (Scao et al., 2023) and English-centric Gemma (Team et al., 2024) series models in this work.

Corpus Two multilingual corpora, CulturaX (Nguyen et al., 2024) and MADLAD-400 (Kudugunta et al., 2023), are used in this work. We set the language sampling exponent to 0.3 following mT5 (Xue et al., 2021).

Evaluation Tasks There are five multilingual tasks, covering natural language inference (Conneau et al., 2018), paraphrase detection (Yang et al., 2019), and multilingual reasoning tasks (Ponti et al., 2020; Lin et al., 2022; Tikhonov and Ryabinin, 2021), selected to evaluate the performance of multilingual LLMs. To reduce the variability of prompt and evaluation method, we choose the default prompt from the language model evaluation harness framework (Gao et al., 2024).

Baselines

- **+ Pre-train**, where the base model continues to pre-train on the same multilingual corpus. It denotes the performance of the vanilla continual multilingual pre-training method.
- **X-ELM** (Blevins et al., 2024) trains a model for two similar languages, and ensembles outputs from top-m models during inference, where m is set to 2 in this work.

- **Branch-Train-Mix** (Sukhbaatar et al., 2024) trains models specialized for one domain and merges them to obtain a mixture-of-experts model, which shows significantly better performance than Branch-Train-Merge (Li et al., 2022). We apply our dynamic language clustering results to it, serving as a strong multilingual mixture-of-experts baseline.

To conduct a fair comparison, the total training token amount is the same for all methods. Hyperparameters are reported in Appendix A.

4.2 Results on 18 Languages

We first conduct experiments on 18 languages from 9 language families. Figure 3 illustrates the pair-wise language similarity calculated by the $\Delta\theta$ of $\text{BLOOM}_{560\text{M}}$, and more details of other models refer to Appendix C.1. It mostly exhibits some linguistic characteristics. For example, Tamil (ta) and Telugu (te), which both come from the Dravidian language family, show a similar trend among languages and have higher similarity than the other languages. Based on the pair-wise similarity, languages are divided into multiple groups by Algorithm 1, and results are reported in Table 1. Appendix C.1 reports the results of other language models investigated.

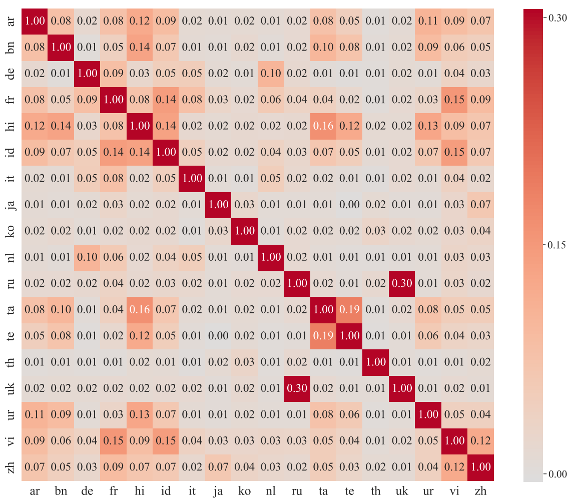


Figure 3: The cosine similarity between 18 languages for $\text{BLOOM}_{560\text{M}}$.

After language clustering, we continue pre-training on 18 languages under a 6.5B tokens amount budget from CulturaX. The perplexity results of models across different parameter amounts are shown in Table 2. We can find that scaling up model parameters brings better language modeling performance compared with the continued

	ar	ur	bn	it	de	nl	ta	te	hi	id	fr	vi	ru	uk	th	ko	ja	zh
9G	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9
6G	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6
$6G^{(L)}$	1	1	2	3	4	3	5	5	5	6	1	6	4	4	6	2	2	3
$6G^{(R)}$	1	2	5	4	6	6	2	3	3	4	6	1	3	5	4	2	5	1
3G	1	1	1	2	2	2	1	3	1	1	2	2	3	3	3	3	3	2
2G	1	1	1	2	2	2	1	2	1	1	1	1	2	2	2	2	2	1

Table 1: The grouping results of $\text{BLOOM}_{560\text{M}}$, where "**2G**" denotes the result that divides into 2 groups. "**$6G^{(L)}$**" and "**$6G^{(R)}$**" indicate the LANG2VEC and random language clustering results, respectively.

pre-training method ("+ Pre-train"). DMoE obtains the highest average improvement on perplexity (+11.4% over "+ Pre-train") than the other two strong baseline methods (+0.8% and +2.2% respectively) and requires the least additional parameters. It is noted that DMoE with 9 experts outperforms X-ELM (Blevins et al., 2024) using 3.6x less parameters. The improvement mostly comes from unseen languages like German (+17.9%) and low-resource languages like Urdu (+13.6%). Appendix C.2 shows similar results of Qwen2.5 base models.

Figure 4 illustrates the trend of perplexity improvement over the continual pre-training baseline using DMoE across 18 languages. It can be found that languages with higher perplexity benefit more from our method. Moreover, with more language groups divided, DMoE shows better multilingual language modeling performance.

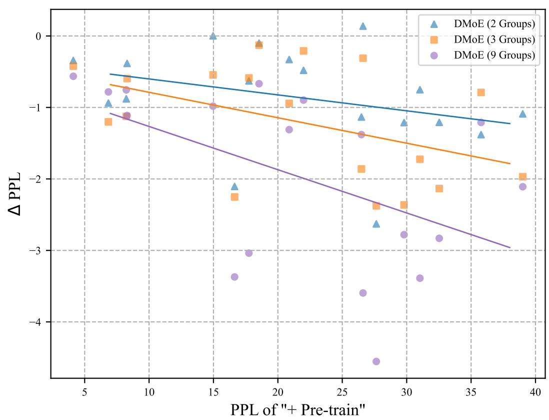


Figure 4: The improvement of DMoE comparing to the continual pre-training baseline method on $\text{BLOOM}_{560\text{M}}$.

Transfer language similarities. To evaluate the effectiveness of dynamic language clustering (Section 3.1), we replace the 6-group dividing into the LANG2VEC (Littell et al., 2017) and random grouping results in Table 1. The "w/ Random Cluster" row reports the language modeling result on $\text{BLOOM}_{560\text{M}}$, which is worse than the DMoE

Model	#Param.	ar	de†	fr	it†	ja†	nl†	ru†	zh	bn	hi	id	ko†	th†	uk†	vi†	ta	te	ur	Avg ↓
		High								Medium							Low			
BLOOM$_{560M}$	560M	56.7	126.4	37.4	85.3	55.7	129.5	31.3	59.0	42.3	32.7	47.4	25.7	14.0	39.7	27.9	64.2	88.4	60.0	56.9
+ Pre-train	560M	39.0	27.6	22.0	17.8	15.0	16.6	8.2	36.0	26.5	20.9	32.6	8.4	4.4	6.8	18.5	31.0	26.9	29.8	21.6
X-ELM	5.03B	**35.2**	34.6	21.3	17.3	18.0	21.5	9.8	37.9	25.1	19.5	**28.8**	9.3	4.4	8.5	18.9	**26.9**	**23.0**	27.5	21.5
Branch-Train-Mix	1.57B	39.4	25.8	21.6	17.1	14.7	15.3	8.0	35.8	26.4	21.0	32.1	8.1	4.1	6.6	18.1	31.4	27.0	30.0	21.2
DMoE (2 Groups)	635M	37.9	25.0	21.5	17.1	15.0	14.5	7.4	34.4	25.4	20.5	31.3	7.9	3.8	5.9	18.4	30.3	26.7	28.6	20.6
DMoE (3 Groups)	710M	37.0	25.3	21.8	17.2	14.4	14.4	**7.1**	35.0	24.6	19.9	30.4	7.7	3.7	**5.6**	18.4	29.3	26.3	27.4	20.3
DMoE (6 Groups)	937M	36.2	23.5	**20.9**	16.0	13.8	13.6	7.3	**34.1**	23.9	**19.1**	30.7	7.4	3.9	5.9	**17.6**	27.9	23.4	**26.8**	**19.5**
w/ Gemma Clusters	937M	36.7	23.5	21.4	16.1	13.8	13.6	7.3	**34.1**	**23.7**	19.7	31.8	7.4	3.6	**5.6**	17.7	27.8	23.3	27.1	19.7
w/ LANG2VEC Clusters	937M	37.6	25.2	**20.9**	15.8	**13.4**	**13.3**	7.6	34.2	25.0	19.3	30.3	**7.2**	**3.5**	6.2	17.7	28.0	23.6	28.1	19.8
w/ Random Clusters	937M	38.5	25.7	21.5	17.0	14.6	15.2	7.9	35.3	25.7	20.6	31.8	8.1	4.1	6.5	18.1	30.6	26.1	29.4	20.9
w/o Class. Loss	937M	36.5	24.2	21.2	16.4	14.4	13.9	7.2	34.8	24.6	19.7	30.9	7.7	3.8	5.8	18.2	28.4	24.1	27.6	20.0
DMoE (9 Groups)	1.16B	36.9	**23.1**	21.1	**14.7**	14.0	**13.3**	7.5	34.6	25.1	19.6	29.7	**7.2**	3.6	6.1	17.9	27.6	**23.0**	27.0	**19.5**
BLOOM$_{1.7B}$	1.72B	41.2	63.0	24.3	44.0	35.2	63.0	20.1	40.1	27.0	22.6	32.7	17.4	9.8	23.9	19.2	40.1	43.6	36.3	33.5
+ Pre-train	1.72B	25.1	21.0	15.3	15.5	11.7	17.4	7.0	**23.4**	**16.8**	15.3	23.0	7.8	4.3	8.5	13.1	21.6	21.1	22.6	16.1
X-ELM	15.50B	**24.9**	20.2	**15.0**	12.1	12.1	12.3	7.1	24.3	17.4	**14.4**	**21.0**	7.5	3.8	6.0	**12.9**	**19.3**	**16.6**	**19.4**	14.8
Branch-Train-Mix	5.75B	25.4	18.7	16.0	15.2	11.3	13.6	6.0	23.6	17.6	15.3	24.1	7.4	3.7	7.0	13.4	23.1	21.1	20.5	15.7
DMoE (3 Groups)	2.33B	26.3	17.5	16.0	12.5	11.0	10.5	**5.7**	25.2	17.5	15.1	22.4	6.3	3.2	**4.5**	13.8	21.1	19.6	19.9	14.9
DMoE (6 Groups)	3.23B	26.1	17.2	15.8	12.4	**10.7**	10.2	5.8	24.9	17.4	14.5	22.8	6.1	3.3	4.6	13.3	20.3	17.5	19.6	14.6
DMoE (9 Groups)	4.14B	26.6	**16.8**	16.0	**11.5**	10.8	**10.0**	5.9	25.1	17.8	14.8	22.2	**5.9**	**3.1**	4.8	13.5	20.1	17.0	19.8	**14.5**
Gemma$_{2B}$	2.51B	54.8	12.5	23.6	13.4	11.1	11.4	5.1	69.1	68.9	44.2	45.5	5.6	2.8	4.0	19.8	70.5	62.8	54.2	32.2
+ Pre-train	2.51B	28.8	10.1	17.1	9.3	7.3	6.5	4.1	29.4	21.7	14.9	21.9	3.8	2.3	**3.1**	13.6	20.4	17.0	20.4	14.0
X-ELM	22.56B	30.2	11.1	19.3	10.2	8.0	7.3	4.3	34.6	22.7	15.3	23.9	3.9	2.3	3.3	14.4	19.9	16.5	22.8	15.0
Branch-Train-Mix	11.57B	27.7	10.7	18.2	9.7	7.5	6.8	4.2	30.6	18.5	14.4	23.8	3.9	2.3	3.2	13.6	17.4	14.5	19.3	13.7
DMoE (9 Groups)	6.53B	**24.8**	**9.9**	**16.8**	**9.1**	**7.0**	**6.3**	**4.0**	**27.8**	**17.6**	**12.9**	**19.6**	**3.6**	**2.2**	**3.1**	**12.2**	**15.3**	**12.8**	**17.3**	**12.4**

Table 2: The normalized perplexity on the valid split of CulturaX. The perplexity is normalized to the vocabulary of Bloom following Wei et al. (2023a). † denotes the language unseen in the pre-training of BLOOM. "**High**", "**Medium**", and "**Low**" indicates the available amount of linguistic resources. The best and second results are denoted in **bold** and underlined, correspondingly.

model (+1.4 PPL on average). We argue that the poor result arises from the negative transfer between dissimilar languages, especially deteriorating the performance of low-resource languages like Urdu (+2.6 PPL). And LANG2VEC brings an inferior result, +0.3 PPL on average, comparing our model-specific method. It demonstrates that better language clustering results can bring better cross-lingual transfer to the low-resource languages. The language clustering result of Gemma$_{2B}$ is further applied on BLOOM$_{560M}$ to investigate the transferability of our method. It is interesting to find that BLOOM$_{560M}$ with Gemma clusters is slightly worse than the vanilla model in Table 2. Although our method shows some transferability, we still recommend using language similarity classification based on its parameter derivation for better results.

Trade-off between learning and forgetting. When new languages come for multilingual models to adapt, it is better to achieve fast adaptation while alleviating the catastrophic forgetting of languages learned (Sun et al., 2020; Zhao et al., 2022; Wu et al., 2024). We adopt 4 unseen languages: Belarusian (be), Malayalam (ml), Marathi (mr), and Serbian (sr) to evaluate the performance of models. As shown in Table 3, the dense model suffers a catastrophic forgetting of the 18 languages learned after **L**anguage **A**daptation **P**re-**T**raining (**LAPT**), especially on the medium and low resource languages (+2.0 PPL). In contrast, the proposed **D**ynamic **L**anguage **A**daptation (**DLA**) method (Section 3.3) for DMoE achieves better adaptation results on new languages, and mitigates the catastrophic forgetting of the languages learned (only +0.7 PPL). It benefits from language-specific expert design and fine-tuning method, which provides a better module for initialization and reduces the inference to the modules learned.

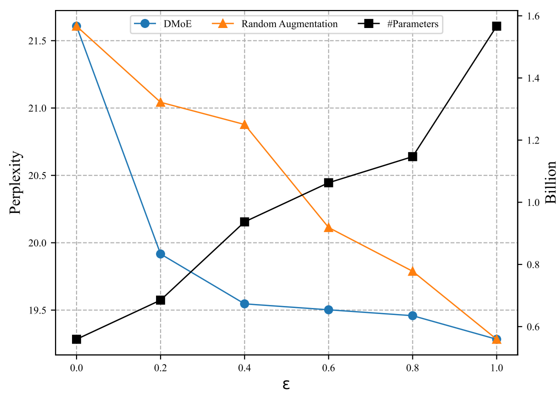
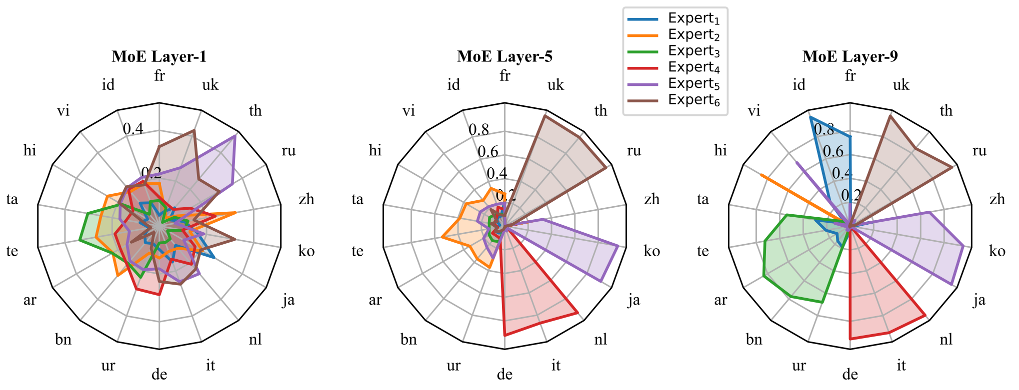
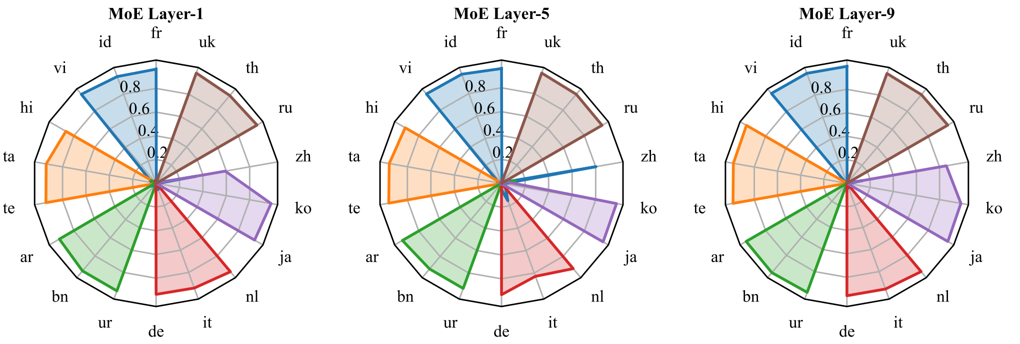


Figure 5: The average perplexity of DMoE across 18 languages under different ϵ using BLOOM$_{560M}$, where $\epsilon = 1$ denotes all layers are extended to MoE layers.



(a) DMoE (6 Groups) w/o language group classification loss.



(b) DMoE (6 Groups) w/ language group classification loss.

Figure 6: The router distribution of top-1 expert for texts in different languages. (a) DMoE trained with randomly initialized router. (b) DMoE trained with language classification loss. Refer to Appendix C.3 for more details.

	New Languages				Old Languages		
Model	be	ml	mr	sr	High	Medium	Low
Gemma_{2B}	10.0	7.1	11.4	12.5	$26.9_{\pm 9.4}$	$15.1_{\pm 7.9}$	$10.5_{\pm 4.3}$
+ Pre-train	9.3	11.7	11.2	17.1	$15.1_{\pm 4.4}$	$\underline{8.5}_{\pm 3.7}$	$\underline{5.4}_{\pm 1.7}$
w/ LAPT	<u>6.4</u>	<u>4.3</u>	<u>6.1</u>	<u>8.3</u>	$16.5_{\pm 4.6}$	$10.6_{\pm 3.8}$	$7.3_{\pm 3.1}$
DMoE	8.9	10.0	11.8	17.2	$\mathbf{14.4}_{\pm 4.2}$	$\mathbf{7.7}_{\pm 3.4}$	$\mathbf{4.8}_{\pm 1.6}$
w/ DLA	**6.2**	**4.0**	**5.4**	**8.2**	$\underline{15.0}_{\pm 4.2}$	$\underline{8.5}_{\pm 3.2}$	$\underline{5.4}_{\pm 1.8}$

Table 3: The perplexity after adding new languages.

Ablation study We first modify the hyper-parameter ϵ to determine the effect of augmenting the layer with higher parameter deviation. Figure 5 shows that scaling up layers with higher deriva-tion is much better than the random augmentation baseline when ϵ is less than 0.5. To balance the parameter amount and performance, we set ϵ to 0.4 in this work.

The router classification loss is removed to quantify its contribution. As shown in the "w.o/ Class. Loss" row of Table 2, the perplexity increases by 0.5 on average. Figure 6 illustrates the statistics of token distribution assigned to the top-1 expert. It can be found that the bottom layer like the first layer does not show language specification without router classification loss (Figure 6(a)). Tokens are mostly assigned to the expert tuned in the same language with router classification loss as expected (Figure 6(b)).

4.3 Extend to 128 Languages

In this section, we further scale up the number of languages from 18 to 128 and increase the amount of pre-training tokens to 17.7B. Following previous findings, the number of language groups is set to 16, and refer to Table 8 in Appendix C.1 for more details of language dividing.

Table 4 reports the perplexity of 20 languages across different resources and the average result of 128 languages. It can be found that DMoE significantly mitigates the curse of multilinguality phenomenon and outperforms Branch-Train-Mix 1.1 PPL on average across 128 languages. The improvement mostly comes from unseen languages and low-resource languages, which reach 2.1 PPL on average for the five extremely low-resource languages in Table 4. The eighty languages with Latin script improved by 2.9 PPL over the continual pre-training model on average, while the other non-Latin languages improved by only 0.7 PPL.

We calculate the improvement across language families and find that the trend is similar where our method outperforms the baseline methods. It

Model	High					Medium					Low					Extremely-Low					ALL 128L
	ar	de†	en	it†	ja†	hi	id	th†	uk†	vi†	kk†	mn†	my†	te	ur	br†	pa†	sw	ug†	zu	Avg ↓
BLOOM$_{560M}$	42.4	111.3	66.8	82.4	55.5	30.6	41.3	13.7	44.5	21.8	29.0	31.2	6.1	91.4	72.5	261.7	131.6	224.9	44.5	1278.9	154.4$_{\pm 157.0}$
+ Pre-train	34.4	23.8	44.6	20.5	12.9	20.5	26.9	3.6	7.3	15.1	5.7	6.8	2.8	30.3	37.6	40.2	32.5	45.5	9.2	36.2	20.7$_{\pm 12.8}$
Branch-Train-Mix	32.5	21.0	**40.3**	17.1	12.1	20.0	26.1	3.5	6.6	**14.6**	5.4	6.3	2.8	31.6	37.1	35.4	34.1	43.5	9.0	31.2	19.2$_{\pm 12.2}$
DMoE	**32.1**	**19.5**	40.8	**16.3**	**11.3**	**19.8**	**25.8**	**3.4**	**6.4**	14.7	**5.2**	**6.2**	**2.7**	**29.3**	**35.2**	**31.4**	**30.5**	**39.7**	**8.1**	**28.3**	**17.7**$_{\pm 11.2}$
BLOOM$_{1.7B}$	30.4	56.4	45.3	44.9	35.0	20.8	27.8	9.7	26.7	15.3	19.1	21.1	4.4	46.8	44.0	113.2	61.7	80.3	28.4	260.8	71.9$_{\pm 52.1}$
+ Pre-train	22.9	15.9	30.3	14.1	9.7	14.7	19.3	3.1	5.4	11.3	4.5	5.4	**2.5**	22.3	26.8	27.9	23.9	30.8	7.4	26.0	14.9$_{\pm 8.7}$
Branch-Train-Mix	**21.1**	15.3	**29.7**	13.1	9.4	**14.1**	**18.6**	3.1	5.3	**10.9**	4.5	5.4	2.6	22.4	**25.7**	26.2	24.2	29.8	7.5	24.1	14.3$_{\pm 8.4}$
DMoE	22.7	**14.9**	31.3	**12.8**	**8.9**	14.7	19.1	**3.0**	**5.1**	11.3	**4.3**	**5.1**	**2.5**	**22.1**	26.0	**23.9**	**23.2**	**29.2**	**6.8**	**22.8**	**13.7**$_{\pm 8.1}$

Table 4: The perplexity of 20 languages on the valid split of MADLAD-400 (Kudugunta et al., 2023). Refer to Table 10 to 17 in Appendix C.4 for all results of 128 languages. † denotes the language unseen in the pre-training of BLOOM. "**High**"(>1%), "**Medium**"(>0.1%), "**Low**"(>0.01%), and "**Extremely-Low**"(<=0.01%) indicates the available amount of linguistic resources on the CommonCrawl following Lai et al. (2023).

Model	Zero-shot Results					Few-shot Results				
	XNLI	PAWS-X	XCOPA	XStoryCloze	XWinograd	XNLI	PAWS-X	XCOPA	XStoryCloze	XWinograd
BLOOM$_{560M}$	36.2$_{\pm 3.3}$	51.5$_{\pm 1.6}$	53.9$_{\pm 4.1}$	53.5$_{\pm 3.5}$	53.7$_{\pm 4.0}$	34.4$_{\pm 2.4}$	51.1$_{\pm 1.2}$	53.4$_{\pm 4.0}$	52.6$_{\pm 3.5}$	53.3$_{\pm 4.2}$
+ Pre-train	37.1$_{\pm 3.5}$	52.9$_{\pm 2.4}$	53.6$_{\pm 2.9}$	53.8$_{\pm 2.6}$	54.9$_{\pm 4.1}$	34.7$_{\pm 2.6}$	51.6$_{\pm 1.1}$	53.8$_{\pm 2.4}$	52.7$_{\pm 2.8}$	53.8$_{\pm 5.1}$
Branch-Train-Mix	37.2$_{\pm 4.1}$	53.1$_{\pm 2.5}$	54.1$_{\pm 2.7}$	53.8$_{\pm 2.8}$	54.4$_{\pm 3.6}$	35.3$_{\pm 2.6}$	51.4$_{\pm 2.1}$	53.9$_{\pm 3.2}$	53.1$_{\pm 2.8}$	54.2$_{\pm 4.3}$
DMoE	**37.5**$_{\pm 4.5}$	**53.2**$_{\pm 1.7}$	**54.4**$_{\pm 2.8}$	**54.1**$_{\pm 2.7}$	**55.1**$_{\pm 4.3}$	**35.7**$_{\pm 2.5}$	**52.2**$_{\pm 1.4}$	**54.7**$_{\pm 2.6}$	**53.4**$_{\pm 2.7}$	**55.1**$_{\pm 4.6}$
BLOOM$_{1.7B}$	39.2$_{\pm 5.4}$	53.9$_{\pm 1.6}$	55.1$_{\pm 5.7}$	56.0$_{\pm 4.7}$	55.1$_{\pm 5.2}$	37.1$_{\pm 4.5}$	50.5$_{\pm 1.1}$	55.2$_{\pm 5.8}$	55.5$_{\pm 5.2}$	55.5$_{\pm 5.2}$
+ Pre-train	39.2$_{\pm 4.9}$	53.7$_{\pm 1.5}$	55.0$_{\pm 3.9}$	56.4$_{\pm 3.6}$	55.5$_{\pm 4.6}$	37.1$_{\pm 3.2}$	51.8$_{\pm 1.7}$	55.3$_{\pm 4.5}$	56.0$_{\pm 3.6}$	55.8$_{\pm 4.6}$
Branch-Train-Mix	39.1$_{\pm 5.1}$	53.5$_{\pm 1.6}$	55.6$_{\pm 3.6}$	56.4$_{\pm 3.3}$	55.6$_{\pm 4.7}$	36.8$_{\pm 3.2}$	51.3$_{\pm 1.4}$	55.5$_{\pm 4.3}$	**56.2**$_{\pm 4.0}$	55.8$_{\pm 4.4}$
DMoE	**39.8**$_{\pm 4.8}$	**54.1**$_{\pm 1.2}$	**56.0**$_{\pm 4.1}$	**56.6**$_{\pm 3.5}$	**56.4**$_{\pm 5.1}$	**37.5**$_{\pm 2.9}$	**52.2**$_{\pm 2.4}$	**55.7**$_{\pm 3.8}$	56.1$_{\pm 3.4}$	**56.4**$_{\pm 5.1}$

Table 5: The in-context learning results of models after training on 128 languages. The number of demonstration samples in the "Few-shot" setting is set to four in this work. Table 18 to 22 report results of all languages.

is interesting to find that languages belonging to the Niger-Congo family, which only take up 0.4GB in the pre-training corpus of BLOOM, benefit the most from our method (Figure 7).

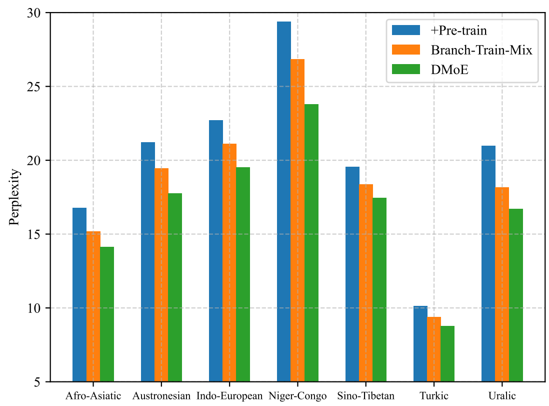


Figure 7: The average perplexity of BLOOM$_{560M}$ across language families after training on the 128 languages.

In-context learning results on five multilingual datasets are shown in Table 5. Appendix C.4 reports the results of all languages on these tasks. Similar to the language modeling results, DMoE also boosts the in-context learning performance and outperforms baseline models across two parameter amounts under zero-shot and few-shot settings. The performance on the multilingual reasoning task XWinograd benifits most from our method, which improves 1.6% on average over the base model. It further demonstrates the effectiveness of our method in improving multilingual large language models.

5 Conclusion and Future Work

In this paper, we propose a method to mitigate the curse of multilinguality by augmenting parameters and boosting cross-lingual transfer. Multilingual large language models trained with our method achieve better language modeling and in-context learning performance than the continued pre-trained dense model and other scaling methods. These language-specialized experts make it easier to learn new languages and keep multilingual knowledge learned.

The specialization of experts can be further improved in the future, e.g., a shared expert learning general knowledge and other experts specializing in language-related knowledge. Designing a method to determine the language similarity with less cost or calculate better language clustering results is another direction. We hope this work can motivate more studies on the curse of multilinguality phenomenon and put forward the development of multilingual language models.

Limitations

The first limitation lies in the additional computation to fine-tune and determine the parameter derivation for each language, which will linearly increase with the number of languages involved and the parameter amount of the model. Transferring the language similarity calculated from the small model into the larger model is a promising method to save computation.

The coverage of training and evaluation languages is another limitation. For example, languages from the Trans-New Guinea language family are not involved in this work. It is mainly due to the constrain of languages provided by the multilingual corpora and datasets used.

Our method relies on dynamic grouping languages and scaling parameters, which brings a higher training cost than the dense model. Due to the limited computation budget, the parameter amount of LLMs investigated in this work is less than 22.6B, and the token amount in training is restrained at 17.7B.

Acknowledgements

We would like to thank Junhong Wu and the anonymous reviewers for their helpful discussions and valuable comments. The research work was supported by the Natural Science Foundation of China (No. 62336008) and the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDA04080400).

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, and et al. 2023. Palm 2 technical report. *Preprint*, arXiv:2305.10403.

Akari Asai, Sneha Kudugunta, Xinyan Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov, Sebastian Ruder, and Hannaneh Hajishirzi. 2024. BUFFET: Benchmarking large language models for few-shot cross-lingual transfer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1771–1800, Mexico City, Mexico. Association for Computational Linguistics.

Terra Blevins, Tomasz Limisiewicz, Suchin Gururangan, Margaret Li, Hila Gonen, Noah A. Smith, and Luke Zettlemoyer. 2024. Breaking the curse of multilinguality with cross-lingual expert language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10822–10837, Miami, Florida, USA. Association for Computational Linguistics.

Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Ben Bergen. 2024. When is multilinguality a curse? language modeling for 250 high- and low-resource languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4074–4096, Miami, Florida, USA. Association for Computational Linguistics.

Ronan Collobert, Samy Bengio, and Yoshua Bengio. 2001. A parallel mixture of svms for very large scale problems. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

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

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. 2024. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *Preprint*, arXiv:2401.06066.

Damai Dai, Li Dong, Shuming Ma, Bo Zheng, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. StableMoE: Stable routing strategy for mixture of experts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7085–7095, Dublin, Ireland. Association for Computational Linguistics.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, Acyr Locatelli, Hangyu Lin, Dwarak Talupuru, Bharat Venkitesh, David Cairuz, Bowen Yang, Tim Chung, Wei-Yin Ko, Sylvie Shang Shi, Amir Shukayev, Sammie Bae, Aleksandra Piktus, Roman Castagné, Felipe Cruz-Salinas, Eddie Kim, Lucas Crawhall-Stein, Adrien Morisot, Sudip Roy, Phil Blunsom, Ivan Zhang, Aidan Gomez, Nick Frosst, Marzieh Fadaee, Beyza Ermis, Ahmet Üstün, and Sara Hooker. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *Preprint*, arXiv:2412.04261.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

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

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.

Shaoxiong Ji, Zihao Li, Indraneil Paul, Jaakko Paavola, Peiqin Lin, Pinzhen Chen, Dayyán O'Brien, Hengyu Luo, Hinrich Schütze, Jörg Tiedemann, and Barry Haddow. 2025. Emma-500: Enhancing massively multilingual adaptation of large language models. *Preprint*, arXiv:2409.17892.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Michael I Jordan and Robert A Jacobs. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214.

Sneha Kudugunta, Isaac Rayburn Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. MADLAD-400: A multilingual and document-level large audited dataset. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

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

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. {GS}hard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chong Li, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. 2024. Improving in-context learning of multilingual generative language models with cross-lingual alignment. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8058–8076, Mexico City, Mexico. Association for Computational Linguistics.

Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2022. Branch-train-merge: Embarrassingly parallel training of expert language models. *Preprint*, arXiv:2208.03306.

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

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Chaitanya Malaviya, Graham Neubig, and Patrick Littell. 2017. Learning language representations for typology prediction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2529–2535, Copenhagen, Denmark. Association for Computational Linguistics.

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

Lingfeng Ming, Bo Zeng, Chenyang Lyu, Tianqi Shi, Yu Zhao, Xue Yang, Yefeng Liu, Yiyu Wang, Linlong Xu, Yangyang Liu, Xiaohu Zhao, Hao Wang, Heng Liu, Hao Zhou, Huifeng Yin, Zifu Shang, Haijun Li, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2024. Marco-llm: Bridging languages via massive multilingual training for cross-lingual enhancement. *Preprint*, arXiv:2412.04003.

MistralAI. 2023. Mixtral of experts. *Mistral AI news*.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.

Robert Östling and Jörg Tiedemann. 2017. Continuous multilinguality with language vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 644–649, Valencia, Spain. Association for Computational Linguistics.

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.

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

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *OpenAI blog*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

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

Stephen Roller, Sainbayar Sukhbaatar, Arthur Szlam, and Jason E Weston. 2021. Hash layers for large sparse models. In *Advances in Neural Information Processing Systems*.

BigScience Workshop: Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, and Alexandra Sasha Luccioni et al. 2023. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Babak Shahbaba and Radford Neal. 2009. Nonlinear models using dirichlet process mixtures. *Journal of Machine Learning Research*, 10(8).

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Sainbayar Sukhbaatar, Olga Golovneva, Vasu Sharma, Hu Xu, Xi Victoria Lin, Baptiste Roziere, Jacob Kahn, Shang-Wen Li, Wen tau Yih, Jason E Weston, and Xian Li. 2024. Branch-train-mix: Mixing expert LLMs into a mixture-of-experts LLM. In *First Conference on Language Modeling*.

Jingyuan Sun, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. 2020. Distill and replay for continual language learning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3569–3579, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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

Lucas Theis and Matthias Bethge. 2015. Generative image modeling using spatial lstms. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Alexey Tikhonov and Max Ryabinin. 2021. It's All in the Heads: Using Attention Heads as a Baseline for Cross-Lingual Transfer in Commonsense Reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3534–3546, Online. Association for Computational Linguistics.

Volker Tresp. 2000. Mixtures of gaussian processes. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press.

Yulia Tsvetkov, Sunayana Sitaram, Manaal Faruqui, Guillaume Lample, Patrick Littell, David Mortensen, Alan W Black, Lori Levin, and Chris Dyer. 2016. Polyglot neural language models: A case study in cross-lingual phonetic representation learning. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1357–1366, San Diego, California. Association for Computational Linguistics.

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

Qian Wang and Jiajun Zhang. 2022. Parameter differentiation based multilingual neural machine translation. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 11440–11448. AAAI Press.

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

Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Bowen Yu, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. 2023b. Polylm: An open source polyglot large language model. *Preprint*, arXiv:2307.06018.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Genta Winata, Lingjue Xie, Karthik Radhakrishnan, Shijie Wu, Xisen Jin, Pengxiang Cheng, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2023. Overcoming catastrophic forgetting in massively multilingual continual learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 768–777, Toronto, Canada. Association for Computational Linguistics.

Junhong Wu, Yuchen Liu, and Chengqing Zong. 2024. F-MALLOC: Feed-forward memory allocation for continual learning in neural machine translation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7180–7192, Mexico City, Mexico. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

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

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Yang Zhao, Junnan Zhu, Lu Xiang, Jiajun Zhang, Yu Zhou, Feifei Zhai, and Chengqing Zong. 2022. Life-long learning for multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:2212.02800*.

Hao Zhou, Zhijun Wang, Shujian Huang, Xin Huang, Xue Han, Junlan Feng, Chao Deng, Weihua Luo, and Jiajun Chen. 2025. Moe-lpr: Multilingual extension of large language models through mixture-of-experts with language priors routing. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 26092–26100. AAAI Press.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, zhifeng Chen, Quoc V Le, and James Laudon. 2022. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, volume 35, pages 7103–7114. Curran Associates, Inc.

A Hyper-parameters

Following Scao et al. (2023), the global batch size is set to 512 samples with 2048 tokens during the language adaptation pre-training stage. AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ is used in this work. We empirically set the learning rate to 2e-5, adopt bf16 mixed precision training (Micikevicius et al., 2018) and ZeRO-3 (Rasley et al., 2020) to save GPU memory cost. And the α in the loss function (Equation 4) is empirically set to 1.28. The model learns language-specialized experts and routers in the dynamic MoE layer extension stage, and is trained normally after that. All MoE models adopt top-2 routing during inference in this work.

B Language Delta

Figure 8 illustrates the cosine similarity of the parameter deviation during fine-tuning. It can be found that the deviation is relatively small after 10 tuning steps, and the cosine similarity of the one at the 10th step between the parameter deviation at the 40th step is higher than 80% for all languages. The language similarity matrices are similar using the parameter deviation at different steps (Figure 9). Therefore, we only fine-tune 10 steps to determine the parameter derivation of models for each language.

Figure 10 shows the distribution of parameter deviation $\|\Delta\theta^x\|$ across layers of BLOOM_{560M} for 18 languages. It is interesting to find that layers near the embedding or output layer often have a relatively high parameter derivation $\|\Delta\theta^x\|$.

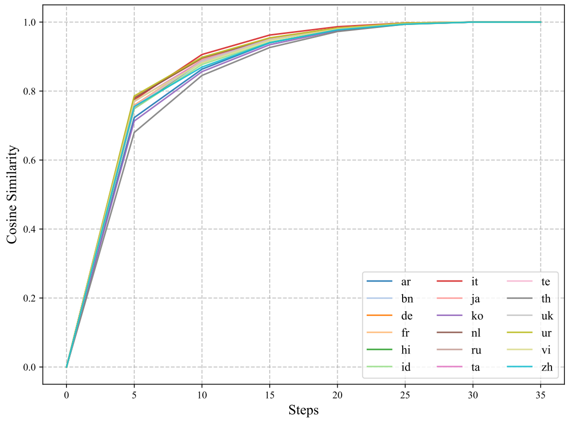


Figure 8: The cosine similarity between $\Delta\theta^x$ at the i-th step and the one at the 40th step for each language using BLOOM_{560M}.

C Additional Results

C.1 Language Similarity and Grouping

Figure 11 and 12 illustrates the language similarity matrix calculated by all layers and the last 3 layers of the parameter derivation of $\text{BLOOM}_{1.7B}$. Except for the difference in absolute value, they are very similar in relative trend. Therefore, we adopt the last 3 layers of the parameter derivation to calculate the language similarity by default.

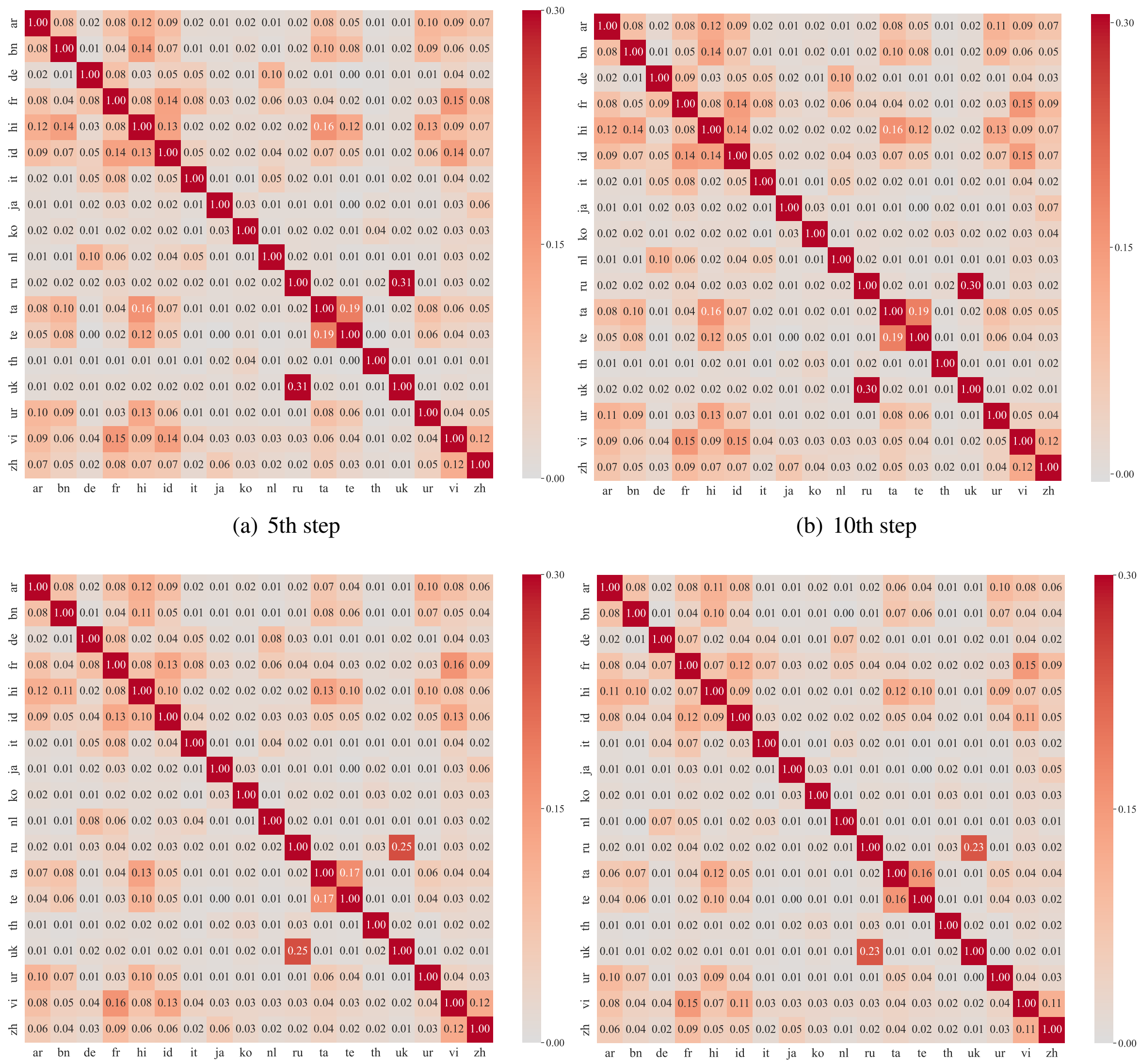
Figure 13 shows pair-wise language similarity matrices of Gemma_{2B}. It can be found that the one of $\text{BLOOM}_{1.7B}$ is similar to the one of BLOOM_{560M} (Figure 3), which may arise from the same pre-training corpus used. In contrast, the language similarity matrix of Gemma_{2B} has a higher average similarity value and different patterns between languages. As shown in Figure 14, It is interesting to find that replacing the multilingual corpus from CulturaX to MADLAD-400 results in a similar matrix.

Given the language similarity matrix calculated, we obtain the language grouping results for $\text{BLOOM}_{1.7B}$ and Gemma_{2B} in the 18-language experiments using Algorithm 1 (Table 6 and 7). Similar languages like Tamil and Telugu are often grouped in the same language cluster. BLOOM_{560M} and $\text{BLOOM}_{1.7B}$ have the same language clustering result under the six and two groups settings.

	ar	ur	bn	it	de	nl	ta	te	hi	id	fr	vi	ru	uk	th	ko	ja	zh
9G	1	2	3	4	5	5	6	6	3	1	7	7	8	8	4	2	9	9
6G	1	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6	6
3G	1	2	2	3	3	3	2	2	1	1	1	1	2	3	3	2	3	1
2G	1	1	1	2	2	2	1	2	1	1	1	1	2	2	2	2	2	1

Table 6: The grouping results of $\text{BLOOM}_{1.7B}$, where "**2G**" denotes the result that divides into 2 groups.

Table 8 reports the 16 language groups used in the 128 languages experiment, which is calcu-



(a) 5th step (b) 10th step (c) 20th step (d) 40th step

Figure 9: The language similarity matrices calculated by the parameter derivation at different fine-tuning steps using $\text{BLOOM}_{560\text{M}}$.

	ar	ur	bn	it	de	nl	ta	te	hi	id	fr	vi	ru	uk	th	ko	ja	zh
9G	1	1	2	6	3	3	4	4	2	5	6	5	7	7	8	8	9	9
6G	1	1	2	3	4	4	2	2	1	3	4	5	3	5	5	6	6	6
3G	1	1	1	2	2	2	1	1	1	2	2	3	2	3	3	3	3	3
2G	1	1	1	2	2	2	1	1	1	2	2	2	2	1	1	1	2	1

Table 7: The grouping results of $\text{Gemma}_{2\text{B}}$, where "**2G**" denotes the result that divides into 2 groups.

lated by the parameter deviation of $\text{BLOOM}_{1.7\text{B}}$. $\text{BLOOM}_{560\text{M}}$ adopts this result to save computation for the similar trend with $\text{BLOOM}_{1.7\text{B}}$ in the 18-language experiment.

C.2 More base models on 18 Languages

We apply our method to Qwen2.5 base models and report the results in Table 9, which are in line with the previous results in Table 2.

C.3 Token Router Distribution

We statisticize the top-1 expert distribution across the mixture-of-experts layers in Figure 15 and 16. As shown in Figure 15, the language specialization emerges at the last five MoE layers, while MoE layers often show language specialization with router language classification loss (Figure 16).

C.4 Extend to 128 Languages

We report the perplexity of all 128 languages on Table 10 to 16. And in-context learning results on five multilingual datasets are shown in Table 18 to 22.

Index	Languages
1	ceb, en, fil, hil, ilo, la, lg, so
2	fr, it, ny, sn, sw, xh, yo, zu
3	am, dv, he, ka, ko, lo, my, ti
4	ar, ca, es, eu, hi, id, pt, vi
5	cnh, cs, de, ha, kha, lus, nl, uz
6	kaa, kk, ky, mn, ru, sah, tt, tyv
7	da, et, fi, fo, gsw, is, no, se
8	av, be, ce, mk, sr, tg, udm, uk

Index	Languages
9	el, grc, hu, os, pl, ro, tr, yi
10	bn, gu, kn, ml, mr, pa, ta, te
11	az, br, ckb, fa, ps, sd, ug, ur
12	ht, ig, jv, mg, ms, sl, su, vec
13	fy, haw, lv, mi, sm, st, tet, to
14	co, eo, gl, ja, ne, oc, yue, zh
15	ee, gd, hmn, lb, mt, om, rm, ts
16	bo, kbd, kl, km, pap, sa, th, tk

Table 8: The 16 language groups divided for the 128 languages experiment.

		High								Medium							Low			
Model	**#Param.**	**ar**	**de**	**fr**	**it**	**ja**	**nl**	**ru**	**zh**	**bn**	**hi**	**id**	**ko**	**th**	**uk**	**vi**	**ta**	**te**	**ur**	**Avg ↓**
Qwen2.5$_{0.5B}$	494M	110.8	147.3	9.5	133.8	19.8	20.2	3.7	36.8	336.2	191.4	85.2	2.5	3.4	5.4	23.3	44.7	$2.1e^5$	689.1	1262.0
+ Pre-train	494M	75.7	89.2	8.3	75.3	15.9	10.9	**3.4**	32.4	40.6	32.8	56.9	2.4	2.8	3.2	18.7	9.0	192.5	48.8	39.9
X-ELM	4.44B	74.3	87.9	8.2	72.9	**15.3**	10.7	**3.4**	**30.3**	51.9	40.2	55.5	**2.3**	2.8	3.2	18.2	11.9	404.1	57.3	52.8
DMoE (9 Groups)	1.75B	**68.8**	**81.1**	**8.0**	**67.9**	**15.3**	**9.8**	**3.4**	31.6	**33.6**	**27.8**	**51.8**	**2.3**	**2.7**	**3.0**	**17.7**	**8.6**	**173.0**	**38.7**	**35.8**
Qwen$_{1.5B}$	1.54B	59.4	70.8	7.0	64.4	13.2	12.2	3.0	25.1	136.2	86.4	45.0	2.2	2.8	3.9	15.0	29.6	4809.7	246.9	312.9
+ Pre-train	1.54B	48.6	52.7	**6.6**	47.1	11.9	8.3	3.0	22.5	26.9	22.6	38.0	**2.1**	**2.5**	2.8	13.6	7.4	121.0	32.7	26.1
X-ELM	13.89B	**47.4**	52.1	**6.6**	**45.0**	**11.5**	**8.1**	**2.9**	**21.6**	31.3	25.9	**36.4**	**2.1**	**2.5**	2.8	**13.2**	8.7	174.2	35.6	29.3
DMoE (9 Groups)	3.19B	47.9	**51.7**	**6.6**	45.9	11.8	**8.1**	3.0	22.4	**25.6**	**21.6**	37.4	**2.1**	**2.5**	**2.7**	13.5	**7.2**	**103.4**	**30.4**	**24.7**

Table 9: The normalized perplexity on the valid split of CulturaX. The perplexity is normalized to the vocabulary of Bloom following Wei et al. (2023a). "**High**", "**Medium**", and "**Low**" indicates the available amount of linguistic resources. The best and second results are denoted in **bold** and underlined, correspondingly.

Model	**am**	**ar**	**av**	**az**	**be**	**bn**	**bo**	**br**	**ca**	**ce**	**ceb**	**ckb**	**cnh**	**co**	**cs**	**da**
BLOOM$_{560M}$	13.84	42.39	48.53	187.51	65.85	45.23	7.17	261.68	41.64	51.08	225.93	35.40	445.37	197.32	146.79	84.96
+ Pre-train	4.39	34.43	8.27	11.68	8.63	33.11	3.44	40.25	24.85	11.25	20.18	8.97	32.45	32.27	17.03	10.04
Branch-Train-Mix	4.12	32.53	7.53	10.85	7.61	32.62	3.47	35.44	23.59	9.62	18.37	8.46	28.38	28.55	15.43	8.67
DMoE	4.08	32.06	7.18	9.82	7.38	31.69	3.39	31.42	23.61	9.39	16.57	8.07	27.59	25.97	14.30	7.98
BLOOM$_{1.7B}$	8.60	30.41	32.15	72.69	36.83	28.67	5.58	113.16	26.55	34.38	93.94	24.93	188.45	103.17	73.24	42.47
+ Pre-train	3.74	22.93	6.40	8.68	6.40	22.04	3.15	27.93	18.00	8.30	15.50	7.51	24.43	22.78	12.06	6.97
Branch-Train-Mix	3.71	21.10	6.25	8.43	6.16	20.82	3.27	26.20	17.12	7.96	14.83	7.22	23.23	22.41	11.82	6.51
DMoE	3.62	22.66	5.94	7.88	5.94	21.89	3.12	23.89	17.88	7.69	13.80	7.07	21.90	20.34	11.15	6.18

Table 10: The perplexity of language "am" to "da" on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	**de**	**dv**	**ee**	**el**	**en**	**eo**	**es**	**et**	**eu**	**fa**	**fi**	**fil**	**fo**	**fr**	**fy**	**gd**
BLOOM$_{560M}$	111.29	4.96	300.78	18.28	66.79	266.55	39.23	310.09	80.16	161.83	250.93	234.67	253.93	43.25	285.93	172.37
+ Pre-train	23.83	2.10	20.79	4.34	44.58	31.29	24.03	29.32	39.50	14.98	24.25	17.72	24.34	27.21	24.39	15.01
Branch-Train-Mix	21.03	2.11	19.75	4.18	40.29	26.97	23.67	24.47	37.73	14.51	21.26	15.86	20.78	25.53	20.67	12.95
DMoE	19.46	2.05	15.78	4.02	40.82	22.84	23.14	22.67	36.67	13.40	18.91	14.18	19.00	25.97	18.13	11.55
BLOOM$_{1.7B}$	56.36	3.49	121.31	11.27	45.33	126.18	27.31	151.08	40.32	86.79	113.01	98.05	111.79	28.28	143.77	84.11
+ Pre-train	15.89	1.97	15.37	3.53	30.27	21.05	18.15	19.85	25.84	11.96	16.76	13.23	16.88	20.03	16.03	11.38
Branch-Train-Mix	15.34	2.00	14.43	3.57	29.69	20.33	17.43	18.64	24.97	11.35	16.27	12.54	15.78	18.89	14.92	10.30
DMoE	14.86	1.94	13.35	3.43	31.34	17.73	18.19	17.50	25.37	10.89	15.07	11.62	14.72	20.01	13.55	9.85

Table 11: The perplexity of language "de" to "gd" on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	gl	grc	gsw	gu	ha	haw	he	hi	hil	hmn	ht	hu	id	ig	ilo	is
BLOOM$_{560M}$	121.12	19.74	143.45	180.03	559.77	73.99	20.93	30.55	220.05	120.38	391.80	156.40	41.35	262.93	271.81	207.29
+ Pre-train	26.66	6.00	42.16	39.27	27.09	10.39	7.01	20.45	17.53	13.11	33.86	15.41	26.93	28.80	26.04	20.08
Branch-Train-Mix	24.84	5.69	37.22	40.53	24.27	9.67	6.71	19.98	14.53	11.94	30.05	14.10	26.10	26.85	22.45	17.36
DMoE	23.70	5.45	35.14	37.51	22.80	9.00	6.44	19.78	13.22	10.78	26.19	13.00	25.81	23.77	19.76	15.98
BLOOM$_{1.7B}$	61.19	12.99	90.97	67.49	191.64	47.25	15.00	20.82	96.72	79.33	169.97	70.31	27.80	92.00	130.53	92.95
+ Pre-train	18.81	4.71	28.90	27.69	19.49	8.48	5.58	14.70	12.23	10.47	23.94	10.98	19.25	20.90	18.45	14.12
Branch-Train-Mix	18.04	4.74	27.91	27.50	18.94	8.24	5.55	14.14	10.97	9.79	21.31	10.91	18.65	19.81	17.42	13.36
DMoE	17.84	4.60	26.20	27.18	17.73	7.91	5.41	14.68	10.44	9.48	20.24	10.27	19.08	18.63	15.59	12.44

Table 12: The perplexity of language “gl” to “is” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	it	ja	jv	ka	kaa	kbd	kha	kk	kl	km	kn	ko	ky	la	lb	lg
BLOOM$_{560M}$	82.38	55.55	277.50	12.20	50.53	30.80	200.91	28.98	231.11	10.09	196.03	24.03	47.03	111.79	261.41	369.16
+ Pre-train	20.47	12.86	36.26	4.16	6.56	7.68	20.35	5.71	16.50	3.85	46.08	7.36	7.37	36.19	25.71	46.19
Branch-Train-Mix	17.12	12.12	34.82	4.08	6.16	7.12	18.22	5.38	14.78	3.81	47.55	7.12	6.89	34.72	21.35	40.71
DMoE	16.33	11.34	30.43	3.89	5.95	6.35	17.05	5.20	13.57	3.63	43.80	6.74	6.61	32.04	18.92	36.13
BLOOM$_{1.7B}$	44.86	35.01	149.22	10.42	32.12	20.19	124.88	19.14	120.42	8.26	92.29	16.12	31.49	66.75	132.24	141.18
+ Pre-train	14.13	9.67	26.62	3.63	5.12	5.86	14.83	4.52	12.68	3.36	32.93	6.00	5.73	27.12	16.92	33.36
Branch-Train-Mix	13.10	9.39	25.49	3.66	5.14	5.83	14.51	4.50	12.57	3.39	32.53	5.96	5.73	27.40	15.66	31.24
DMoE	12.83	8.92	23.54	3.51	4.86	5.50	13.19	4.31	11.14	3.25	32.03	5.72	5.44	26.05	14.69	28.27

Table 13: The perplexity of language “it” to “lg” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	lo	lus	lv	mg	mi	mk	ml	mn	mr	ms	mt	my	ne	nl	no	ny
BLOOM$_{560M}$	8.72	312.58	169.08	189.81	129.33	62.40	116.36	31.15	127.22	98.74	97.91	6.08	125.47	118.66	213.49	286.37
+ Pre-train	2.44	37.91	15.10	18.63	16.50	9.37	34.71	6.75	36.21	33.69	9.95	2.77	51.30	15.22	22.27	22.99
Branch-Train-Mix	2.42	34.92	12.49	16.38	15.14	8.45	35.88	6.31	36.48	32.98	8.94	2.77	53.62	13.01	19.45	20.69
DMoE	2.37	32.79	10.65	15.08	13.82	8.05	33.29	6.19	34.98	31.17	7.42	2.69	49.34	12.22	17.92	18.37
BLOOM$_{1.7B}$	4.58	159.69	83.05	82.55	74.40	36.15	50.08	21.05	61.00	51.52	59.86	4.41	63.23	58.05	97.54	126.71
+ Pre-train	2.17	28.49	10.26	14.08	13.02	6.96	23.58	5.37	24.57	23.49	7.26	2.52	35.24	10.26	15.45	17.08
Branch-Train-Mix	2.20	28.23	9.05	12.81	12.46	6.74	23.43	5.35	24.15	22.88	6.67	2.57	35.40	9.67	14.52	15.98
DMoE	2.13	26.04	8.45	12.54	11.84	6.51	23.58	5.13	24.06	22.18	6.27	2.48	34.91	9.37	13.86	15.11

Table 14: The perplexity of language “lo” to “ny” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	oc	om	os	pa	pap	pl	ps	pt	rm	ro	ru	sa	sah	sd	se	sl
BLOOM$_{560M}$	80.82	260.13	30.78	131.60	367.89	91.73	67.38	37.27	354.95	195.23	33.77	181.38	32.12	83.51	255.36	218.52
+ Pre-train	20.72	27.23	8.08	32.55	28.52	12.84	10.99	23.22	27.98	16.72	8.13	43.64	6.88	13.97	27.01	22.97
Branch-Train-Mix	19.13	23.77	7.56	34.14	25.53	11.91	10.65	22.23	23.60	15.09	7.40	47.25	6.51	14.48	23.05	19.99
DMoE	17.82	21.06	7.06	30.48	22.65	11.12	9.59	22.51	18.69	14.06	7.24	41.47	6.20	12.62	21.32	17.32
BLOOM$_{1.7B}$	45.64	150.00	21.03	61.74	187.88	46.24	44.31	24.85	183.77	87.28	21.70	104.87	22.39	54.03	156.62	107.83
+ Pre-train	14.69	20.86	6.20	23.89	19.32	9.20	8.68	17.07	18.42	11.82	6.16	34.04	5.45	11.45	18.82	15.75
Branch-Train-Mix	14.14	19.52	6.36	24.24	17.96	9.25	8.46	16.35	15.92	11.42	5.93	36.76	5.52	11.82	17.97	14.49
DMoE	13.36	17.80	5.68	23.17	16.37	8.66	7.95	17.17	14.33	10.93	5.82	32.97	5.18	10.71	16.31	13.48

Table 15: The perplexity of language “oc” to “sl” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	sm	sn	so	sr	st	su	sw	ta	te	tet	tg	th	ti	tk	to	tr
BLOOM$_{560M}$	114.75	435.89	239.57	51.57	275.07	224.84	224.92	80.03	91.45	206.03	40.17	13.71	16.22	217.70	82.62	153.97
+ Pre-train	15.29	28.67	18.92	9.57	18.37	27.30	45.46	37.65	30.32	16.00	7.22	3.64	5.13	14.84	14.60	14.62
Branch-Train-Mix	13.78	25.29	16.27	8.58	16.60	25.26	43.52	38.29	31.56	14.34	6.62	3.49	4.83	13.70	12.54	13.49
DMoE	12.63	22.63	14.53	8.07	14.96	22.70	39.65	36.32	29.26	12.48	6.36	3.43	4.70	12.08	11.60	12.80
BLOOM$_{1.7B}$	75.21	143.65	131.67	31.61	115.55	129.35	80.34	47.14	46.77	100.13	25.36	9.70	10.17	87.48	49.50	61.45
+ Pre-train	12.08	20.74	13.86	7.08	13.90	20.01	30.75	26.90	22.30	11.61	5.56	3.12	4.31	10.63	11.29	10.42
Branch-Train-Mix	11.27	19.18	12.80	6.88	13.11	18.71	29.84	26.40	22.36	10.45	5.49	3.09	4.31	10.80	10.27	10.20
DMoE	10.91	18.48	11.95	6.52	12.52	17.68	29.23	26.61	22.14	9.76	5.24	3.03	4.11	9.39	9.81	9.75

Table 16: The perplexity of language “sm” to “tr” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

Model	ts	tt	tyv	udm	ug	uk	ur	uz	vec	vi	xh	yi	yo	yue	zh	zu
BLOOM$_{560M}$	175.78	39.70	40.14	54.51	44.53	44.49	72.47	388.73	319.37	21.85	606.11	17.27	257.05	71.22	36.26	1278.88
+ Pre-train	13.05	7.46	8.37	8.87	9.21	7.30	37.61	18.66	73.68	15.12	34.44	5.10	28.08	25.72	15.07	36.17
Branch-Train-Mix	11.26	6.97	7.54	7.95	8.99	6.61	37.13	16.74	64.42	14.62	32.00	4.86	27.41	25.82	14.83	31.24
DMoE	9.90	6.71	7.33	7.65	8.10	6.40	35.19	15.80	55.43	14.66	27.73	4.71	24.28	23.99	14.31	28.31
BLOOM$_{1.7B}$	85.50	27.57	28.15	39.09	28.37	26.68	44.00	144.46	188.36	15.30	154.98	11.37	100.09	42.71	25.86	260.80
+ Pre-train	9.51	5.77	6.38	6.75	7.35	5.44	26.84	13.32	46.42	11.34	25.61	4.13	21.09	19.02	11.70	26.00
Branch-Train-Mix	8.67	5.74	6.20	6.54	7.47	5.28	25.68	13.08	43.24	10.88	24.70	4.17	21.20	17.44	11.33	24.14
DMoE	8.10	5.45	5.86	6.30	6.81	5.15	25.96	12.12	38.54	11.27	22.65	3.96	19.29	18.25	11.30	22.82

Table 17: The perplexity of language “ts” to “zu” on the valid split of MADLAD-400 (Kudugunta et al., 2023).

		High							Medium						Low			
Model	**#shot**	en	de†	es	eu	fr	ru†	zh	ar	bg†	el†	th†	tr†	vi	hi	sw	ur	**Avg**
BLOOM$_{560M}$	0	43.9	34.4	40.5	37.9	39.1	34.6	35.5	33.5	34.0	35.2	32.1	31.6	39.0	39.8	33.9	34.5	36.2
	4	40.3	34.0	38.6	35.0	37.0	34.4	32.4	33.3	33.4	31.8	33.5	32.1	36.0	33.7	31.8	34.1	34.4
BLOOM$_{560M}$ + Pre-train	0	43.5	37.6	41.6	40.6	40.9	36.1	33.4	33.5	34.1	33.1	33.8	33.4	42.0	38.5	33.9	37.1	37.1
	4	40.3	33.9	38.5	36.7	38.4	32.7	35.5	33.4	31.6	33.4	31.8	32.8	36.3	35.3	32.2	32.1	34.7
Branch-Train-Mix	0	47.4	36.0	41.9	41.2	41.1	36.7	33.8	33.5	34.1	34.9	32.6	34.1	41.7	35.5	33.5	36.7	37.2
	4	41.6	35.8	39.8	35.0	38.6	36.1	34.9	32.7	34.2	33.1	34.3	33.1	35.8	33.9	32.0	33.9	35.3
DMoE	0	48.3	36.3	43.8	38.6	42.9	37.3	33.4	33.3	34.5	34.1	34.0	33.9	43.0	37.7	33.2	36.2	**37.5**
	4	41.2	35.2	40.5	35.2	39.0	35.8	35.2	33.9	35.4	33.9	33.3	33.6	37.7	35.1	32.1	34.0	**35.7**
BLOOM$_{1.7B}$	0	49.2	36.6	47.7	47.0	45.2	37.8	34.9	33.3	35.6	33.6	33.7	35.3	42.9	42.2	34.1	38.4	39.2
	4	46.3	34.7	43.1	40.9	45.0	35.3	38.0	32.9	35.0	33.1	32.9	31.2	37.9	38.4	32.6	35.5	37.1
BLOOM$_{1.7B}$ + Pre-train	0	49.4	40.8	46.7	42.6	43.5	42.4	33.7	33.5	35.1	34.3	37.4	32.7	42.6	38.6	34.9	38.6	39.2
	4	45.0	37.3	40.4	37.8	42.4	38.8	35.4	34.0	35.2	33.7	34.3	33.4	37.5	36.8	34.4	36.6	37.1
Branch-Train-Mix	0	49.7	43.4	46.0	41.4	44.1	41.4	33.8	33.7	34.5	35.8	35.4	31.8	44.5	38.0	35.5	37.1	39.1
	4	44.7	38.4	41.2	37.3	41.5	38.2	35.3	34.4	34.9	33.6	33.8	33.6	37.8	35.9	34.3	34.1	36.8
DMoE	0	50.6	42.6	45.4	44.3	43.1	41.3	34.6	33.5	35.2	37.1	35.1	34.3	44.7	39.2	36.2	39.7	**39.8**
	4	44.9	36.9	40.6	38.2	41.8	39.2	37.4	33.9	36.0	36.2	35.9	32.8	37.7	36.8	34.5	36.7	**37.5**

Table 18: In-context learning results on XNLI across all languages. “**High**”, “**Medium**” and “**Low**” denotes the available amount of linguistic resources. † denotes the unseen language in the pre-training corpus of BLOOM.

Model	#shot	High de†	en	es	fr	ja†	zh	Medium ko†	Avg
$BLOOM_{560M}$	0	49.4	49.9	50.4	52.8	52.8	54.1	51.0	51.5
	4	52.5	50.0	49.7	51.9	51.3	52.5	49.7	51.1
$BLOOM_{560M}$ + Pre-train	0	49.7	50.2	50.7	54.8	55.7	55.1	54.1	52.9
	4	50.9	51.5	51.1	51.9	50.3	54.1	51.3	51.6
Branch-Train-Mix	0	51.3	49.1	50.9	54.6	56.2	55.3	54.7	53.1
	4	48.3	50.5	53.0	52.3	50.1	55.1	50.4	51.4
DMoE	0	51.4	51.1	51.2	54.6	54.9	55.2	54.0	**53.2**
	4	51.3	50.1	52.4	52.8	51.8	54.8	52.2	**52.2**
$BLOOM_{1.7B}$	0	52.6	53.8	50.7	54.9	55.7	54.8	54.8	53.9
	4	48.9	50.4	49.8	51.4	50.1	50.4	52.8	50.5
$BLOOM_{1.7B}$ + Pre-train	0	53.3	51.1	52.7	54.9	55.9	54.9	53.5	53.7
	4	51.4	48.6	51.6	53.2	54.0	53.2	50.8	51.8
Branch-Train-Mix	0	51.3	53.1	52.8	54.8	55.8	55.3	51.7	53.5
	4	50.0	49.4	52.2	53.2	50.3	52.9	51.2	51.3
DMoE	0	53.0	53.7	52.8	54.8	55.9	55.4	53.0	**54.1**
	4	48.8	49.2	50.9	54.6	54.7	54.6	52.5	**52.2**

Table 19: In-context learning results on PAWS-X across all languages. "**High**" and "**Medium**" denotes the available amount of linguistic resources. † denotes the unseen language in the pre-training corpus of BLOOM.

Model	#shot	High zh	Medium id	it†	th†	tr†	vi	Low et†	sw	ta	Ex-Low ht†	qu†	Avg
$BLOOM_{560M}$	0	57.6	60.0	52.4	53.0	52.8	61.0	48.0	52.4	56.4	50.8	49.0	53.9
	4	57.4	60.6	50.2	53.0	50.6	59.2	49.4	50.6	56.4	51.6	48.6	53.4
$BLOOM_{560M}$ + Pre-train	0	53.6	57.2	54.4	53.8	51.6	58.8	52.0	51.4	57.0	49.0	50.4	53.6
	4	54.4	56.8	54.0	53.8	52.8	58.0	52.2	50.8	56.6	52.2	49.8	53.8
Branch-Train-Mix	0	55.8	57.4	54.4	56.2	53.6	58.8	50.2	53.0	54.6	49.8	51.6	54.1
	4	55.0	57.2	53.2	55.4	52.8	61.2	50.8	52.0	55.6	49.6	50.4	53.9
DMoE	0	55.8	59.0	54.2	55.6	53.2	58.6	53.2	51.2	55.6	52.4	49.2	**54.4**
	4	56.6	57.4	53.6	57.0	52.8	59.6	52.6	51.0	56.0	52.4	52.4	**54.7**
$BLOOM_{1.7B}$	0	61.4	63.2	52.4	53.2	53.0	66.2	47.4	51.8	56.4	50.4	50.8	55.1
	4	63.8	62.0	51.2	53.0	52.0	66.2	49.2	52.0	57.0	51.0	50.2	55.2
$BLOOM_{1.7B}$ + Pre-train	0	58.6	61.4	52.6	55.0	52.4	61.8	49.6	54.2	56.0	53.4	50.0	55.0
	4	60.4	61.6	53.0	55.2	51.4	63.4	50.0	54.8	56.2	51.8	50.0	55.3
Branch-Train-Mix	0	58.6	61.2	55.2	55.2	54.2	62.6	51.0	52.6	55.2	54.0	51.4	55.6
	4	59.6	61.4	56.0	53.8	52.2	63.8	50.8	54.0	56.6	50.4	51.8	55.5
DMoE	0	59.6	62.8	54.2	56.2	54.8	63.6	51.2	54.2	56.0	52.6	51.2	**56.0**
	4	60.8	60.2	53.6	56.2	53.8	63.0	51.2	53.4	55.2	54.8	50.6	**55.7**

Table 20: In-context learning results on XCOPA across all languages. "**High**", "**Medium**", "**Low**" and "**Ex-Low**" denotes the available amount of linguistic resources. † denotes the unseen language in the pre-training corpus of BLOOM.

Model	#shot	High				Medium		Low			Ex-Low		
		en	es	ru†	zh	ar	id	hi	sw	te	eu	my†	**Avg**
$BLOOM_{560M}$	0	59.9	55.9	48.4	55.1	52.5	55.3	55.1	49.9	55.1	53.5	47.3	53.5
	4	59.0	54.3	48.6	54.3	49.9	54.9	53.3	49.6	56.5	51.8	46.9	52.6
$BLOOM_{560M}$ + Pre-train	0	59.1	55.1	51.6	54.0	50.4	55.6	54.7	52.9	55.3	54.2	49.4	53.8
	4	57.8	54.3	49.3	53.9	48.5	53.7	53.5	52.5	54.8	52.8	48.4	52.7
Branch-Train-Mix	0	59.2	56.1	51.8	53.6	50.6	55.2	53.8	52.5	55.7	54.9	48.2	53.8
	4	58.5	54.6	51.4	53.5	49.4	54.7	53.6	51.6	55.7	52.9	47.9	53.1
DMoE	0	59.0	56.3	51.4	54.7	51.0	55.7	54.3	52.7	55.8	54.7	49.0	**54.1**
	4	58.6	54.9	50.5	54.3	50.6	54.5	53.8	52.2	55.7	53.9	48.4	**53.4**
$BLOOM_{1.7B}$	0	64.4	61.0	50.3	58.1	54.8	59.9	56.9	52.1	56.6	54.9	47.0	56.0
	4	65.1	61.7	50.0	58.2	53.7	59.0	56.5	51.8	55.4	53.3	45.9	55.5
$BLOOM_{1.7B}$ + Pre-train	0	63.7	60.6	52.4	57.0	54.5	58.0	55.7	54.5	57.1	56.7	49.9	56.4
	4	63.1	60.0	51.7	56.7	54.7	58.3	55.3	55.0	57.0	54.9	49.4	56.0
Branch-Train-Mix	0	62.6	60.2	54.0	57.2	55.1	58.2	56.3	55.0	57.2	55.3	49.2	56.4
	4	64.5	60.1	52.8	56.5	53.9	58.6	55.9	54.9	57.6	55.1	48.3	**56.2**
DMoE	0	63.4	60.4	54.3	57.0	53.6	58.8	56.8	55.1	57.8	55.5	49.4	**56.6**
	4	62.3	59.6	53.3	57.1	55.3	57.6	56.1	54.9	57.4	55.1	48.4	56.1

Table 21: In-context learning results on XStoryCloze across all languages. "**High**", "**Medium**", "**Low**" and "**Ex-Low**" denotes the available amount of linguistic resources. † denotes the unseen language in the pre-training corpus of BLOOM.

Model	#shot	High					Medium	
		en	fr	ru†	zh	ja†	pt	**Avg**
$BLOOM_{560M}$	0	54.0	51.8	50.8	51.7	62.3	51.3	53.7
	4	53.6	48.2	50.1	52.4	61.7	53.6	53.3
$BLOOM_{560M}$ + Pre-train	0	52.9	55.4	50.2	52.1	63.1	55.9	54.9
	4	53.7	48.2	51.1	53.7	64.5	51.7	53.8
Branch-Train-Mix	0	52.9	54.2	49.8	54.9	61.5	52.9	54.4
	4	54.5	51.8	49.4	55.9	62.5	51.0	54.2
DMoE	0	53.2	51.8	50.7	57.8	63.3	53.6	**55.1**
	4	53.7	55.4	50.2	55.6	64.3	51.3	**55.1**
$BLOOM_{1.7B}$	0	55.7	50.6	50.8	54.3	65.9	53.2	55.1
	4	56.1	51.8	51.8	54.3	66.7	52.1	55.5
$BLOOM_{1.7B}$ + Pre-train	0	55.1	51.8	51.8	54.6	65.3	54.4	55.5
	4	55.4	53.0	51.2	54.6	65.5	55.1	55.8
Branch-Train-Mix	0	55.1	51.8	50.6	55.2	65.3	55.5	55.6
	4	55.7	53.0	50.8	55.6	64.9	54.8	55.8
DMoE	0	54.6	50.6	52.6	57.1	66.7	57.0	**56.4**
	4	56.0	53.0	51.2	56.5	67.1	54.8	**56.4**

Table 22: In-context learning results on XWinograd across all languages. "**High**" and "**Medium**" denotes the available amount of linguistic resources. † denotes the unseen language in the pre-training corpus of BLOOM.

D Licenses of Scientific Artifacts

We follow and report the licenses of scientific artifacts involved in Table 23.

Name	License
Transformers	Apache 2.0 license
X-ELM	Apache 2.0 license
lm-evaluation-harness	MIT license
matplotlib	PSF license
Bloom	BigScience RAIL 1.0 license
Gemma	Gemma license
CulturaX	ODC-BY and CC0 license
MADLAD-400	CC-BY-4.0 license

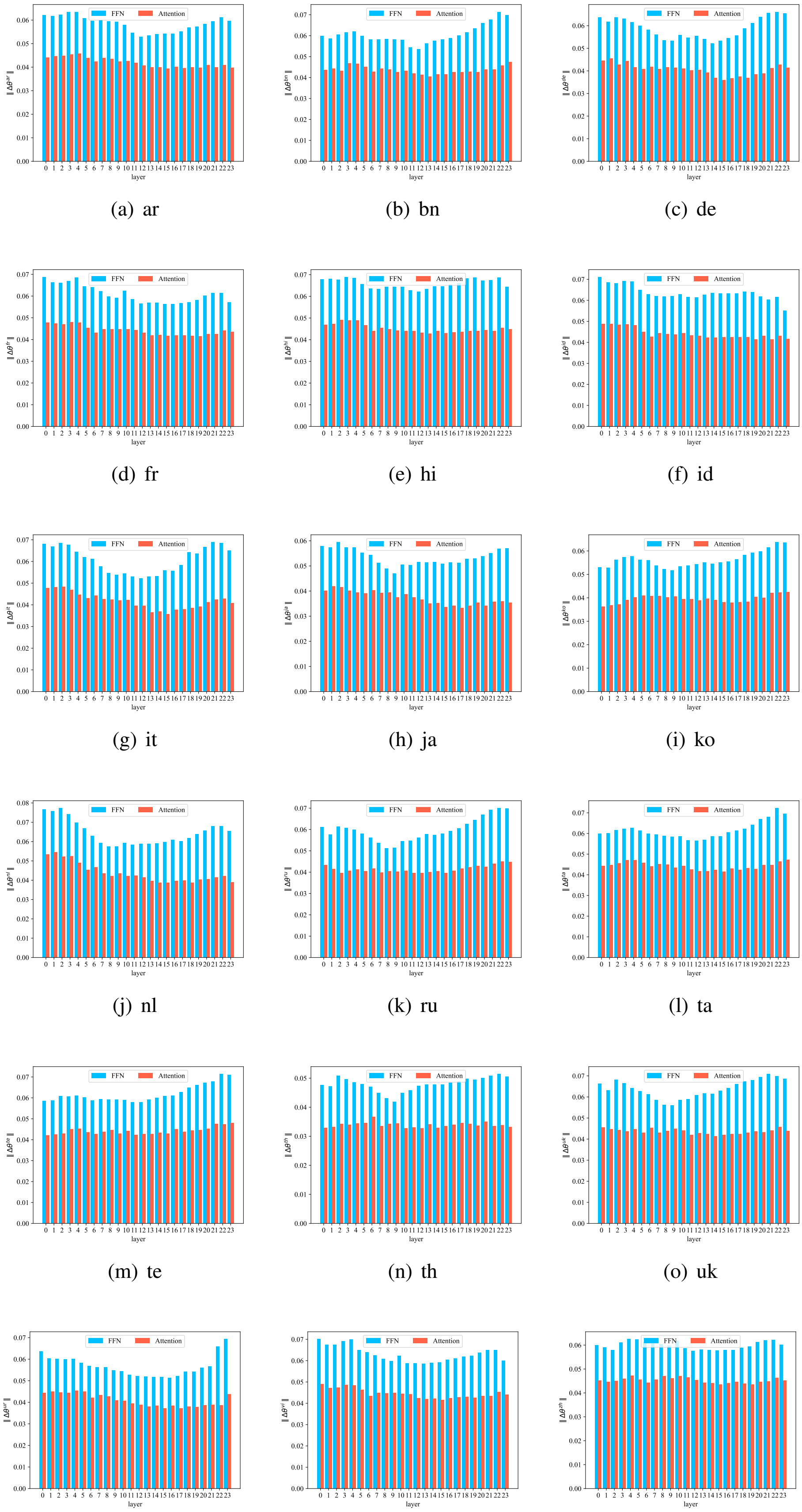
Table 23: Licenses of scientific artifacts involved in this work.

E Additional Information about Language Codes

Table 24 reports more information about the language codes involved in this work.

ISO 639-1/2	Language	Family
am	Amharic	Afro-Asiatic, Semitic
ar*	Arabic	Afro-Asiatic, Semitic
av	Avaric	Northeast Caucasian, Avar-Andic
az	Azerbaijani	Turkic, Common Turkic
be	Belarusian	Indo-European, Balto-Slavic
bn*	Bangla	Indo-European, Indo-Iranian
bo	Tibetan	Sino-Tibetan, Tibeto-Burman
br	Breton	Indo-European, Celtic
ca	Catalan	Indo-European, Italic
ce	Chechen	Northeast Caucasian, Nakh
ceb	Cebuano	Austronesian, Malayo-Polynesian
ckb	Central Kurdish	Indo-European, Indo-Iranian
cnh	Chin Haka	Sino-Tibetan, Tibeto-Burman
co	Corsican	Indo-European, Italic
cs	Czech	Indo-European, Balto-Slavic
da	Danish	Indo-European, Germanic
de*	German	Indo-European, Germanic
dv	Divehi	Indo-European, Indo-Iranian
ee	Ewe	Niger-Congo, Atlantic-Congo
el	Greek	Indo-European, Graeco-Phrygian
en	English	Indo-European, Germanic
eo	Esperanto	Indo-European, Italic
es	Spanish	Indo-European, Italic
et	Estonian	Uralic, Finno-Ugric
eu	Basque	Language isolate
fa	Persian	Indo-European, Indo-Iranian
fi	Finnish	Uralic, Finno-Ugric
fil	Filipino	Austronesian, Malayo-Polynesian
fo	Faroese	Indo-European, Germanic
fr*	French	Indo-European, Italic
fy	Western Frisian	Indo-European, Germanic
gd	Scottish Gaelic	Indo-European, Celtic
gl	Galician	Indo-European, Italic
grc	Ancient Greek	Indo-European, Hellenic
gsw	Swiss German	Indo-European, Germanic
gu	Gujarati	Indo-European, Indo-Iranian
ha	Hausa	Afro-Asiatic, Chadic
haw	Hawaiian	Austronesian, Malayo-Polynesian
he	Hebrew	Afro-Asiatic, Semitic
hi*	Hindi	Indo-European, Indo-Iranian
hil	Hiligaynon	Austronesian, Malayo-Polynesian
hmn	Hmong	Hmong-Mien, Hmongic
ht	Haitian Creole	French Creole, Circum-Caribbean French
hu	Hungarian	Uralic, Finno-Ugric
id*	Indonesian	Austronesian, Malayo-Polynesian
ig	Igbo	Niger-Congo, Atlantic-Congo
ilo	Iloco	Austronesian, Malayo-Polynesian
is	Icelandic	Indo-European, Germanic
it*	Italian	Indo-European, Italic
ja*	Japanese	Japonic
jv	Javanese	Austronesian, Malayo-Polynesian
ka	Georgian	Kartvelian, Karto-Zan
kaa	Karakalpak	Turkic, Common Turkic
kbd	Kabardian	Northwest Caucasian, Circassian
kha	Khasi	Austroasiatic, Khasi-Palaungic
kk	Kazakh	Turkic, Common Turkic
kl	Greenlandic	Eskaleut, Eskimo
km	Khmer	Austroasiatic, Khmer
kn	Kannada	Dravidian, Proto-Dravidian
ko*	Korean	Koreanic, Korean
ky	Kyrgyz	Turkic, Common Turkic
la	Latin	Indo-European, Italic
lb	Luxembourgish	Indo-European, Germanic
lg	Ganda	Niger-Congo, Atlantic-Congo
lo	Lao	Kra-Dai, Tai
lus	Mizo	Sino-Tibetan, Tibeto-Burman
lv	Latvian	Indo-European, Balto-Slavic
mg	Malagasy	Austronesian, Malayo-Polynesian
mi	Maori	Austronesian, Malayo-Polynesian
mk	Macedonian	Indo-European, Balto-Slavic
ml	Malayalam	Dravidian, Southern
mn	Mongolian	Mongolic, Central Mongolic
mr	Marathi	Indo-European, Indo-Iranian
ms	Malay	Austronesian, Malayo-Polynesian
mt	Maltese	Afro-Asiatic, Semitic
my	Burmese	Sino-Tibetan, Tibeto-Burman
ne	Nepali	Indo-European, Indo-Iranian
nl*	Dutch	Indo-European, Germanic
no	Norwegian	Indo-European, Germanic
ny	Chewa	Niger-Congo, Atlantic-Congo
oc	Occitan	Indo-European, Italic
om	Oromo	Afro-Asiatic, Cushitic
os	Ossetian	Indo-European, Indo-Iranian
pa	Punjabi	Indo-European, Indo-Iranian
pap	Papiamento	Portuguese Creole, Afro-Portuguese
pl	Polish	Indo-European, Balto-Slavic
ps	Pashto	Indo-European, Indo-Iranian
pt	Portuguese	Indo-European, Italic
rm	Romansh	Indo-European, Italic
ro	Romanian	Indo-European, Italic
ru*	Russian	Indo-European, Balto-Slavic
sa	Sanskrit	Indo-European, Indo-Iranian
sah	Yakut	Turkic, Common Turkic
sd	Sindhi	Indo-European, Indo-Iranian
se	Northern Sami	Uralic, Sami
sl	Slovenian	Indo-European, Balto-Slavic
sm	Samoan	Austronesian, Malayo-Polynesian
sn	Shona	Niger-Congo, Atlantic-Congo
so	Somali	Afro-Asiatic, Cushitic
sr	Serbian	Indo-European, Balto-Slavic
st	Sotho	Niger-Congo, Atlantic-Congo
su	Sundanese	Austronesian, Malayo-Polynesian
sw	Swahili	Niger-Congo, Atlantic-Congo
ta*	Tamil	Dravidian, Southern
te*	Telugu	Dravidian, Southern
tet	Tetum	Austronesian, Malayo-Polynesian
tg	Tajik	Indo-European, Indo-Iranian
th*	Thai	Kra-Dai, Tai
ti	Tigrinya	Afro-Asiatic, Semitic
tk	Turkmen	Turkic, Common Turkic
to	Tongan	Austronesian, Malayo-Polynesian
tr	Turkish	Turkic, Common Turkic
ts	Tsonga	Niger-Congo, Atlantic-Congo
tt	Tatar	Turkic, Common Turkic
tyv	Tuvan	Turkic, Common Turkic
udm	Udmurt	Uralic, Permic
ug	Uyghur	Turkic, Common Turkic
uk*	Ukrainian	Indo-European, Balto-Slavic
ur*	Urdu	Indo-European, Indo-Iranian
uz	Uzbek	Turkic, Common Turkic
vec	Venetian	Indo-European, Italic
vi*	Vietnamese	Austroasiatic, Vietic
xh	Xhosa	Niger-Congo, Atlantic-Congo
yi	Yiddish	Indo-European, Germanic
yo	Yoruba	Niger-Congo, Atlantic-Congo
yue	Yue Chinese	Sino-Tibetan, Sinitic
zh*	Chinese	Sino-Tibetan, Sinitic
zu	Zulu	Niger-Congo, Atlantic-Congo

Table 24: Details of language codes in this work. $\star$ denotes the language used in the 18 languages experiment.



(a) ar (b) bn (c) de (d) fr (e) hi (f) id (g) it (h) ja (i) ko (j) nl (k) ru (l) ta (m) te (n) th (o) uk (p) ur (q) vi (r) zh

Figure 10: The distribution of parameter deviation $\|\Delta\theta^x\|$ across layers of BLOOM$_{560M}$ for 18 languages.

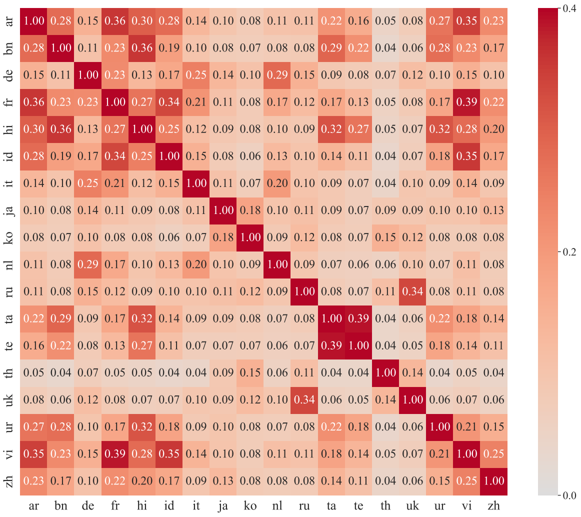


Figure 11: The cosine similarity between 18 languages using all parameter deviation of $BLOOM_{1.7B}$.

Figure 12: The cosine similarity between 18 languages using the parameter deviation of $BLOOM_{1.7B}$ at the last 3 layers.

Figure 13: The cosine similarity between 18 languages using the parameter deviation of $Gemma_{2B}$ at the last 3 layers.

Figure 14: The cosine similarity between 18 languages using the parameter deviation of $BLOOM_{1.7B}$ on the MADLAD-400 multilingual corpus.

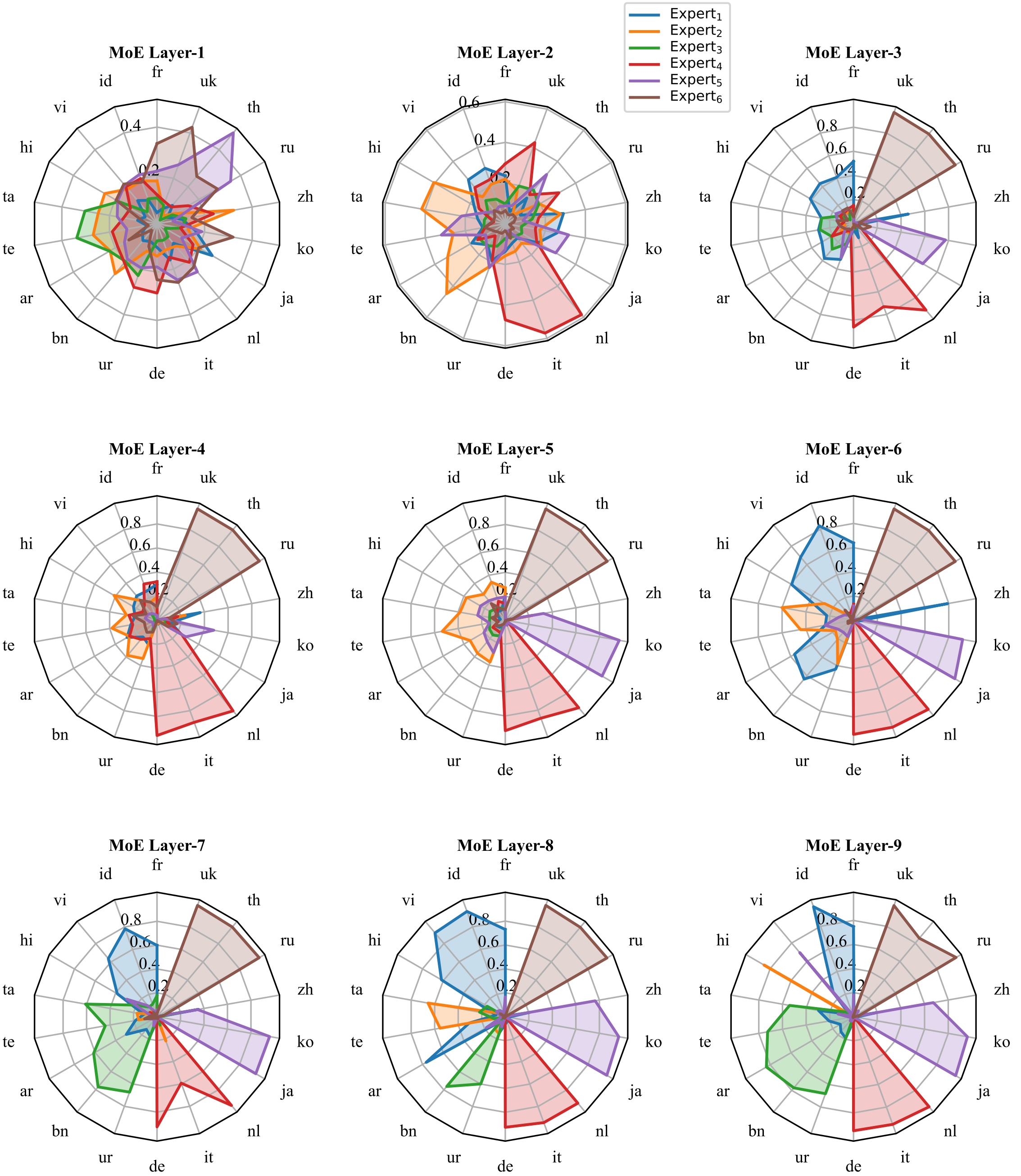


Figure 15: The router distribution of top-1 expert for texts in different languages on models trained with randomly initialized router.

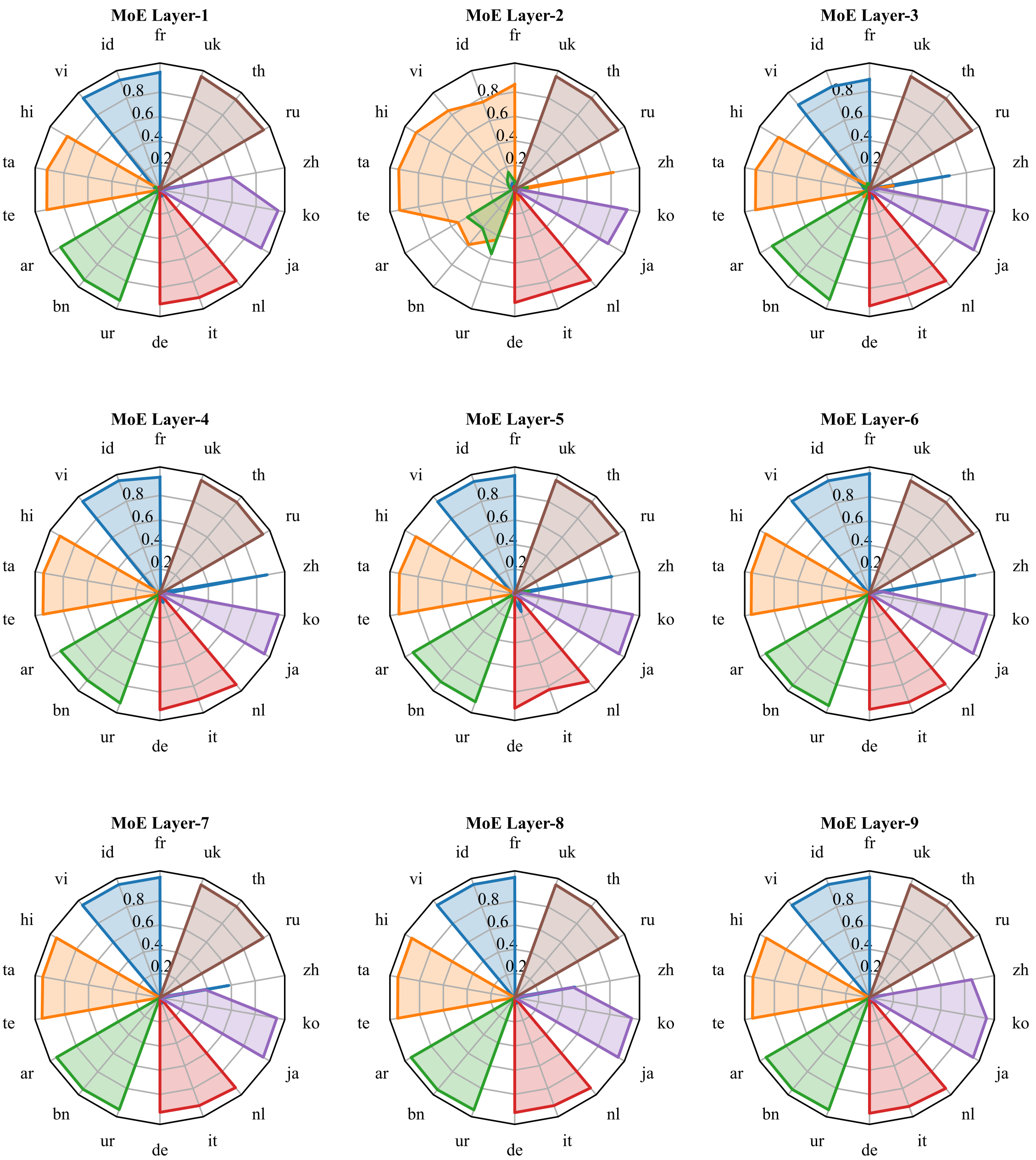


Figure 16: The router distribution of top-1 expert for texts in different languages on models trained with router language classification loss.