How Vocabulary Sharing Facilitates Multilingualism in LLaMA?

Fei Yuan[1], **Shuai Yuan** [2], **Zhiyong Wu**[1], **Lei Li**[3]
[1] Shanghai Artificial Intelligence Laboratory
[2] Hong Kong University of Science and Technology
[3] Carnegie Mellon University
`{yuanfei, wuzhiyong}@pjlab.org.cn`, `syuanaf@connect.ust.hk`, `leili@cs.cmu.edu`

Abstract

Large Language Models (LLMs), often show strong performance on English tasks, while exhibiting limitations on other languages. What is an LLM's multilingual capability when it is trained only on certain languages? The underlying mechanism remains unclear. This study endeavors to examine the multilingual capability of LLMs from the vocabulary sharing perspective by conducting an exhaustive analysis across 101 languages. Through the investigation of the performance gap before and after embedding fine-tuning, we discovered four distinct quadrants. By delving into each quadrant we provide actionable and efficient guidelines for tuning these languages. Extensive experiments reveal that existing LLMs possess multilingual capabilities that surpass our expectations, and we can significantly improve the multilingual performance of LLMs based on these attributes of each quadrant [1].

1 Introduction

Large Language Models (LLM), such as GPT (Brown et al., 2020; OpenAI, 2023), PaLM (Chowdhery et al., 2022), and LLaMA (Touvron et al., 2023a,b), are trained on massive amounts of text data. While these models show strong capabilities on English tasks, their performance in other languages is often limited (Zhu et al., 2023a; Bang et al., 2023).

Significant research effort has been dedicated to enhancing multilingual capabilities by using methods such as continued training with abundant monolingual data (Cui et al., 2023; Yang et al., 2023), or employing instruction-tuning [2] techniques (Zhu et al., 2023b; Li et al., 2023). Despite the encouraging results, the underlying mechanism of LLM's multilingual capability remains mysterious.

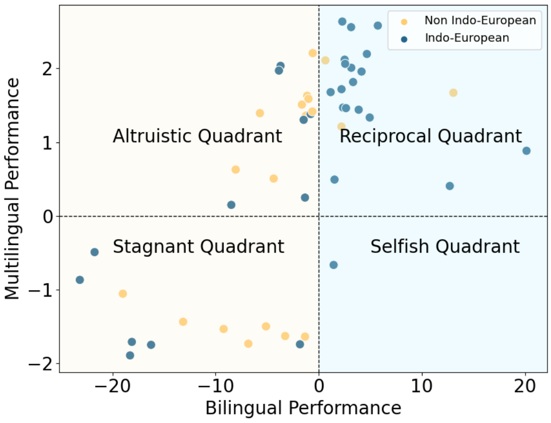


Figure 1: Multilingual capability quadrant. This graph, based on the TED dataset, plots the performance of models fine-tuned with bilingual instructions. Each point represents a model's performance gain over the original LLaMA. The horizontal axis measures the improvement in bilingual performance, while the vertical axis indicates the enhancement in multilingual performance.

Multilingual capability (Lin et al., 2019) refers to how effectively models that have been fine-tuned in one source language can be applied to tasks in other languages and achieve decent performance. This ability has been extensively studied in machine translation (Johnson et al., 2017; Gu et al., 2018; Neubig and Hu, 2018; Aharoni et al., 2019; Zhang et al., 2020) and multilingual pre-trained models (Pires et al., 2019; Libovický et al., 2019; Wu and Dredze, 2020). However, it has not been investigated for English-centric LLMs, given that the pre-training data is predominantly in English. We aim to address this issue by focusing on the multilingual foundation of pre-trained LLMs and providing some guidance to help other people train LLMs more efficiently for non-English languages. Generally, multilingual capabilities are built on two key foundations: the volume of multilingual data used during the pre-training stage (Touvron et al., 2023a,b; Li et al., 2023; Scao et al., 2022), and the

[1] `https://github.com/CONE-MT/Vocabulary-Sharing-Facilitates-Multilingualism`.

[2] Instruction tuning is a method used to train large language models to follow specific instructions to solve a task. We provide an example of instruction tuning format in Appendix A.

vocabulary (Pires et al., 2019; Chung et al., 2020; Liang et al., 2023). In this work, we focus on the latter: vocabulary.

To investigate the multilingual foundation provided by the vocabulary of an existing LLM, we only fine-tune the embedding layer and keep the rest of the parameters frozen, denoted as Embed FT. This approach requires fewer adjustments to the model parameters than full fine-tuning, and unlike LoRA (Hu et al., 2021), it doesn't require any additional model structure. In our experiments, we focus on the LLaMA as a case study, but the analysis method can be applied to other LLMs.

To examine the multilingual capabilities of LLMs without loss of generality, we applied Embed FT to a 10k en→x bilingual instruction translation dataset generated by 10k sentences pairs across four distinct datasets: Lego-MT (Yuan et al., 2023), Wikimatrix (Schwenk et al., 2021) and Newscommentary (Tiedemann, 2012), and Ted (Ye et al., 2018). We evaluated the bilingual performance (refers to the performance of the fine-tuning languages) and multilingual performance (refers to the performance of other languages) of each model to determine if there was a significant positive or negative change compared to the original model. From the results, all languages can be categorized into four distinct quadrants.

The multilingual capability quadrant of the TED dataset, illustrated in Figure 1, includes four quadrants: the reciprocal quadrant, the altruistic quadrant, the stagnant quadrant, and the selfish quadrant. The full definition of each quadrant is in Section 3. The selfish quadrant refers to scenarios where the fine-tuned model only improves on the fine-tuning language directions but not other languages. It is considered a default quadrant, as languages that fall into the selfish quadrant exhibit behavior that aligns intuitively with the effects of bilingual fine-tuning.

Certain languages such as Bulgarian fall into the reciprocal quadrant, where training with bilingual data (e.g. English→Bulgarian) not only enhances bilingual performance but also boosts the multilingual capabilities of other languages. The majority of these languages in this quadrant are from the Indo-European family, benefiting from the pretraining data and vocabulary sharing. For these languages, we find that there is no need to fine-tune all parameters, which could lead to overfitting to a specific language. We recommend fine-tuning only the embedding layer, which yields bilingual performance on par with full fine-tuning while preserving the model's multilingual capabilities.

Remarkably, certain languages exhibit altruistic characteristics. When we use these languages as training data, their primary effect is to enhance multilingual performance. Upon further analysis, we discovered that the decline in bilingual performance is primarily due to a change in error types: from those that are easy to score to those that are more challenging. The improvement in multilingual performance, on the other hand, stems from vocabulary sharing. For such languages, employing a small dataset for full fine-tuning can be more effective for multilingual capabilities.

Indeed, there are certain languages located in the stagnant quadrant that are quite stubborn. This means that using data from these languages doesn't improve bilingual performance or bring about multilingual benefits. Regardless of parameter-effective tuning strategies (LoRA) or extensive fine-tuning on large datasets, the results are still disappointing. Interestingly, even expanding the vocabulary for full fine-tuning doesn't lead to better results. Then, we find that existing LLMs often over-tokenized these languages, which reduces the density of information they carry. By simply removing the common prefix of tokenized representation, we have seen an average improvement of 2.5 spBLEU points. Our main contributions are:

- We conduct a systematic analysis of the impacts of LLM's vocabulary on their multilingual capabilities, and discover four quadrants based on their embedding fine-tuning performance gap.
- We provide practical and efficient technical guides to improve multilingual capabilities for each quadrant.
- We perform extensive experiments to verify the effectiveness of quadrant-specific fine-tuning techniques (e.g. 2.5 spBLEU improvement in stagnant quadrant).

2 Background

Multilingual Large Language Model Large language models (LLMs; OpenAI, 2023; Zhang et al., 2022; Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023a,b have shown demonstrated performance in English, but the performance in other languages is limited. To address this limitation, researchers have proposed multilingual language models (MLLMs) that can handle multiple languages simultaneously. The first line of research proposes to learn a shared representation space for multiple languages by first pre-training on multilingual data and then fine-tuning for specific tasks or languages. Representative works include mBERT (Devlin et al., 2019), XLM (Lample and Conneau, 2019), XLMR (Conneau et al., 2020), BLOOM (Scao et al., 2022), XGLM (Lin et al., 2022b), and PolyLM (Wei et al., 2023). Another line of research adopted existing monolingual LLMs to multilingual using techniques such as prompt engineering (Muennighoff et al., 2023; Yong et al., 2023), instruction tuning (Zhu et al., 2023b; Li et al., 2023; Jiao et al., 2023), or continue training (Cui et al., 2023; Yang et al., 2023).

The Multilingual Foundation of LLM The robust multilingual capabilities of LLM are founded on: the presence of diverse multilingual data (Touvron et al., 2023a,b; Li et al., 2023; Scao et al., 2022) and vocabulary (Pires et al., 2019; Chung et al., 2020; Liang et al., 2023).

The size of multilingual data is a critical factor in the multilingual capabilities of LLM. LLaMA (Touvron et al., 2023a) is pre-trained on a vast scale, with over 1.6 trillion tokens, of which less than 9% is multilingual data, [3] spanning 20 different languages. LLaMA2 (Touvron et al., 2023b) further enhances the proportion of multilingual data to approximately 11% and increases the number of languages to around 26. PolyLM (Wei et al., 2023) is trained on 640 billion tokens and supports 18 of the most commonly spoken languages. BLOOM (Scao et al., 2022) is trained with data from 46 natural languages. The existing language data in the pre-training phase provides LLM with a robust foundation for multilingual capabilities.

Another key factor is vocabulary construction. A common approach to constructing vocabulary involves tokenizing text into subwords: including Byte-level Byte-Pair-Encoding (BBPE), Byte-Pair-Encoding (BPE), SentencePiece (SP) (Sennrich et al., 2016; Kudo and Richardson, 2018; Wang et al., 2019), which are units smaller than words that can encapsulate morphological variations. Nevertheless, in a multilingual context encompassing a diverse range of scripts, the base vocabulary comprising subwords can become exceedingly large, leading to inefficiency and sparsity. Further Information on BBPE is in Appendix B.

3 Inherent Multilingual Capabilities

In this section, we begin by exploring the inherent multilingual capabilities of LLMs and give some fascinating observations detailed in Section 3.1. Drawing on these insights, we then proceed to conduct an in-depth examination of the multilingual capability of LLM in Section 3.2.

3.1 Observation

Setting We fine-tune a single LLaMA model using en→x data on the Lego-MT dataset, yielding 101 bilingual-tuned models. We train the LLaMA-7B with en→ro, en→no, en→ms, and en→luo data separately, and then thoroughly evaluate each bilingual-tuned model on all 101 language pairs (en→x) to probe its multilingual translation performance on Flores-101's devtest set. Additionally, we follow the same settings for all models throughout the paper. Over 50 bilingual models were tuned using the Wikimatrix and Newscommentary datasets, and more than 55 bilingual models were tuned using the Ted dataset. All these models were trained with identical parameter settings, specifically a learning rate of 2e-5 and a total of 3 epochs, and evaluated bilingual and multilingual performance with beam size = 4.

Phenomena We observe that LLM demonstrates superior multilingual capabilities far beyond expectation. Some interesting phenomena are:

Phenomenon 1: LLaMA can support additional languages beyond those explicitly mentioned in their pretraining corpus. In the leftmost part of Figure 2, it is evident that the bilingual-finetuned en→ro, en→ms, and en→no models exhibit a significant improvement over the original model in en→af translation. This outcome is quite surprising considering that neither LLaMA's pretraining corpus [4] nor our fine-tuning data contain any text related to af. Similar observations can be made for numerous other languages, as depicted in Figure 2. This indicates the LLaMA may possess a more robust capability for handling multiple languages than previously expected.

[3] The original wording (4.5%) in the LLaMA paper, which only mentioned the inclusion of 20 languages of Wikipedia data. After meticulously checking the datasets involved in the LLaMA pre-training to provide a rigorous account of the quantity of non-English data, we discovered that the Gutenberg dataset includes some multilingual data.

[4] LLaMA utilizes Wikipedia for its pre-training data which includes 20 languages: bg, ca, cs, da, de, en, es, fr, hr, hu, it, nl, pl, pt, ro, ru, sl, sr, sv, uk.

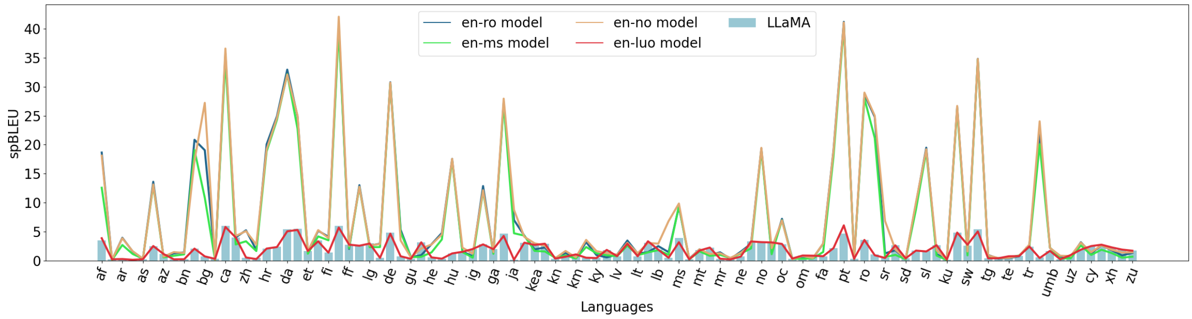


Figure 2: We evaluated the multilingual capabilities of various models on the Flores-101 dataset. The bar graph represents the direct inference results from the original LLaMA, while the line graph illustrates the multilingual performance of models trained on bilingual instruction data from en→ro, en→ms, en→no, and en→luo.

Phenomenon 2: The performance distribution of bilingual-tuned models across multiple languages exhibits remarkable consistency. Intuitively, different bilingual models shall have very different multilingual performance distributions. However, from line plots in Figure 2, we observe that three bilingual models (en→ro, en→no, and en→ms) showcase an exceptional level of consistency. We speculate such a phenomenon might be caused by a similar instruction-tuning process. However, further experiments on the en→luo model reject the above hypothesis. Therefore, we hypothesize that such a phenomenon only occurs in certain languages and might be related to certain underexplored mechanisms.

3.2 Quantify Multilingual Capability at Scale

Given the phenomena above, we scale our evaluation to more languages to validate our findings.

Setting We conduct experiments on LLaMA with 3 epochs on all 101 language pairs(en→x) in Flores-101 using parallel multilingual corpora. For each language pair, we sample at most 10k sentence pairs at random unless it has fewer than 10k sentence pairs. We then train models using the Embed FT method. For evaluation on Flores-101's devtest including 12 respective languages (details provided in Appendix E), we use a beam size of 4 and spBLEU (SentencePiece BLEU) as the metric.

Observation Inspecting the large-scale evaluation results, we make the following observation: some bilingual models exhibit highly similar yet surprising behaviors. As a counterintuitive example, we showcase a group of "selfless" bilingual models (see Table 1). In common belief, fine-tuning LLM on one language pair shall definitely improve its performance. However, to our surprise, fine-tuning these "selfless" bilingual models(column LG) might even hurt their performance(comparing en→LG column with LLaMA column). What's even more interesting is that the multilingual performance of these models is significantly improved.

Type	LG	LLaMA	en→af	en→ro	en→LG	Multilingual
	af	3.5	15.6	20.0	15.6	17.8
	ro	3.6	18.6	28.7	28.7	23.7
selfless	ln	2.9	7.9	20.9	0.9	14.4
	ns	3.3	7.9	22.6	1.4	15.3
	lo	1.8	8.7	17.8	0.1	13.3
	km	1.1	9.7	21.3	0.1	15.5
	ig	2.0	9.7	19.8	1.2	14.7
	ps	0.9	8.9	17.2	0.5	13.1
	my	0.3	11.2	22.8	0.0	17.0
	lv	0.7	10.5	22.5	0.4	16.5
	xh	2.3	9.4	21.7	2.0	15.5
	mn	0.2	12.0	22.8	0.0	17.4
	am	0.2	8.3	14.9	0.0	11.6
	pa	0.3	8.8	18.8	0.1	13.8

Table 1: Consistent performance gains in translation across multiple languages. Each row represents a model that has been trained using en→LG bilingual dataset. Multilingual performance refers to the average result of en→af and en→ro.

To quantitatively investigate the language clustering behavior, as well as dig the root of the phenomena mentioned above, we propose to categorize languages into four quadrants using a two-dimensional Cartesian system. As shown in Figure 1, the x-axis represents bilingual performance, and the y-axis represents multilingual performance. Before clustering, we first establish a categorization criteria.

Criteria We use the bilingual/multilingual performance changes before and after fine-tuning to measure whether the tuning results in gain or loss:

$$\Delta_{\text{lg}} = \begin{cases} \frac{P_{\text{post}}}{P_{\text{pre}}} - 2, & \text{if } P_{\text{pre}} \geq T \\ \frac{P_{\text{post}} - 2T}{P_{\text{pre}}}, & \text{otherwise} \end{cases} \quad (1)$$

where the P_{post} represents the translation performance after fine-tuning, P_{pre} indicates the performance before the fine-tuning process, T serves as a threshold for smoothing, and 2 is a hyperparameter quantifies the extend for significant changes. We select based on a preliminary study, for further information see Appendix D. The calculation of Δ_{lg} for bilingual performance is straightforward, for the multilingual performance, we consider the average performance of en→af and en→ro translations. This is primarily due to our observation that changes in multilingual performance are significantly mirrored in that of en→af and en→ro, details are in Appendix E.

Quadrant Details We calculate the above criteria on four multilingual corpora: Lego-MT (Yuan et al., 2023), Wikimatrix (Schwenk et al., 2021) and Newscommentary (Tiedemann, 2012), and Ted (Ye et al., 2018), and obtain a consistent language classification results as in Table 2. The details of datasets and categorization are in Appendix E. We summarize the behavior of four quadrants below (also shown in Figure 1):

- *Reciprocal Quadrant*: Models trained on languages from reciprocal quadrant, demonstrate strong bilingual and multilingual performance at the same time.
- *Altruistic Quadrant*: Models trained on these languages prioritize enhancing others, with minimal impact on their bilingual performance.
- *Stagnant Quadrant*: Existing tuning strategies appear to have minimal impact on these languages.
- *Selfish Quadrant*: The selfish quadrant is the most intuitive one: training in a specific language typically improves the performance of that language and merely affects other languages.

Please note that the categorization proposed is merely one possibility derived from certain criteria, and there might exist alternatives that lead to slightly different classification results. Nonetheless, We only focus on the consistent classification, produced by Eq. 1, across four distinct datasets for our later analysis. We leave the exploration of a better classification metric as future work.

4 Enhancing Multilingual Capability

This section conducts a comprehensive analysis of the properties and training strategies of each quadrant to effectively enhance the multilingual capability of LLMs.

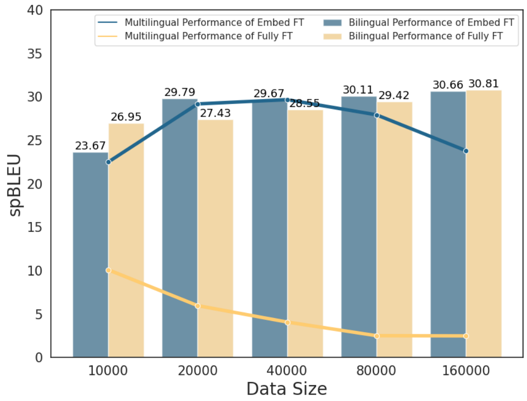


Figure 3: Comparing the Embed FT and Full FT Strategies. In the realm of bilingual performance, both strategies prove equally effective. However, when it comes to multilingual performance, the Embed FT strategy stands out for its adaptability across various languages, while the Full FT strategy tends to over-specialize the model to a single language. The numerical results for each language pair can be found in Appendix G.

4.1 Reciprocal Quadrant

Language within the reciprocal quadrant indicates that using any of these languages as training data invariably improves performance in other languages within the same group. We will delve into this relationship to uncover some intriguing insights.

Interpretation: Reciprocal quadrant consists of linguistically similar languages. The reciprocal quadrant is predominantly occupied by Indo-European languages. These languages are grouped mainly due to their shared vocabulary and grammatical affixes. Furthermore, the original 20 languages supported by LLaMA are predominantly Indo-European, providing a solid foundation. Consequently, tuning one language within the Indo-European family can effectively enhance the performance of other languages within the same family.

Practice Guidance 1: The recommended training strategy for reciprocal languages is Embed FT, which achieves the best performance-generalization trade-off. Figure 3 illustrates the performance disparity between the models obtained through Embed FT and Full FT strategies under varying amounts of training data. We randomly selected 11 languages from the reciprocal quadrant for testing, including es, pt, ca, de, da, cs, bg, pl,

Dataset	Reciprocal Quadrant	Selfish Quadrant	Altruistic Quadrant	Stagnant Quadrant
Lego-MT	af, bs, bg, ca, hr, cs, da, mk, ms, no, oc, pl, pt, ro, sk, sl	ast, be, tl, fr, gl, de, hu, id, it, ky, lt, ml, mt, mi, ny, fa, ru, sr, es, sw, sv, tg, uk	am, ar, hy, as, bn, my, ceb, zh, et, fi, gu, he, is, ig, ga, jv, km, ko, lo, lv, ln, mr, mn, ne, ns, ps, pa, sd, so, tr, ur, uz, vi, cy, xh, zu	te, zhtrad, ff, lg, el, ja, kam, kk, luo, lb, or, om, sn, ku, ta, th, umb, wo, yo
New	bs, bg, ca, hr, cs, da, nl, fr, gl, de, el, hi, hu, id, it, ja, mk, no, pl, pt, ro, ru, sr, sk, sl, es, sv, uk		ar, az, be, zh, et, tl, fi, ka, he, is, jv, kk, ko, lt, lb, mr, ne, oc, fa, sw, tg, te, tr, vi	bn, ml, ta
Ted	bg, hr, cs, da, nl, fr, de, el, hu, id, it, ja, mk, pl, pt, ro, ru, sk, sl, es, sv	hi	ar, et, fi, gl, ka, he, ko, lt, mr, fa, sr, th, tr, uk, vi	hy, az, be, bn, bs, my, zh, kk, ms, mn, ku, ta, ur
Summary	bg, id, de, ru, da, mk, hu, it, pl, cs, hr, sl, es, sk, sv, ro, pt, fr		mr, ko, he, fi, et, vi, tr, ar	-

Table 2: The distribution of various languages across different quadrants. Various factors such as data influence and tuning strategy can lead to instability in some language quadrants. However, we concentrate on languages that demonstrate consistent stability within these quadrants. In the stagnant quadrant, given that different datasets encompass varying numbers of languages, we also take into account the observations.

fr, ru, nl, and averaged the bilingual/multilingual performance across all 11 languages.

For bilingual performance, the Embed FT strategy works as well as the Full FT strategy. As depicted by the bar in Figure 3, the results indicate that when working with a limited dataset, the model trained by Embed FT demonstrates a slightly inferior performance compared to Full FT. However, as the size of the dataset increases, the model developed using Embed FT not only matches but may even exceed the performance of Full FT.

For multilingual performance, the Embed FT strategy excels in adapting to various languages, while the Full FT strategy tends to make the model overly specialized to a particular language. As illustrated by the line in Figure 3, the findings suggest that full fine-tuning of a bilingual dataset may lead to overfitting, but this can be effectively mitigated by using the Embed FT strategy.

Practice Guidance 2: While the Full FT model's multilingual capabilities are influenced by language quantity, the Embed FT model remains unaffected. Considering Phenomenon 3, which observes a consistent multilingual distribution, we are curious to explore whether a richer language number could bring additional performance gains. To investigate this, we randomly select some languages from the reciprocal quadrant to establish a multilingual setting, and the results of this experiment are displayed in Table 3. In the Full FT, the performance of the multilingual model improves with an increase in the number of languages. However, in the Embed FT, the number of languages does not have a significant impact.

4.2 Altruistic Quadrant

Languages that fall into this quadrant demonstrate a "selfless" characteristic. Training based on the data from these languages does not necessarily improve, and may even decrease their performance. Interestingly, it can lead to performance enhancements in other languages. We will conduct a thorough examination of the underlying causes of this phenomenon and propose potential solutions.

Interpretation for bilingual performance decline: The model transitions from an error type that is easy to score to a less score-friendly error type. The primary error for LLaMA is "source copy", which simply duplicates the source sentence as the translation. This error often leads to moderate scores when there are names, numbers, and punctuation in the translation tasks. However, after tuning, the main error shifts to "oscillatory hallucination" (Li et al., 2023), a state where the model becomes stuck in a specific translation state and generates repeated n-grams until it reaches the maximum length. This error makes it challenging to earn the score of spBLEU. Therefore, the performance of the fine-tuned model is lower than that of the original model.

Interpretation for multilingual performance improvement: Those languages' vocabulary encompasses the majority of English tokens. We estimate the linguistics of these languages on the Flores-101 benchmark, a multilingual parallel corpus translated by professional translators through a controlled process. For an altruistic language, *LG* we first employ LLaMA's tokenizer to segment the words in both the *LG* and English data from Flores. This allows us to compile the sets of tokens that belong to the *LG* language, denoted as S_{LG}, and the English language, denoted as S_{En}. Finally, we calculate the ratios of the size of $S_{LG} \bigcap S_{En}$ to the size of S_{LG} and the size of S_{En} respectively. Intriguingly, as shown in Figure 4, we discovered that most tokenized results used in these languages exhibit a high degree of consistency with English.

Practice Guidance: Full FT with a minimal dataset can effectively enhance bilingual per-

# Lang	Data Size	en→hr	en→da	en→no	en→ro	en→ca	en→cs	en→bg	en→pl	en→es	en→fr	en→de	en→pt	en→nl	AVG.
Bilingual Full Fine-Tuning															
	20k	20.2	32.2	22.2	28.8	35.8	24.5	26.5	18.4	23.8	31.7	24.8	41.1	18.9	26.8
	40k	21.2	32.8	24.0	29.6	37.0	25.4	27.4	18.8	25.2	34.1	25.9	41.3	22.1	28.1
	80k	22.4	34.8	25.6	30.8	38.5	26.4	29.3	19.1	23.6	32.9	30.8	40.6	23.5	29.1
Multilingual Full Fine-Tuning															
2	160k	22.9	17.2	8.7	19.0	24.9	17.8	5.1	8.7	10.7	4.5	5.4	9.6	23.7	13.7
4	40k	20.0	31.1	18.6	28.6	35.6	24.0	20.6	18.4	26.4	36.2	27.3	38.3	23.6	26.8
8	80k	20.2	28.1	21.7	28.8	36.4	24.9	27.1	19.4	25.9	37.1	25.5	41.2	24.8	27.8
Multilingual Embedding Fine-Tuning															
2	160k	21.5	33.1	18.5	29.5	36.0	25.6	20.5	18.8	26.9	41.8	30.7	41.5	24.8	28.4
4	40k	19.9	33.3	19.2	29.7	37.1	24.9	26.7	19.6	26.8	42.8	30.8	41.0	25.3	29.0
8	80k	20.3	32.8	19.2	28.6	34.6	24.6	27.0	19.1	27.0	40.0	29.9	40.7	24.5	28.3

Table 3: Performance comparison of bilingual and multilingual models. In full fine-tuning, multilingual models improve with more languages. However, in embedding fine-tuning, language quantity doesn't significantly affect performance. Notably, multilingual models slightly underperform compared to bilingual models. In the table, a data size of 80k for 8 languages implies that each language contributes 10k sentence pairs. Out of curiosity about the performance of LLaMA in Indo-European languages, which it does not claim to support, we have added two additional languages, hr and no, during the inference process, based on Guidance 1.

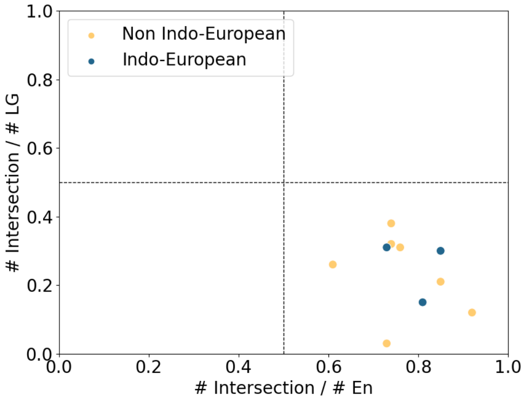


Figure 4: Analyzing linguistics in altruistic languages. A significant overlap in tokenized results with English may enhance performance in Indo-European languages.

formance and maintain a robust multilingual effect. As shown in Table 4, the altruistic trait is exemplified across different training strategies. However, with Full FT and LoRA, as the dataset size increases, the model tends to overfit the specific language, thereby diminishing its multilingual capabilities. For Embed FT, an increase in data volume does not significantly alter bilingual performance, but it does markedly enhance the multilingual effect. Interestingly, the multilingual effect is not significantly different from that of Full FT with a small dataset. In summary, by employing a small dataset for full fine-tuning, we can strike a balance between bilingual and multilingual performance.

4.3 Stagnant Quadrant

Languages in this quadrant exhibit remarkable inertia, as training with their data neither enhances their own performance nor influences the performance of other languages. In this section, we will delve deeper into the inertia phenomenon, examining its potential causes and proposing possible solutions.

Setting	Size	en→vi		en→tr		en→ar		AVG.	
		B	M	B	M	B	M	B	M
LLaMA		1.9	3.6	2.4	3.6	0.26	3.6	1.5	3.6
FT	10k	14.8	24.4	7.2	19.9	5.4	**24.7**	9.1	23.0
	20k	18.5	22.3	8.3	9.3	6.9	22.9	11.2	18.2
	40k	**22.3**	15.9	**10.1**	6.6	**9.3**	21.5	**13.9**	14.7
LoRA	10k	4.9	**24.8**	4.1	23.8	4.3	23.3	4.4	24.0
	20k	6.5	24.4	4.6	23.0	5.3	23.5	5.5	23.6
	40k	7.2	18.0	5.1	17.0	5.8	21.0	6.0	18.7
Embed	10k	3.1	14.5	2.7	14.2	3.1	11.9	3.0	13.5
	20k	3.6	23.3	2.8	23.5	4.2	23.0	3.5	23.3
	40k	3.5	24.7	2.9	**24.8**	4.5	23.6	3.6	**24.4**

Table 4: The altruistic characteristic is evident in a range of training strategies when trained with the en→vi, en→tr, and en→ar bilingual datasets. Here, "B" denotes the bilingual performance, while "M" signifies the average performance of en→af and en→ro.

Interpretation: Most languages in the stagnant quadrant are characterized by over-tokenization. The LLaMA tokenizer, based on the BBPE algorithm, is fundamental for multilingual language processing tasks. Its universal applicability to all languages and the lack of a need for an 'unknown' token make it optimal for vocabulary sharing and increase its robustness. Despite being suitable for multilingual learning, BBPE results in byte sequence representation of text that is often much longer (up to 4x) than a character sequence representation. Upon investigation, we find that the over-tokenization phenomenon is prevalent in LLaMA. In an extreme case, a sentence in lo that contains 6 words expands to 352 tokens after tokenization. Additional details in Appendix F.

A comparison between active and stagnant languages, as shown in Table 5, reveals that:

Setting	Ratio	LLaMA	Full Bilingual Fine-Tuning					LoRA Bilingual Tuning				
			10k	20k	40K	80k	160k	10k	20k	40K	80k	160k
en→es	1.7	4.8	23.5	23.8	25.2	23.6	25.9	26.4	25.8	26.6	26.3	26.9
en→pt	1.9	6.0	41.3	41.1	41.3	40.6	39.7	42.0	42.0	42.4	42.0	41.6
en→ca	1.9	5.7	34.9	35.7	37.0	38.5	39.2	37.3	37.7	38.1	38.6	39.2
en→de	2.0	4.7	22.5	24.8	25.9	30.8	31.2	27.8	26.8	27.3	31.9	32.6
en→no	2.2	3.2	21.2	22.2	24.0	25.6	28.4	19.6	20.1	21.0	22.1	24.0
en→ro	2.3	3.5	28.3	28.7	29.6	30.8	34.3	29.8	30.0	30.9	31.2	32.7
en→da	2.3	4.9	31.9	32.2	32.8	34.8	36.4	33.4	34.0	34.5	35.3	36.1
en→bs	2.6	2.0	23.2	25.2	26.5	28.5	30.0	21.7	22.8	24.2	25.0	25.2
en→gu	15.0	0.3	2.3	2.2	4.4	10.0	13.2	1.0	1.1	1.5	1.9	3.1
en→kn	16.9	0.3	1.0	1.5	3.0	5.6	9.9	0.5	0.4	0.5	0.8	1.0
en→te	17.4	0.7	4.2	8.2	12.8	17.3	20.3	0.6	0.8	1.7	2.9	5.3
en→ku	17.6	0.2	0.1	0.1	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
en→my	21.7	0.3	1.0	2.0	4.1	7.3	9.4	0.1	0.1	0.3	0.3	0.4
en→mr	38.8	0.3	5.0	7.2	10.7	13.7	15.8	1.8	1.9	2.2	2.0	1.7
en→lo	39.8	1.8	1.5	2.3	3.7	7.1	9.8	0.4	0.7	0.7	0.6	0.7
en→km	43.0	1.1	1.6	3.1	6.2	10.1	13.4	0.2	0.2	0.5	0.9	1.5

Table 5: The relationship between stagnant languages and the characteristic of over-tokenization. The "Ratio" is defined as the number of tokens in a sequence after applying the tokenizer, divided by the sentence length, which is measured by the number of words for space-separated languages and characters.

Source	**Type**	**3k**	**6k**	**12k**	**Source**	**Type**	**3k**	**6k**	**12k**
km - 10.1					lo - 7.1				
MC4	BBPE	5.2	3.7	2.3	MC4	BBPE	6.2	1.7	3.6
	BPE	4.7	11.0	2.1		BPE	6.7	1.8	3.6
	SP	6.2	**11.8**	10.3		SP	**7.0**	6.4	4.9
Flores	BBPE	4.6	3.5	8.5	Flores	BBPE	4.6	3.9	1.5
	BPE	4.4	3.7	8.8		BPE	4.3	1.5	1.6
	SP	5.5	4.4	-		SP	2.4	4.2	-
gu - 10.0					te - 17.3				
MC4	BBPE	0.4	0.3	0.3	MC4	BBPE	9.6	8.4	6.0
	BPE	0.4	0.2	0.3		BPE	9.7	7.7	6.7
	SP	0.3	0.2	0.4		SP	**10.0**	9.7	8.1
Flores	BBPE	0.3	0.3	0.3	Flores	BBPE	9.0	8.8	7.1
	BPE	0.3	0.3	0.3		BPE	8.9	8.2	7.2
	SP	**0.4**	-	-		SP	9.8	-	-

Table 6: Exploring various strategies for vocabulary expansion: The term "km - 10.1" denotes the bilingual performance (10.1) of full fine-tuning on Lego-MT 80k bilingual data (en→km) without any vocabulary extension. "3k", "6k", and "12k" refer to the extended vocabulary size. Most vocabulary expansion methods do not significantly enhance the performance of stagnant languages. Due to the limited data in Flores dev, some settings are missing in the table.

- activating a stagnant language with full fine-tuning requires more data;
- the performance improvement with increasing data is modest;
- certain parameter efficiency fine-tuning strategies, like LoRA, do not affect them.

Practice Guidance 1: Expanding the vocabulary is not an effective strategy for stagnant languages. Over-tokenization leads to an increased demand for data. When a language is not adequately represented by its vocabulary, the common approach is to expand the lexicon (Tai et al., 2020; Cui et al., 2023; Ji et al., 2023). Regrettably, in most instances, this strategy of vocabulary enlargement proves ineffective for stagnant languages. As shown in Table 6, we present three distinct methods to expand the vocabulary:

- BBPE (Wang et al., 2019): This follows the approach used in LLaMA for vocabulary construction and involves learning a vocabulary for stagnant language;
- BPE (Sennrich et al., 2016)): This utilizes the BPE algorithm and is based on subword units to learn a vocabulary;
- SP (Kudo and Richardson, 2018): This learns a vocabulary using the SentencePiece algorithm.

Meanwhile, to mitigate potential issues from data quality, we have utilized both MC4 and Flores-101 dev to construct vocabulary.

After training LLaMA on Lego-MT 80k bilingual data, the experimental results indicate that:

- When there is a substantial amount of data, the impact of data quality on vocabulary expansion can be disregarded;
- Among all the vocabulary expansion methods, SP tends to yield better results compared to other solutions;
- Almost all vocabulary expansion techniques fail to enhance the performance of stagnant languages significantly.

Practice Guidance 2: Shortening the subword sequences can significantly boost the performance of stagnant languages. Given the existence of the over-tokenization problem, we find that among these over-tokenized languages, there are a

Setting	en→km	en→lo	en→gu	en→te	AVG.
Ratio	47.6%	67.0%	66.8%	73.8%	63.8%
Full FT	10.1	7.1	10.0	17.3	11.1
Extend (Best)	11.8	7.0	0.4	10.0	7.3
Our Strategy	**12.6**	**9.2**	**11.3**	**21.5**	**13.7**
Δ	+ 2.5	+ 2.1	+ 1.3	+ 4.2	+ 2.6

Table 7: Over-tokenization leads to a decrease in information density for LLM. However, by simply removing the over-tokenized character that shares the same prefix, we can enhance performance, achieving results that surpass both full fine-tuning and vocabulary extension.

large amount characters. For example, a Chinese character “饕” is encoded into three code units “[227, 234, 260]”. We refer to such characters as ‘over-tokenized characters’ for the sake of simplicity. We then gather all these over-tokenized characters along with their three-byte representations. Interestingly, these over-tokenized characters constitute a significant proportion, about 63.8%, of the corpus, as indicated in Table 7. Moreover, in the case of over-tokenized languages, all over-tokenized characters begin with the same token (e.g., 227). Therefore, the obtained three-byte representations are very sparse and result in low information density in representation.

Furthermore, we propose a post-tokenization technique to address the over-tokenization problem. We simply remove the shared prefix of over-tokenization characters and obtain the shortened yet lossless new representations. As a concrete example, we remove 饕’s prefix [227] from its three-byte representation [227, 234, 260] to get a more compact two-byte representation [234, 260]. Subsequently, we utilized this adjusted representation to train LLaMA on the 80k Lego-MT bilingual dataset. Remarkably, our method outperforms both direct fine-tuning of LLaMA and vocabulary extension, achieving a substantial performance boost with an average of 2.5 points.

4.4 Guidance Summary

Guidance for reciprocal languages For languages, primarily Indo-European languages, situated in the reciprocal quadrant, the optimal strategy is to solely fine-tune the embedding layer and keep the remaining parameters frozen. This is primarily due to these languages having shared vocabulary and grammar rules.

Guidance for altruistic languages For languages residing in the altruistic quadrant, applying full fine-tuning with a minimal dataset can effectively enhance bilingual performance while maintaining a robust multilingual effect. This is mainly because the tokens in the vocabulary of these languages highly overlap with English.

Guidance for stagnant languages Shortening subword sequences can markedly enhance the performance of stagnant languages. Most languages in the stagnant quadrant are over-tokenized, which refers to a situation where a text in this language is typically segmented by a tokenizer into an excessively large number of tokens on average. Expanding the vocabulary does not necessarily enhance the performance of these languages. However, in this paper, we demonstrate that simply removing the identical prefix from over-tokenized characters can significantly improve performance.

We provide more analysis about stagnant languages (in Appendix F), tuning analysis (in Appendix G), and different NLP tasks (in Appendix H) of LLaMA experiments in the Appendix.

5 Conclusion

In this study, we performed a comprehensive analysis of 101 languages, categorizing them based on shared characteristics into four distinct quadrants: reciprocal, altruistic, selfish, and stagnant quadrants. Upon examining each quadrant in-depth, we identified the primary reasons for the placement of languages within their respective quadrants and provided some practical guidance for training. However, the primary focus of this study is the analysis of persistent language characteristics within each quadrant. A thorough investigation into the conditions that trigger language migration across various phenomena is a subject for our future research.

Limitation

In this paper, we find some interesting phenomena in LLaMA. After expanding our evaluation to include more languages, we found that many of them demonstrated remarkably similar behaviors on translation tasks. Then we grouped them with categorization criteria. Although language classification is not our primary focus, our main interest lies in understanding the reasons behind these classifications and enhancing the multilingual capabilities of LLM. Meanwhile, to delve deeper into the role of Embed FT, we provide a more detailed analysis in Appendix H.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. *arXiv preprint arXiv:1903.00089*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Dan Garrette, Kiat Chuan Tan, and Jason Riesa. 2020. Improving multilingual models with language-clustered vocabularies. *arXiv preprint arXiv:2010.12777*.

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.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. 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*. Association for Computational Linguistics.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

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

Yunjie Ji, Yan Gong, Yong Deng, Yiping Peng, Qiang Niu, Baochang Ma, and Xiangang Li. 2023. Towards better instruction following language models for chinese: Investigating the impact of training data and evaluation.

Wenxiang Jiao, Jen tse Huang, Wenxuan Wang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Parrot: Translating during chat using large language models. In *ArXiv*.

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

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining.

Jiahuan Li, Hao Zhou, Shujian Huang, Shanbo Cheng, and Jiajun Chen. 2023. Eliciting the translation ability of large language models via multilingual finetuning with translation instructions.

Davis Liang, Hila Gonen, Yuning Mao, Rui Hou, Naman Goyal, Marjan Ghazvininejad, Luke Zettlemoyer, and Madian Khabsa. 2023. Xlm-v: Overcoming the vocabulary bottleneck in multilingual masked language models.

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2019. How language-neutral is multilingual bert?

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. 2022a. 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.

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. 2022b. Few-shot learning with multilingual language models.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Rose Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir R. Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2022. Crosslingual generalization through multitask finetuning. *ArXiv*, abs/2211.01786.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

OpenAI. 2023. Gpt-4 technical report.

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

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. Xcopa: A multilingual dataset for causal commonsense reasoning.

Teven Le Scao, Angela Fan, Christopher Akiki, Elizabeth-Jane Pavlick, Suzana Ili'c, Daniel Hesslow, Roman Castagn'e, Alexandra Sasha Luccioni, Franccois Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Rose Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurenccon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa Etxabe, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris C. Emezue, Christopher Klamm, Colin Leong, Daniel Alexander van Strien, David Ifeoluwa Adelani, Dragomir R. Radev, Eduardo Gonz'alez Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady ElSahar, Hamza Benyamina, Hieu Trung Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jorg Frohberg, Josephine L. Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro von Werra, Leon Weber, Long Phan, Loubna Ben Allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, Mar'ia Grandury, Mario vSavsko, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad Ali Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto L'opez, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, S. Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal V. Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Févry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiang Tang, Zheng Xin Yong, Zhiqing Sun, Shaked Brody, Y Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre Franccois Lavall'ee, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aur'elie N'ev'eol, Charles Lovering, Daniel H Garrette, Deepak R. Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Xiangru Tang, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, S. Osher Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdenvek Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ananda Santa Rosa Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Olusola Ajibade, Bharat Kumar Saxena, Carlos Muñoz Ferrandis, Danish Contractor, David M. Lansky, Davis David, Douwe Kiela, Duong Anh Nguyen, Edward Tan, Emily Baylor, Ezinwanne Ozoani, Fatim T Mirza, Frankline Ononiwu, Habib Rezanejad, H.A. Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jan Passmore, Joshua Seltzer, Julio Bonis Sanz, Karen Fort, Lívia Macedo Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, M. K. K. Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nourhan Fahmy, Olanrewaju Samuel, Ran An, R. P. Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas L. Wang, Sourav Roy, Sylvain Viguier, Thanh-Cong Le, Tobi Oyebade, Trieu Nguyen Hai Le, Yoyo Yang, Zachary Kyle Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Kumar Singh, Benjamin Beilharz, Bo Wang, Caio Matheus Fonseca de Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel Le'on Perin'an, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully A. Burns, Helena U. Vrabec, Iman I.B. Bello, Isha Dash, Ji Soo Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthi Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, María Andrea Castillo, Marianna Nezhurina, Mario Sanger, Matthias Samwald, Michael Cullan, Michael Weinberg, M Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patricia Haller, R. Chandrasekhar, R. Eisenberg, Robert Martin, Rodrigo L. Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Pratap Bharati, T. A. Laud, Th'eo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yashasvi Bajaj, Y. Venkatraman, Yifan Xu, Ying Xu, Yun chao Xu, Zhee Xao Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2022. Bloom: A 176b-parameter open-access multilingual language model. *ArXiv*, abs/2211.05100.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

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. 2022. Language models are multilingual chain-of-thought reasoners.

Wen Tai, H. T. Kung, Xin Dong, Marcus Comiter, and Chang-Fu Kuo. 2020. exBERT: Extending pre-trained models with domain-specific vocabulary under constrained training resources. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1433–1439, Online. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

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

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

Changhan Wang, Kyunghyun Cho, and Jiatao Gu. 2019. Neural machine translation with byte-level subwords.

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. 2023. Polylm: An open source polyglot large language model.

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. Bigtrans: Augmenting large language models with multilingual translation capability over 100 languages. *arXiv preprint arXiv:2305.18098*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. Paws-x: A cross-lingual adversarial dataset for paraphrase identification.

Qi Ye, Sachan Devendra, Felix Matthieu, Padmanabhan Sarguna, and Neubig Graham. 2018. When and why are pre-trained word embeddings useful for neural machine translation. In *HLT-NAACL*.

Zheng-Xin Yong, Ruochen Zhang, Jessica Zosa Forde, Skyler Wang, Arjun Subramonian, Holy Lovenia, Samuel Cahyawijaya, Genta Indra Winata, Lintang Sutawika, Jan Christian Blaise Cruz, Yin Lin Tan, Long Phan, Rowena Garcia, Thamar Solorio, and Alham Fikri Aji. 2023. Prompting multilingual large language models to generate code-mixed texts: The case of south east asian languages.

Fei Yuan, Yinquan Lu, Wenhao Zhu, Lingpeng Kong, Lei Li, Yu Qiao, and Jingjing Xu. 2023. Lego-MT: Learning detachable models for massively multilingual machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11518–11533, Toronto, Canada. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. *arXiv preprint arXiv:2004.11867*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023a. Multilingual machine translation with large language models: Empirical results and analysis. *arXiv preprint arXiv:2304.04675*.

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

A Instruction Tuning

Instruction Tuning is a method used to train large language models to follow specific instructions to solve a task. It's a form of supervised learning where the model is trained on a dataset consisting of pairs of instructions and corresponding outputs.

Table 8 shows a simple example in the Lego-MT dataset, presented in the format used for instruction tuning:

Below is an instruction that describes a task, paired with an input that provides further context. Write a response that appropriately completes the request.
Instruction: Translate the following sentences from English to French.
Input: Dogs are the main source of transmission of rabies to humans.
Response: Les chiens sont la principale source de transmission de la rage.

Table 8: Instruction tuning case based on Lego-MT dataset.

B BBPE

In a multilingual context encompassing a diverse range of scripts, the base vocabulary comprising subwords can become exceedingly large, leading to inefficiency and sparsity. To mitigate this problem, BBPE has emerged as the standard practice in most modern language modeling efforts (Muennighoff et al., 2022; Scao et al., 2022; Zhang et al., 2022; Touvron et al., 2023a,b), which leverages UTF-8 encoding that encodes each Unicode character into 1 to 4 one-byte (8-bit) code units. BBPE is a tokenization algorithm capable of tokenizing any word in any language, thereby eliminating the necessity for an unknown token. It optimizes vocabulary sharing across numerous languages and delivers superior performance, facilitating knowledge transfer between languages with non-overlapping character sets.

C Language Information

In this section, we classify languages according to their respective language families, as depicted in Table 9. We standardize all language codes using the ISO 639-1 standard. For clarity, we list all languages by their full names and shade the corresponding languages in gray for easy identification.

D Hyper-parameter Setting

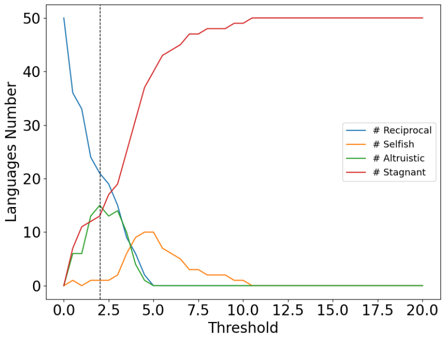


Figure 5: Hyper-parameter setting. "Threshold" refers to the significant changes before and after tuning, which are calculated by dividing the performance after tuning by the performance before the tuning. "# Reciprocal" denotes the count of languages in the Reciprocal quadrant. The experimental result demonstrates that a substantial increase in the threshold value could lead to all languages being classified into the Stagnant quadrant.

We use the criteria to measure the bilingual/multilingual performance changes before and after fine-tuning:

$$\Delta_{\text{lg}} = \begin{cases} \frac{P_{\text{post}}}{P_{\text{pre}}} - 2, & \text{if } P_{\text{pre}} \geq T \\ \frac{P_{\text{post}} - 2T}{P_{\text{pre}}}, & \text{otherwise} \end{cases}$$

The threshold term, T, is used to smooth the dramatic numerical change that might be caused by low-performing languages(e.g., performance change from 0.01 to 0.02, although negligible, will be considered significant without re-balancing using T). We set T to the vanilla model's average translation performance on the Flores-101 dataset.

The hyper-parameter, set to a value of 2, defines the thresholds for determining significant changes before and after tuning. Here, we consider a language to have significant bilingual/multilingual performance changes if the performance after tuning is twice that of the performance before tuning. In Figure 5, we have thoroughly tested different significance thresholds and found that if we consider a 20-fold difference (a very large value) in performance before and after tuning, then all languages would be regarded as stagnant languages.

E Quadrant Division

We use some different publicly multilingual datasets: Lego-MT, Wikimatrix & Newcommentary, and Ted, which come from a different domain, as shown in Table 10.

Conducting a comprehensive evaluation of the translation performance for all en→x pairs in Flores-101 across all models is a task that demands

Family-1	Family-2	Family-3	ISO	Language	Lang Family-1	Family-2	Family-3	ISO	Language
Indo-European	Armenian		hy	Armenian	Kartvelian	Karto-Zan	Georgian	ka	Georgian
	Balto-Slavic	Baltic	lt	Lithuanian	Koreanic	Korean		ko	Korean
			lv	Latvian	Kra–Dai	Tai	Southwestern Tai	lo	Lao
		Slavic	be	Belarusian				th	Thai
			bg	Bulgarian	Mongolic	Central	Mongolian	mn	Mongolian
			bs	Bosnian	Niger–Congo	Atlantic–Congo	Atlantic	wo	Wolof
			cs	Czech			Benue–Congo	ln	Lingala
			hr	Croatian				ns	Northern Sotho
			mk	Macedonian			Volta-Congo	lg	Luganda
			pl	Polish				ny	Nyanja
			ru	Russian				sn	Shona
			sk	Slovak				sw	Swahili
			sl	Slovenian				umb	Umbundu
			sr	Serbian				xh	Xhosa
			uk	Ukrainian				yo	Yoruba
	Celtic	Insular Celtic	cy	Welsh				zu	Zulu
			ga	Irish				ig	Igbo
	Germanic	North Germanic	is	Icelandic				kam	Kamba
			sv	Swedish			West Atlantic	ff	Fulani
		Northwest Germanic	da	Danish	Nilo-Saharan	Eastern	Nilotic	luo	Dholuo
			no	Norwegian	Portuguese	Afro-Portuguese	Upper Guinea Creole	kea	Kabuverdianu
		West Germanic	af	Afrikaans	Sino-Tibetan	Sinitic	Chinese	zh	Chinese
			de	German				zhtrad	Chinese
			en	English		Tibeto-Burman	Lolo-Burmese	my	Burmese
			lb	Luxembourgish	Turkic	Common	Karluk	uz	Uzbek
			nl	Dutch			Kipchak	kk	Kazakh
	Graeco-Phrygian	Hellenic	el	Greek				ky	Kyrgyz
	Indo-Aryan	Eastern	bn	Bengali			Oghuz	az	Azerbaijani
	Indo-Iranian	Indo-Aryan	as	Assamese				tr	Turkish
			gu	Gujarati	Uralic	Finno-Permic	Finno-Samic	et	Estonian
			hi	Hindi		Finno-Ugric	Finnic	fi	Finnish
			mr	Marathi			Ugric	hu	Hungarian
			ne	Nepali	Afro-Asiatic	Chadic	West Chadic	ha	Hausa
			or	Odia		Cushitic	Lowland East Cushitic	om	Oromo
			pa	Punjabi				so	Somali
			sd	Sindhi		Semitic	West Semitic	am	Amharic
			ur	Urdu				ar	Arabic
		Iranian	fa	Persian				he	Hebrew
			ku	Kurdish				mt	Maltese
			ps	Pashto	Austroasiatic	Khmer		km	Khmer
			tg	Tajik		Vietic	Viet–Muong	vi	Vietnamese
	Italic	Latino-Faliscan	ast	Asturian	Austronesian	Malayo-Polynesian	Javanese	jv	Javanese
			ca	Catalan			Malayic	id	Indonesian
			es	Spanish				ms	Malay
			fr	French			Oceanic	mi	Maori
			gl	Galician			Philippine	ceb	Cebuano
			it	Italian				tl	Tagalog
			oc	Occitan	Dravidian	South-Central	Telugu	te	Telugu
			pt	Portuguese		Southern	Tamil–Kannada	kn	Kannada
			ro	Romanian				ml	Malayalam
Japonic	Japanese		ja	Japanese				ta	Tamil

Table 9: This table provides information on the language families of all 101 languages included in FLores-101. The language family information is presented at three levels, denoted as "Lang Family-x", where 'x' stands for the level (1, 2, or 3). For ease of reference, we also provide the ISO code and the full name of each language. Languages that are used in the inherent multilingual analysis are highlighted with a gray background.

Dataset	# Language	Domain
Lego-MT	100	Web
Wikimatrix & Newscommentary	50	Wikipedia and News
Ted	55	TED talk

Table 10: Statistics of various publicly accessible parallel multilingual corpora.

Lang	Language Family	Lang	Language Family
ha	Afro-Asiatic	he	Afro-Asiatic
mi	Austronesian	ta	Dravidian
af	Indo-European	ro	Indo-European
th	Kra–Dai	ns	Niger–Congo
luo	Nilo-Saharan	zh	Sino-Tibetan
tr	Turkic	et	Uralic

Table 11: Representative languages information. Within the Indo-European language family, we choose to include af in addition to ro, which is a first language in South Africa and not initially listed as a supported language by LLaMA.

significant labor and resources. Therefore, we randomly select one representative language from each language family for subsequent testing, as shown in Table 11.

The bilingual and multilingual performance of the model trained on the TED dataset on Flores-101 devtest is shown in Table 12.

F Stagnant Quadrant

The LLaMA tokenizer, built on the BBPE algorithm, serves as the foundation for multilingual language processing tasks. Its universal applicability across all languages, coupled with the elimination of the need for an "unknown" token, enhances

LG	en→mi	en→luo	en→ns	en→ha	en→ta	en→tr	en→he	en→af	en→ro	en→th	en→zh	en→et	en→LG
LLaMA	2.3	3.1	3.3	3.1	0.4	2.4	0.5	3.5	3.6	0.8	0.5	1.6	-
ar	0.9	1.5	0.8	0.3	0.4	1.9	2.5	7.8	16.1	0.3	1.4	1.4	2.2
hy	1.0	0.9	0.6	0.0	0.3	1.9	1.2	3.7	4.3	0.1	0.9	1.1	0.9
az	1.2	2.1	1.9	1.3	0.0	1.8	0.0	2.5	0.1	0.0	0.5	1.1	0.0
be	0.9	2.4	1.9	2.1	0.0	1.3	0.0	2.0	0.0	0.0	0.5	0.9	0.0
bn	0.6	2.0	1.4	1.5	0.0	0.9	0.0	1.8	0.0	0.0	0.5	0.7	0.1
bs	1.1	2.2	1.9	1.7	0.0	1.4	0.0	1.8	0.0	0.0	0.4	0.9	0.4
bg	1.5	1.7	1.3	0.5	0.3	2.1	2.0	8.2	12.1	0.2	1.1	1.5	18.8
my	0.3	0.3	0.4	0.0	0.2	1.5	1.1	3.0	0.9	0.1	1.0	0.8	0.0
zh	0.8	1.7	1.4	1.3	0.0	1.7	0.0	1.7	0.1	0.0	0.5	0.9	0.5
hr	2.0	2.6	2.2	1.8	0.4	2.6	2.8	10.4	19.2	0.5	1.5	1.9	13.8
cs	1.6	2.1	1.3	0.9	0.4	2.1	2.0	9.1	14.4	0.3	1.1	1.7	16.5
da	2.2	2.7	2.3	2.1	0.4	2.6	2.6	10.8	18.2	0.5	1.3	1.9	24.5
nl	1.6	2.3	1.6	1.3	0.4	2.0	2.1	10.4	15.8	0.4	1.3	1.7	23.1
et	1.3	2.0	1.7	1.1	0.4	2.5	2.6	8.7	16.9	0.4	1.3	1.9	1.9
fi	1.5	2.4	2.0	1.6	0.4	2.3	2.2	8.0	15.7	0.3	1.2	1.7	2.0
fr	2.2	2.9	2.7	2.3	0.4	2.9	2.6	9.5	18.4	0.5	1.6	1.8	36.8
gl	1.7	2.3	2.0	0.7	0.4	2.7	2.4	8.9	14.9	0.3	1.1	1.8	3.1
ka	0.9	0.7	0.3	0.0	0.4	1.7	1.8	6.7	11.8	0.2	1.2	1.4	0.1
de	1.8	2.4	1.9	1.6	0.4	2.2	2.4	10.1	16.7	0.5	1.6	1.8	25.9
el	1.4	2.2	1.7	0.9	0.4	2.3	2.8	11.2	21.4	0.3	1.8	1.7	5.4
he	1.7	2.4	2.0	1.1	0.4	2.6	3.4	8.2	21.4	0.5	2.3	1.8	3.4
hi	0.3	0.4	0.2	0.0	0.3	1.4	1.6	3.6	5.8	0.1	1.2	0.8	4.1
hu	1.6	2.3	1.6	1.1	0.4	2.2	1.6	8.4	14.2	0.3	1.1	1.6	6.4
id	2.4	3.1	2.9	2.5	0.4	3.0	2.9	9.1	19.9	0.6	1.4	1.8	7.3
it	2.2	2.7	2.5	2.0	0.4	2.7	2.4	9.7	18.6	0.5	1.5	2.0	23.8
ja	1.6	2.0	1.8	1.1	0.4	1.9	3.0	6.7	19.2	0.5	2.3	1.6	5.4
kk	1.2	2.8	2.6	2.6	0.1	1.9	0.1	2.9	0.6	0.3	0.5	1.3	0.3
ko	1.0	1.9	1.4	1.1	0.4	1.9	1.8	7.4	17.3	0.3	1.9	1.5	2.9
lt	1.6	2.3	2.1	1.6	0.4	2.5	2.4	8.9	19.6	0.5	1.4	1.9	1.0
mk	1.1	0.2	0.2	0.0	0.3	1.7	1.9	7.7	9.8	0.2	1.2	1.2	4.4
ms	1.3	2.6	2.1	1.9	0.1	1.7	0.0	2.5	0.1	0.0	0.5	1.1	2.7
mr	0.8	2.5	2.5	1.3	0.3	2.2	2.2	5.8	9.4	0.4	1.1	1.4	1.1
mn	1.1	1.5	1.2	0.1	0.2	2.0	0.7	4.3	2.3	0.1	0.9	1.1	0.0
fa	0.7	0.9	0.3	0.1	0.4	1.6	1.8	7.3	15.9	0.2	1.7	1.3	2.6
pl	1.8	2.4	2.1	1.7	0.4	2.3	2.3	9.2	15.0	0.4	1.4	1.7	12.6
pt	2.0	2.6	2.2	1.9	0.3	2.6	2.5	11.6	20.7	0.5	1.7	1.9	36.1
ro	1.8	2.3	1.6	1.1	0.4	2.1	2.1	8.9	15.5	0.4	1.2	1.6	15.5
ru	1.3	2.0	1.3	0.7	0.4	1.9	1.5	7.2	9.7	0.3	1.1	1.5	16.6
sr	1.8	2.2	1.7	1.3	0.4	2.3	2.6	10.3	17.7	0.5	1.3	1.7	2.0
sk	1.9	2.3	2.1	1.5	0.4	2.5	2.5	9.0	16.9	0.4	1.2	1.9	5.7
sl	1.8	2.4	1.9	1.4	0.3	2.3	2.3	8.3	16.1	0.4	1.3	1.8	8.0
ku	0.4	1.2	1.5	0.4	0.3	1.6	2.0	4.5	6.1	0.3	1.2	1.2	0.0
es	2.2	2.8	2.5	2.1	0.4	2.8	2.8	11.3	20.8	0.6	1.8	1.9	24.8
sv	1.9	2.6	2.1	1.9	0.4	2.3	2.6	10.4	18.2	0.5	1.3	1.8	24.7
ta	1.3	2.9	2.4	2.1	0.1	2.1	0.0	2.9	0.4	0.0	0.6	1.5	0.1
th	0.7	0.8	0.7	0.1	0.4	2.2	1.8	4.5	13.2	0.4	1.5	1.3	0.4
tr	1.7	2.3	1.8	1.4	0.4	2.4	2.3	8.2	17.1	0.4	1.4	1.7	2.4
uk	1.3	1.9	1.4	0.8	0.4	1.8	1.4	7.0	8.8	0.2	0.9	1.3	3.0
ur	0.6	0.9	0.5	0.3	0.0	1.0	0.0	0.7	0.0	0.0	0.6	0.4	0.0
vi	1.8	2.7	2.4	1.9	0.3	2.5	2.6	8.5	15.6	0.5	1.2	1.8	2.6

Table 12: Assessing the bilingual and multilingual capabilities: a performance evaluation of the model trained on the TED dataset across all representative languages using the Flores-101 devtest. The experimental results show the significant improvement in multilingual performance embodied in the en→af and en→ro.

vocabulary sharing and boosts its robustness. However, a phenomenon known as over-tokenization, marked by excessive segmentation of text into tokens, may occur in certain languages, which could potentially affect the efficiency of language processing tasks.

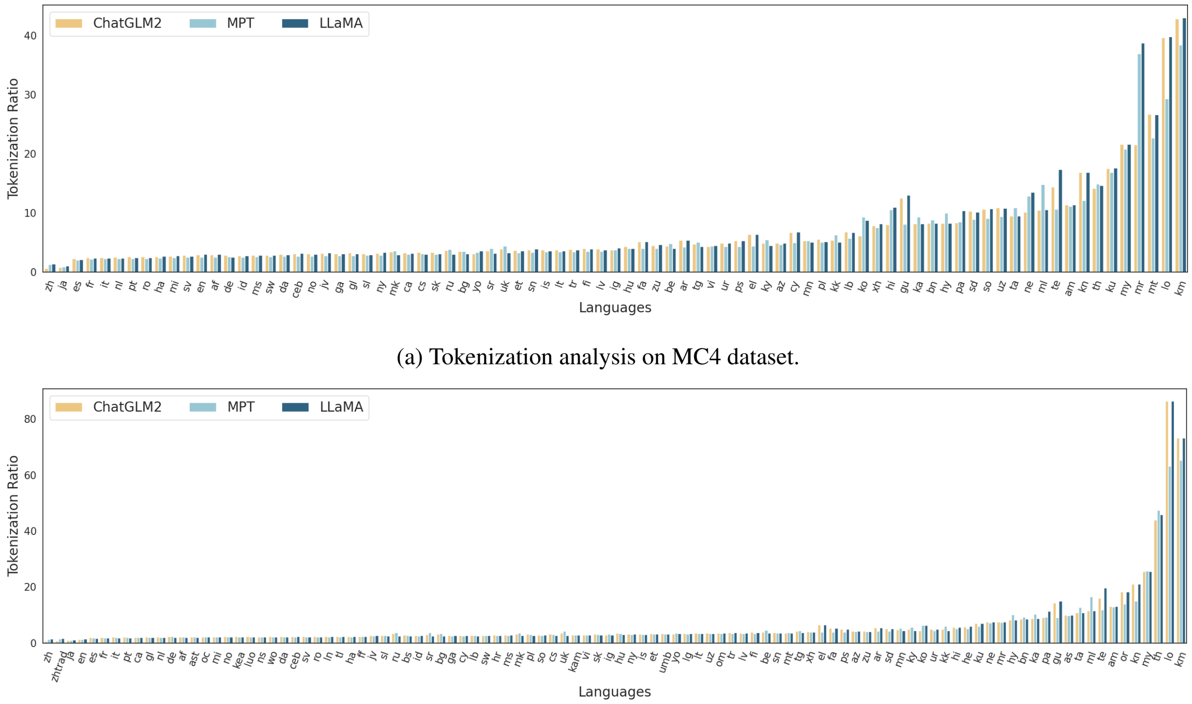
To thoroughly examine the "over-tokenization", we conduct our research using the MC4 (Xue et al., 2021) and Flores-101 (Goyal et al., 2022) dataset. Despite having only 1012 samples, Flores-101 provides a high-quality multilingual parallel corpus that allows for an in-depth exploration of the variations in expressing the same sentence across different languages.

The over-tokenization phenomenon is observable across various datasets and LLMs. For certain languages, such as te and lo, the length of the tokenized sequence that LLaMA processes can extend to 300 or even more. Interestingly, analysis results from the Flores-101 dataset reveal that languages prone to over-tokenization require more tokens to express the same meaning. The magnitude of this phenomenon is notably larger than what was observed in the MC4 dataset, as shown in Figure 6.

We also present tuning results based on our analysis of the Flores-101 dataset, where we examined the effects of full bilingual fine-tuning and Lora tuning on varying amounts of data, as shown in Table 13. Interestingly, we found that the characteristics of stagnant language are preserved.

G Single-layer Tuning

To determine whether fine-tuning parameters of layers other than the embedding layer in the model is equally effective, we conducted a bilingual translation task on eight language pairs in the Flores101



(a) Tokenization analysis on MC4 dataset.

(b) Tokenization analysis on Flores-101 dataset.

Figure 6: An over-tokenization phenomenon in low-resource languages across different datasets and LLMs. The tokenization ratios of LLaMA, ChatGLM2, and MPT are calculated by dividing the length of the tokenized sequence by the sentence length. For space-separated languages, the sentence length is measured by the number of words, while for other languages it is measured by the number of characters. The length of the tokenized sequence refers to the number of tokens obtained after applying the tokenizer. Languages characterized by over-tokenization will exhibit this trait across various LLMs.

Setting	**Ratio**	**LLaMA**	**Full Bilingual Fine-Tuning**					**LoRA Bilingual Tuning**				
			10k	20k	40K	80k	160k	10k	20k	40K	80k	160k
en→es	1.7	4.8	23.5	23.8	25.2	23.6	25.9	26.4	25.8	26.6	26.3	26.9
en→pt	1.9	6.0	41.3	41.1	41.3	40.6	39.7	42.0	42.0	42.4	42.0	41.6
en→ca	1.9	5.7	34.9	35.7	37.0	38.5	39.2	37.3	37.7	38.1	38.6	39.2
en→de	2.0	4.7	22.5	24.8	25.9	30.8	31.2	27.8	26.8	27.3	31.9	32.6
en→no	2.2	3.2	21.2	22.2	24.0	25.6	28.4	19.6	20.1	21.0	22.1	24.0
en→ro	2.3	3.5	28.3	28.7	29.6	30.8	34.3	29.8	30.0	30.9	31.2	32.7
en→da	2.3	4.9	31.9	32.2	32.8	34.8	36.4	33.4	34.0	34.5	35.3	36.1
en→bs	2.6	2.0	23.2	25.2	26.5	28.5	30.0	21.7	22.8	24.2	25.0	25.2
en→as	10.0	0.2	3.2	4.7	6.8	8.2	9.6	0.5	0.6	0.9	1.4	2.2
en→ta	11.0	0.4	2.2	4.3	9.6	15.3	21.4	0.4	0.6	1.0	1.9	3.4
en→pa	11.4	0.3	2.3	4.2	6.8	9.7	14.5	0.4	0.8	1.2	1.7	2.7
en→ml	11.6	0.2	3.1	7.4	13.5	20.3	22.5	0.6	0.9	1.7	3.3	4.1
en→am	13.1	0.2	1.3	4.6	9.6	14.5	18.2	0.1	0.1	0.2	0.4	1.1
en→gu	15.0	0.3	2.3	2.2	4.4	10.0	13.2	1.0	1.1	1.5	1.9	3.1
en→or	18.4	0.3	0.9	1.6	1.6	1.0	0.8	0.3	0.5	0.5	0.1	0.1
en→te	19.7	0.7	4.2	8.2	12.8	17.3	20.3	0.6	0.8	1.7	2.9	5.3
en→kn	21.1	0.3	1.0	1.5	3.0	5.6	9.9	0.5	0.4	0.5	0.8	1.0
en→my	25.7	0.3	1.0	2.0	4.1	7.3	9.4	0.1	0.1	0.3	0.3	0.4
en→th	45.9	0.8	2.6	4.0	6.0	8.4	12.3	1.2	1.5	1.9	2.8	4.1
en→km	73.3	1.1	1.6	3.1	6.2	10.1	13.4	0.2	0.2	0.5	0.9	1.5
en→lo	86.5	1.8	1.5	2.3	3.7	7.1	9.8	0.4	0.7	0.7	0.6	0.7

Table 13: This refers to the relationship between stagnant languages and the characteristic of over-tokenization. Here, the "Ratio" is defined as the number of tokens in a sequence after applying the tokenizer, divided by the sentence length. The sentence length is measured by the number of words for space-separated languages and characters for others.

dataset. These models were fine-tuned on the Alpaca-En dataset, which was primarily used as the training set to minimize any potential impact from language variations. The results of these tests are displayed in Table 14. In these tests, English served as the source language, while the target languages comprised eight different languages.

As observed from the table, the average scores of fine-tuning the embedding layer and Layer 0 are the highest, and they are very close to each other. The model's performance gradually decreases as the layer number increases, with a noticeable drop around the middle layers (Layers 15-17). This trend is remarkably consistent across all language pair tests.

The aforementioned results suggest that solely fine-tuning the parameters of the lower layers can also activate the model's multilingual capabilities, and its effectiveness is comparable to that of embedding fine-tuning. Furthermore, the activation of different language capabilities in the model through single-layer fine-tuning occurs synchronously.

Additionally, we fine-tuned all the lower layers, from Layer 0 to Layer 14, together. As shown in Table 15, this strategy did not yield any additional gains compared to the other tuning strategies.

H More Analysis

The performance of Embed FT remains stable across reciprocal languages, regardless of the dataset being utilized. As depicted in Table 16, the Embed FT strategy delivers performance that is competitive with the FT and LoRA strategies across all training sets: Alpaca-En, Alpaca-X, and Bilingual. Alpaca-En is a comprehensive English dataset with 52k instructions and demonstrations. Alpaca-X is derived from Alpaca-En through translation, with X denoting the target languages. The Bilingual dataset comprises 52k instruction data for translation tasks, based on the open-source Lego-MT dataset. Unlike the FT strategy, which updates all model parameters. Furthermore, it avoids the need for an additional model structure, like the LoRA strategy. This implies that Embed FT is a more effective strategy for activating multilingual capabilities.

In the Flores-101 dataset, the same evaluation metric, spBLEU, is used. Before calculating BLEU, all data is de-tokenized and sentence piece tokenization is applied to each language. This allows for a more accurate assessment of model quality on the long tail of low-resource languages.
NLU: We evaluate various tasks to test different aspects of the model. These include XCOPA (Ponti et al., 2020), a multilingual common reasoning task supporting 11 languages; XStoryCloze (Lin et al., 2022a), a story completion task in 11 languages; XNLI (Conneau et al., 2018), a cross-lingual natural language inference task for 15 languages; PAWS-X (Yang et al., 2019), a paraphrase identification task in 7 languages; and MGSM (Shi et al., 2022), a mathematical reasoning task in 11 languages.
Besides fine-tuning the embedding layer, adjusting the lower layers can also be effective. To further investigate the functionality of the Embed FT strategy, we separately fine-tuned each layer of LLaMA using the Alpaca-En dataset and then tested on the Flores-101 en→ro devtest. The layers of the LLaMA model, excluding the embedding layer, are numbered from 0 to 31, with 0 being the closest to the embedding layer and 31 being the furthest. The bilingual performance of en→ro is illustrated in Table 16. Our experiments showed that fine-tuning the lower layers is just as effective as fine-tuning the embedding layer. However, we found that fine-tuning the higher layers did not produce satisfactory results.

I Used Scientific Artifacts

Below lists scientific artifacts that are used in our work. For the sake of ethic, our use of these artifacts is consistent with their intended use.

- *Stanford Alpaca (Apache-2.0 license)*, a project that aims to build and share an instruction-following LLaMA model.
- *Lego-MT (MIT license)*, a dataset for machine translation.
- *Transformers (Apache-2.0 license)*, a framework that provides thousands of pretrained models to perform tasks on different modalities such as text, vision, and audio.

Acknowledge

This work is partially supported by the National Key R&D Program of China (NO.2022ZD0160100).

Setting	en→ro	en→es	en→de	en→ca	en→pt	en→da	en→no	en→bs	AVG.
FT	27.1	23.5	24.5	34.3	40.5	**32.3**	**20.9**	**22.4**	28.2
LoRA	28.8	26.6	30.3	**36.6**	40.3	31.5	18.2	20.3	29.1
Embed FT	**29.1**	**26.8**	**31.0**	35.9	**41.1**	32.1	18.3	19.4	**29.2**
Layer 0	**29.2**	26.6	**30.9**	**37.2**	**41.5**	**32.4**	18.7	20.8	**29.7**
Layer 1	28.9	26.2	30.1	36.2	40.5	32.1	19.0	20.3	29.2
Layer 2	28.9	26.7	30.6	36.4	40.6	32.2	18.9	**21.4**	29.5
Layer 3	28.8	26.6	30.4	36.6	40.6	31.8	18.6	20.6	29.2
Layer 4	29.0	26.8	30.4	36.7	40.5	32.1	18.9	20.4	29.3
Layer 5	28.9	**27.0**	30.9	37.1	41.3	32.1	19.0	20.8	29.6
Layer 6	29.0	26.8	30.5	36.7	40.8	31.5	19.0	20.3	29.3
Layer 7	28.7	26.4	30.7	36.1	40.3	32.0	18.7	19.6	29.1
Layer 8	29.1	26.2	30.0	36.4	40.4	31.6	**19.2**	19.7	29.1
Layer 9	28.8	26.3	30.2	35.8	40.2	31.6	19.1	19.5	28.9
Layer 10	27.8	25.8	29.7	35.5	39.9	30.8	18.7	16.1	28.0
Layer 11	28.0	25.6	29.9	35.5	39.4	30.9	18.8	17.1	28.2
Layer 12	27.9	25.5	29.2	34.8	38.2	30.6	17.2	15.4	27.4
Layer 13	27.8	25.6	29.1	34.1	38.3	30.4	17.3	16.5	27.4
Layer 14	25.1	24.7	28.5	32.1	36.2	29.4	15.8	10.1	25.2
Layer 15	15.7	22.6	25.4	27.2	27.7	24.2	11.2	2.5	19.6
Layer 16	15.2	20.3	23.2	26.5	18.9	20.2	10.4	3.2	17.2
Layer 17	19.0	21.0	23.1	23.6	22.1	20.2	11.1	5.0	18.1
Layer 18	7.1	6.7	8.9	7.5	5.8	10.1	5.6	3.1	6.8
Layer 19	6.2	4.0	6.4	3.0	4.5	4.7	3.9	1.7	4.3
Layer 20	6.1	5.4	4.0	3.9	6.0	5.9	4.7	2.5	4.8
Layer 21	5.0	5.0	3.2	2.5	4.2	5.1	3.9	2.2	3.9
Layer 22	5.4	5.3	2.9	3.7	6.6	7.7	3.9	2.6	4.8
Layer 23	4.2	2.6	0.8	1.4	2.8	6.1	3.2	1.7	2.9
Layer 24	4.3	3.5	2.9	1.8	5.2	5.1	3.4	2.1	3.5
Layer 25	4.7	2.7	2.0	1.9	7.7	6.3	3.1	2.0	3.8
Layer 26	4.7	2.7	3.8	2.2	6.3	4.7	3.0	2.4	3.7
Layer 27	5.1	1.3	4.4	2.5	6.3	5.6	4.6	2.3	4.0
Layer 28	4.6	1.6	4.3	2.7	4.9	3.8	3.3	2.6	3.5
Layer 29	4.1	2.9	5.2	4.3	6.7	6.8	3.6	2.9	4.6
Layer 30	4.8	2.6	5.6	4.2	6.1	5.3	4.1	2.8	4.4
Layer 31	4.3	2.8	3.8	4.2	4.6	6.3	3.9	2.8	4.1

Table 14: Single-layer fine-tuning results on Alpaca-En dataset. The layers of the LLaMA-7B model, excluding the embedding layer, are numbered according to their distance from the embedding layer, with the closest being Layer 0 and the furthest being Layer 31. The term "+ Layer i" indicates that only the i th layer is fine-tuned, with the other parts of parameters fixed.

Size	en→da	en→ca	en→cs	en→bg	en→pl	en→es	en→fr	en→de	en→pt	en→ru	en→nl	AVG.
Bilingual Full Fine-Tuning												
10k	31.9	34.9	23.9	26.0	17.0	23.5	32.5	22.5	41.3	24.3	18.7	27.0
20k	32.2	35.8	24.5	26.5	18.4	23.8	31.7	24.8	41.1	24.2	18.9	27.4
40k	32.8	37.0	25.4	27.4	18.8	25.2	34.1	25.9	41.3	24.1	22.1	28.6
160k	36.4	39.2	27.1	31.8	19.7	25.9	39.1	31.2	39.7	24.6	24.3	30.8
Bilingual Embedding Fine-Tuning												
10k	26.4	30.1	16.6	19.6	12.6	23.7	34.7	23.1	33.3	19.1	21.2	23.7
20k	33.1	37.3	24.4	26.5	18.6	26.4	41.1	30.4	40.8	24.7	24.4	29.8
40k	33.9	36.9	25.5	27.3	19.5	26.7	39.7	28.3	40.7	25.4	22.6	29.7
160k	34.7	37.7	26.2	28.2	19.9	27.0	40.9	31.3	40.7	25.7	24.9	30.7
Bilingual Lower Layers [0-14] Fine-Tuning												
10k	33.4	36.2	25.6	27.1	18.4	24.2	32.8	23.1	42.1	25.5	18.5	27.9
20k	33.1	36.9	25.4	27.2	18.3	24.1	33.1	25.6	41.8	25.1	19.3	28.2
40k	33.9	37.8	25.6	27.5	19.2	25.7	34.8	25.8	41.2	25.3	21.7	29.0
160k	35.5	39.3	27.0	30.1	19.7	25.9	39.4	31.3	39.9	25.2	24.6	30.7

Table 15: The bilingual performance under different training strategies shows that fine-tuning the embedding layer performs as well as full fine-tuning in terms of bilingual performance. Interestingly, fine-tuning all lower layers does not yield additional gains.

Models	**XCOPA**	**MGSM**	**XStoryCloze**	**PAW-X**	**XNLI**	**Flores-101**	**AVG.**
Parrot-7B	54.2	3.7	56.1	56.5	39.0	25.2	46.9
LLaMA-7B	53.9	5.8	55.5	53.2	37.1	4.4	35.0
LLaMA-7B + Alpaca-En							
FT	**54.5**	4.5	**57.6**	57.1	**40.3**	28.2	**48.4**
LoRA	54.4	6.0	57.0	54.1	38.4	29.1	47.8
Embed FT	54.0	**6.2**	55.9	54.4	38.0	29.2	47.6
LLaMA-7B + Alpaca-X							
FT	54.4	4.9	57.2	**57.1**	40.2	28.0	48.4
LoRA	54.5	5.6	57.0	53.8	38.3	28.0	47.4
Embed FT	54.1	5.9	55.9	54.6	38.3	27.9	47.3
LLaMA-7B + Bilingual							
FT	53.9	3.4	55.6	55.9	38.8	30.1	47.6
LoRA	54.3	4.7	55.9	54.3	38.0	31.1	47.6
Embed FT	54.3	4.7	55.9	54.3	38.0	**31.4**	47.7

Table 16: Comparative analysis of training strategies. XCOPA, MGSM, XStoryCloze, PAW-X and XNLI are natural language understanding tasks, evaluated on all languages with accuracy metric; Flores-101 is an NLG task, each score in the cell represents an average spBLEU, encompassing bilingual translation performances from en→{ro, es, de, ca, pt, da, no, bs}. The experimental result reveals that Embed FT can perform as well as another strategy.