Parameter Efficient Approach for Zero-shot Cross Lingual Transfer

Manan Sharma
MLD, CMU
msharma2@andrew.cmu.edu

Annanya Chauhan
MLD, CMU
annanyac@andrew.cmu.edu

ABSTRACT

The paper proposes a novel parameter-efficient fine-tuning (PEFT) technique for the cross-lingual task transfer in the pretrained multilingual language models (MLLMs). Fine-tuning a subset of parameters of large pre-trained models has become the standard for transfer learning, and it is also a viable solution for improving the performance of language models in a task-specific setting. PEFT has been demonstrated to perform at par with the full-tinetuning and hence is a hotly researched topic in recent years. However, the leveraging of PEFT in the transfer learning of information across languages, known as *cross-lingual transfer*, has only started gaining attention recently. Previous work on multilingual language models emphasizes the performance gaps of MLLMs in languages other than English and often attributes this to the phenomenon of the curse of multilinguality. This also affects the use of PEFT for cross-lingual transfer tasks as the performance of the transfer techniques takes a hit. Our study finds that not all parameters are uniformly fine-tuned and their spread follows a set pattern. Leveraging this, we propose a parameter subset selection method that identifies the most relevant parameters while constraining the spread to pre-set distribution and only fine-tunes this set. Our best model, **SC-SFT**, results in performance gains of up to 7% in accuracy. We believe that our work provides a novel direction for future studies in model pruning.

1 INTRODUCTION

The advent of large-scale multilingual language models, such as multilingual BERT (mBERT) and XLM-R, has significantly improved accessibility in natural language processing (NLP). These models are capable of performing tasks in multiple languages by leveraging a shared sub-word vocabulary and joint training on multilingual corpora. However, despite their impressive performance, the scalability and efficiency of these models remain a challenge, especially for low-resource languages and languages unseen during pre-training. To address these issues, and given the ever increasing size of the Transformer [1] based models, recent research has been focused towards parameter-efficient techniques for transferring knowledge across languages with minimal number of extra trainable parameters while maintaining or improving performance.

We direct our attention towards the low-resource languages, languages which are often under-represented in the English-dominated field of NLP and hence suffer from the lack of accessibility to the recent technological advancements. To highlight the severity of the problem, [2] focuses on the challenges in applying NLP technology to the languages spoken in Indonesia, world's second most linguistically diverse country and also one of the most populous. The authors discuss the issues of limited resources, language diversity, writing standards, and limited access to computational resources and recommend better documentation, data-efficient and compute-efficient NLP methods, and greater engagement with local communities to overcome these challenges and improve the development of NLP technologies for Indonesian languages.

Specifically, this paper focuses on parameter-efficient cross-lingual transfer techniques in multilingual language models with a focus on low-resource languages. Specifically, we propose method that allow models to generalize across languages with minimal fine-tuning, reducing computational costs and memory requirements, and preferably in a few-shot or a zero-shot manner (to factor in the data scarcity of such languages). Our methods offer promising alternatives to full model fine-tuning by either freezing large portions of the model or introducing lightweight components that can be adapted to specific languages or tasks in a resource constrained setting.

2 RELATED WORKS

We briefly summarize our findings on parameter-efficient fine-tuning techniques, and their applications and extensions for cross-lingual task transfer problem for low-resource languages.

2.1 PEFT for Cross Lingual Task Transfer

Cross-lingual Task Transfer refers to transferring task knowledge from one language to the other using the Multilingual Language Models (MLLMs). These are particularly targeted for the low-resource languages for which the task-data is often scarce or entirely unavailable. These techniques often use a high-resource language like English to extract the task knowledge and then use transfer learning techniques to employ it for the low-resource language. Thus, the application of PEFT to cross-lingual transfer has gained traction due to its ability to handle the inherent complexity of multilingual models. Here, MLLMs like mBERT[3], XLM-Roberta[4] and mT5[5], are trained on data from multiple languages to generalize across them. [6] explores the multilingual skills of pre-trained language models (LMs) in few-shot learning for multi-class classification tasks. The authors use GPT and T5 models and evaluate their performance on English, French, German, and Spanish. The results show that pre-trained LMs can effectively predict not only English test samples but also non-English ones, given a few English examples as context. This highlights the potential of few-shot learning in addressing the low-resource issue in non-English languages. However, directly fine-tuning these models for specific tasks in low-resource languages often leads to suboptimal performance, as we shall describe in the next section.

Need to cater to low-resource languages: [7] propose a taxonomy to classify the world's languages into six categories based on their digital richness in terms of unlabeled and labeled resources.

The authors found that the availability of language resources is highly skewed, with a small number of languages dominating NLP research. They also note that a lot of the recent methods only require large unlabeled corpora across languages and labeled data in a few languages. The authors highlight the need to prioritize the development of resources and technologies for under-resourced languages to ensure inclusivity in the NLP world. To emphasize the non-triviaility of the problem, [8] showed that simply translating the data from high-resource languages to another target language has significant computational overhead and often leads to datasets of unreliable quality.

PEFT for Cross Lingual Transfer: PEFT methods occupy a large portion of approaches that cater to the cross-lingual transfer task in the low resource setting. This is due to the majority of the PEFT techniques being portable: where they learn a component of the model or a sub-network that can easily be transferred depending on the use case. These methods have shown promise in improving cross-lingual transfer by allowing models to retain their multilingual knowledge while specializing in new tasks or languages with minimal parameter updates. We broadly classify various approaches as:

- **Adapter-Based Methods**: Approaches like **MAD-X** [9] propose a modular framework where language-specific and task-specific adapters are learned separately and then composed during inference. **MAD-G** [10] improves on MAD-X by learning task agnostic language adapters whose parameters are obtained by a contextual parameter generator that is learnable by MLM. This generator takes as input the typological vector of the target language. This allows for parameter-efficient transfer across both seen and unseen languages without modifying the main model architecture in a portable manner. **mmT5** [11] improves seq-to-seq multilingual models by utilizing language-specific adapters while pre-training to minimize the language-specific hallucinations that might arise from increasing the language support of MLLM.
- **Subnetwork selection**: These techniques, like [12] leverage sparse masks to fine-tune only a small subset of parameters relevant to each task or language. This method is inspired by the Lottery Ticket Hypothesis and offers both modularity and expressivity without increasing inference time. Among the simplest of settings, BitFit[13] only fine-tunes the bias parameter throughout the model. We choose the latter as one of the baselines to compare with.
- **Monolingual-only Transfer**: These methods propose transferring monolingual representations, where the task information is extracted only through datasets in one language and sometimes even use a monolingual base model. The study by[14] suggests that deep monolingual models learn some abstractions that generalize across languages and compete with the corresponding multilingual models. They also propose XSQuAD, a benchmark dataset for cross-lingual classification transfer. [15] propose a layer-swapping method where they swapped the first and last few layers of the task-tuned model in English by the corresponding layers of language experts (models with same architecture trained on the language modeling task in the target language).

2.2 Challenges and Contributions

Despite the progress made with PEFT techniques for cross-lingual transfer, several challenges remain:

- **Model Capacity**: Multilingual models often struggle with limited capacity when scaling across many languages, especially low-resource ones. The trade-off between language coverage and model performance is a persistent issue. [16] conducts experiments which indicate that adding multilingual data during pre-training begins to hurt performance for both low-resource and high-resource languages, likely due to limited model capacity (phenomenon also known as "curse of multilinguality").
- **Interference among Languages**: When adapting models for multiple tasks or languages, there is a risk of interference where knowledge from one language negatively impacts performance on another. This might especially happen for distantly related or totally unrelated languages. [17] argue that while Massively Multilingual Transformers (MMTs) have achieved good performance in zero-shot transferring to languages similar to English or with large monolingual corpora, they perform poorly when transferring to distant or low-resource languages and notes that few-shot transfer, is more effective across the board.
- **Efficiency vs. Performance**: While PEFT methods reduce computational costs, striking a balance between efficiency and performance remains challenging. Sparse fine-tuning methods must carefully select which parameters to update without sacrificing accuracy.
- **Initial training overhead for LT-SFT method**: The method requires a initial training phase for parameter selection. This is computationally costly and limits it's application for fine-tuning of really large language models. However the authors have tried addressing it in their follow-up work[18].

To this end, we propose a method that is efficient and ensures fewer interferences between tasks and languages by design. It is also extremely modular and also provides insight into the model framework. In summary, we highlight following contributions in this paper:

(1) We propose a novel sparse fine tuning strategy that helps in easily capturing the distribution of the beneficial parameters for the downstream task and language.
(2) We also compare our approach with a genralized soft masking strategy and extend the work by [19] for cross-lingual transfer.

3 LOTTERY TICKET HYPOTHESIS BASED PARAMETER SELECTION

3.1 LT-SFT

We begin by describing the work of LT-SFT[20], which leverages the Lottery Ticket Hypothesis[21]: dense, randomly-initialized, feed-forward networks contain subnetworks ("winning tickets") that -

when trained in isolation - reach test accuracy comparable to the original network in a similar number of iterations. LT-SFT tries to identify a winning ticket by choosing the parameters with highest absolute value difference in their pre-trained and fully-fine-tuned value. The network is then reset and only these parameters are fine-tuned. For zero-shot transfer, their method employs two SFTs: a) Language SFT: trained on MLM task on target language corpus; and b) Task SFT: trained on the task loss, on a dataset in the source languages. The SFTs are simply added on top of the pre-trained weights of the model during the zero-shot inference on task data in the target language:

$$\theta_{task,t} = \theta_0 + SFT_{lang,t} + SFT_{task,s}$$

We refer the reader to Appendix A of [12] for the formal training algorithm. In this work, we shall base our approaches on the LT-SFT framework.

3.2 Need for Controlled Sparsity

LT-SFT chooses parameters from all the layers of the model, and thus the identified SFT can have arbitrary parameter distribution across its layers. Since different layers of a model correspond to the different aspects and fidelity of the input information, we hypothesize that a lottery ticket for a specific {task, language} set would have a distribution that mimics the shape of the layer-wise distribution of parameters that correspond the best with the set.

To demonstrate this, we analyzed the parameter distribution of the learned SFTs in the LT-SFT approach. The language SFTs for all the languages(Figure 1) follow as U-shaped distribution, indicating that the initial and final layers are more important for MLM tasks in all languages, which was also the task on which the model was pre-trained. The task SFTs for the tasks of Natural Language Inference (NLI) and Sentiment Analysis (SA) have distributions with inverted patterns(Figure 2), where middle layers have larger number of parameters as against the other layers. To this end we propose the first method for controlled sparsity:

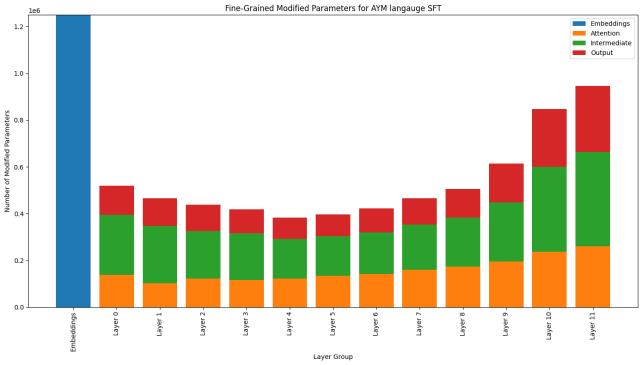

Figure 1: Layer-wise distribution of modified/trained parameters for the Aymara language SFT model. The graph further shows the fine-grained distribution of modified parameters between attention and non-attention components in a RoBERTa layer.

3.3 Sparsity Control with Distribution Forcing

We hypothesize that the final parameter-distribution we observe is critical in determining the performance of SFTs. Thus we want to modify the learning problem to search only for lottery tickets of specific shape.

Inspired by the observed distribution, we explicitly enforce the parameters of SFTs to follow a set sparsity distribution throughout the training while keeping the overall sparsity ratio constant. Thus, while original LT-SFT eventually converges to the observed parameter distribution, we constrain each step of training to follow the observed distributional constraint. This is equivalent to having layer-wise mini-SFTs in union.

To formalize this, denote the original SFT learning problem as $SFT_{[L],K}(t)$ where $[L]$ denotes the set of all L layers of the model where we allow the parameters of SFT to be selected from; K denotes our parameter budget: the number of parameters we are allowed to select for the SFT; and t denotes the task. In our approach we break the SFT learning problem by introducing layer-wise mini-SFTs jointly learnt by the model. If we constrain the SFT to lie in a subset of layers $\mathcal{L} \subseteq L$:

$$\text{Learn} \left(\bigcup_{i:L_i \in \mathcal{L}} SFT_{L_i,K_i} \right)(t); \qquad \text{s.t.} \sum_i K_i = K$$

We call this approach of layer-wise mini-SFTs as **Sparsity Controlled SFT (SC-SFT)**. This can be seen as forcing a histogram-based distribution that is admissible w.r.t. the original parameters, since the total sparsity budget of the mini-SFT layers.

It can be noted that the learning problem of SC-SFT is a strict subset of the original SFT learning problem since LT-SFT can have arbitrarily any K-distribution of parameters in theory. Thus, SC-SFT can be seen as a *regularized* version of LT-SFT.

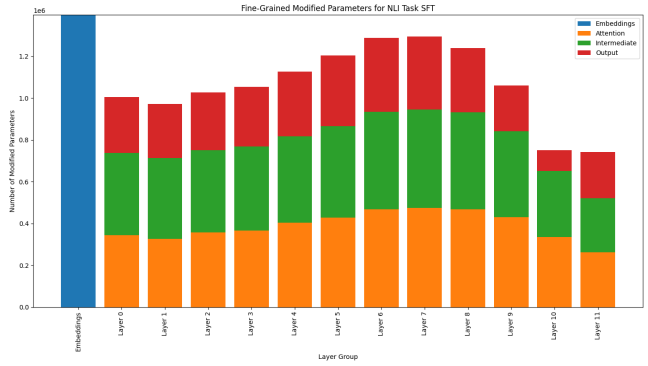

Figure 2: Layer-wise distribution of modified/trained parameters for the NLI task SFT model. The graph further shows the fine-grained distribution of modified parameters between attention and non-attention parts in a RoBERTa layer.

3.4 Generalized Adaptive Formulation

Both LT-SFT and SC-SFT algorithms can be seen as a mask learning problem where the model first learns a mask and fine-tunes after applying the mask. The masks learned by SFTs are hard (binary). One can further generalize this problem by learning a soft mask, where the model first learns a soft mask during the fine-tuning stage; and the final learned mask is fixed and applied to the pre-trained model again while fine-tuning the parameters.

Formally, for a parameter matrix $\mathbf{W} \in \mathbb{R}^{n \times m}$, we learn a mask $\mathbf{M} \in [0,1]^{n \times m}$ as:

$$\mathbf{S} \in \mathbb{R}^{n \times m}; \quad \mathbf{M} = \sigma(\mathbf{S}); \quad \mathbf{W} = \mathbf{W} \odot \mathbf{M}$$

Where $\mathbf{S}$ is the learnable parameter that maps to the mask and $\sigma : \mathbb{R} \to [0,1]$ is an element-wise map, like the sigmoid. This can potentially give the model more control over what parameters to "focus on" at each training step. Thus, we first replace the model parameters with replace model operations for $\mathbf{W}$ as $\mathbf{W} \odot \mathbf{M}$ followed by a two-step learning procedure that looks like:

- Fixed the parameters $\mathbf{W}$, only learn $\mathbf{M}$.
- Fix $\mathbf{M}$, learn $\mathbf{W}$.

We call the most basic version of the approach **Ada-SFT** and do a brief comparison with SC-SFT in one of the task. We note that while the performance seems to be poorer than the baselines, we believe further study and implementation fine-tuning should help in finding even better models. Note that Ada-SFT differs from LT-SFT in first, mask-selection step, in a way that the mask directly gets signals from the downstream task gradients.

3.5 Random Masking to find Winning Tickets

Following the work of [19], where the authors claim that finding random winning tickets is an easier problem, with performance often comparable to methods like LoRA. We extend their work in our cross-lingual transfer setting to assess the presence of winning tickets in the the joint task and linguistic transfer setting. Specifically, we pick out random parameters to constitute the SFT and then fine-tune the selected subset of params. We call this approach as **Rand-SFT**. Formally:

- Select $\mathbf{M} \in \{0,1\}^{n \times m}$ such that $\mathbf{M}_{i,j} \sim Bernuolli(p)$.
- Apply while freezing $\mathbf{M}$ to $\mathbf{W}$ while fine-tuning the latter.

Here, $1 - p$ denotes the sparsity percentage (fraction of parameters that will be masked).

4 EXPERIMENTS AND RESULTS

4.1 Setup

We evaluated our proposal using two tasks: Natural Language Inference (NLI) and Sentiment Analysis (SA). Our choice was motivated by the relative difficulty gap, with NLI being the relatively harder task to perform. In all our variants, we kept the parameter budget K as 14.1 million. The base model we used was XLM-Roberta-base[4], which has around 276 million parameters. This leads to a sparsity of 5.1% for the task SFTs. We used the default learning rate of 2e-5 to train the task SFTs. Also, for language SFTs, we use the author-provided parameters as they did not release their version of low-resource languages' raw text data.

4.1.1 Datasets. For training task-specific SFTs and language-specific SFTs, we utilized the following datasets:

- For NLI, we use the **AmericasNLI** [22] dataset for evaluation, that contains NLI examples in 10 South American languages (low-resource). For learning the task SFT, we use the MultiNLI[23] dataset (English).
- For SA evaluation, we use NUSAX-senti [24] sentiment analysis dataset of low-resource Indonesian languages. For task SFT, we use the SMSA SA dataset in English and its translation in some Indonesian languages.
- For all the low-resource language SFTs, we use the Wikipedia corpora that the authors use in creating their corpus. We did not have access to their exact language corpus as it was not made public, but we try to include majority of their sources they employ in the creation. The authors also note that due to extremely low or no Wiki corpus for certain languages, the language SFTs learned is likely poor.

4.1.2 Baseline. To construct our baselines, we reproduced the results from the LT-SFT paper [12] on the two tasks. The task SFT model using the code directly from the SFT paper's GitHub repository [20]. These results were generated by applying the SFT module in the author-provided language and the SFT module for trained tasks to the base model and then generating predictions for both tasks. Similar to LT-SFT, we compare the NLI tasks through two baselines: BitFit and Mad-X.

4.1.3 Variants for the Natural Language Inference task. For the NLI task, we experimented with multiple configurations of SC-SFT to train the task SFT. Our best-performing sparsity configuration was:

$$\frac{K_{\text{embedding}}}{K} = \frac{K_{11}}{K} = \frac{K_{10}}{K} = 0.02, \quad \frac{\sum_{i=0}^{9} K_i}{K} = 0.94$$

Furthermore, we also explored the Ada-SFT approach for training the task-specific SFT.

4.1.4 Variants for the Sentiment Analysis Task. For the Sentiment Analysis (SA) task, we experimented with multiple variants of SC-SFT to train the task-specific SFT. Our best-performing sparsity configuration was:

$$\frac{K_{\text{embedding}}}{K} = 0.005, \quad \frac{K_{11}}{K} = \frac{K_{10}}{K} = 0.02, \quad \frac{\sum_{i=0}^{9} K_i}{K} = 0.955$$

Additionally, we employed the Rand-SFT approach to train three task-specific SFT variants: the first utilizes the full 14.1 million parameter budget; the second is similar to the first but with the embedding parameters frozen; and the third is a significantly smaller variant, tuning only 141k parameters (a 100x reduction in parameter count) while using a higher learning rate (2e-3 instead of 2e-5). This smaller variant was inspired by [19], where the authors demonstrated that extremely small SFTs can achieve performance comparable to larger SFTs when trained with a higher learning rate.

4.1.5 Device Specifications. We used A100 and RTX 4090 GPUs to train our task SFT models in all our experiments.

Language (Code)	LT-SFT	SC-SFT	Ada-SFT	MAD-X	BitFit
Aymara (aym)	**0.5747**	0.5653	0.5517	0.516	0.408
Asháninka (cni)	0.4600	**0.4947**	0.4694	0.4760	0.345
Bribri (bzd)	0.424	**0.448**	0.409	0.4400	0.367
Guarani (gn)	0.6280	**0.6373**	0.6086	0.5880	0.464
Náhuatl (nah)	0.5108	0.5176	0.5191	**0.537**	0.388
Otomí (oto)	0.3930	**0.4693**	0.4182	0.4680	0.398
Quechua (quy)	0.6093	0.6053	**0.6106**	0.5830	0.345
Rarámuri (tar)	0.4173	0.4307	0.4098	**0.4390**	0.367
Shipibo-Konibo (shp)	**0.4960**	0.4840	0.4764	0.4890	0.388
Wixárika (hch)	0.4053	**0.4493**	0.3982	0.415	0.363
Average	0.4918	**0.5102**	0.4870	0.495	0.383

Table 1: Comparison of Accuracy Scores for Natural Language Inference of American Languages on the AmericasNLI dataset. The numbers of MAD-X and Bitfit are taken from the LT-SFT paper.

Language (Code)	LT-SFT	SC-SFT	Rand-SFT
Acehnese (ace)	0.7839	**0.7852**	0.7821
Balinese (ban)	0.8308	**0.8344**	0.8044
Banjarese (bjn)	**0.8248**	0.811	0.8012
Madurese (mad)	**0.7921**	0.7851	0.6655
Minangkabau (min)	0.8353	**0.8378**	0.7980
Javanese (jav)	0.8427	0.8376	**0.8507**
Sundanese (sun)	**0.8500**	0.8493	0.8364
Average	**0.8228**	0.8201	0.7912

Table 2: Comparison of F1 Scores for Sentiment Analysis Evaluation of Indonesian Languages on the NUSAX-senti dataset

4.2 Results and Discussion

For the NLI task we report the performance of our models in Table 1. Our SC-SFT variant outperformed the LT-SFT baseline in 7 out of the 10 American languages. In particular, the languages in which the gains were observed were all resource-challenged. We hypothesize that task SFT dominate over the language SFT for these languages and hence the results were pronounced. The greatest accuracy gain was observed in Otomí, with an absolute increase of 7%. Overall, our approach improved the average accuracy from 49% to 51%.

For the sentiment analysis task (Table 2), the results were competetive (very minor improvement in 5/10). Our SC-SFT variant achieved comparable performance to the baseline, with an accuracy of 0.820, almost similar to the baseline's 0.822. The authors of LT-SFT point out that the dataset they employed to build the language SFTs were severely limited. Thus we believe that a very poor language SFT does not provide enough signals for accurate prediction on the task dataset in target language and only acts as a confounder. However, our other variants performed worse than the baseline.

Interestingly, the extremely low-parameter Rand-SFT variant delivered results nearly equivalent to the larger Rand-SFT variant, except for the Madurese language. This was achieved despite having a 100x smaller parameter budget.

We also did a quick experiment with the general AdaSFT framework which is not parameter efficient but still performs comparably. The Ada-SFT was only trained and performed for the linear layers as they have the most parameters. The results are depicted in table 1.

5 ANALYSIS

In this section, our aim is to assess the model behavior by modifying or ablating various components of the proposed models.

5.1 SFT w/o Embedding Matrix

Learnable embedding matrices have appreciable number of trainable parameters compared to the overall model. As evident in figures 1 and 2, embedding matrices have the largest fraction of SFT parameters compared to all other layers. Here we hypothesize that *embedding matrices are relatively less crucial for the task transfer task* than other layers' parameters.

To assess this, we froze the embedding matrix's parameters for Rand-SFT and observe that it performs better than the normal Rand-SFT and report the result in Table 3.

5.2 Promising Low-Parameter Regimes

As shown in Table 3, the extremely low-parameter Rand-SFT variant achieved results nearly equivalent to the larger Rand-SFT variant, with the exception of the Madurese language. This performance was achieved despite having a 100x smaller parameter budget. This could potentially encourage further research in the development of an extremely parameter-efficient variant of LT-SFT, where, after an initial fine-tuning phase, a very small subset of parameters is selected and trained using a much higher learning rate.

5.3 Tuning the Language SFT

Our experiments have so far focused on task SFT while retaining the LT-SFT formulation for language SFT. However, we did not have access to the full dataset that the authors trained on.

Language (Code)	Rand-SFT	Rand-SFT (excluding embeddings)	Rand-SFT (100x Low Params)
Acehnese (ace)	0.7821	0.7675	0.7028
Balinese (ban)	0.8044	0.8147	0.7732
Banjarese (bjn)	0.8012	0.7905	0.8068
Madurese (mad)	0.6655	0.7794	0.4873
Minangkabau (min)	0.798	0.7983	0.811
Javanese (jav)	0.8507	0.8436	0.8441
Sundanese (sun)	0.8364	0.8489	0.8356
Average	0.7912	**0.8061**	0.7515

Table 3: F1 scores with and without SFT parameters in embedding layer. On SA task on NUSAX-SA. 100x-low setting refers to 100x less parameters than our standard setting.

We created our own datasets using the datasets referenced by the authors of LT-SFT and built our own dataset to train the language SFTs. We first note that the training corpora are very small for the listed South American languages. The problem was exacerbated for Indonesian languages where we were not able to find any Wiki dataset. Hence we only focus on the NLI task for this section.

We tested our SC-SFT language SFT in five languages with least corpora sizes, since these were among the languages that we observed improvement on task SFT. We observe slightly poorer performance by using the language SC-SFT as evident in Table 4. We hypothesize that since the pre-trained model was also trained on the masked language-modeling task, further training on the restricted parameter space might cause the model to overfit to a poorly performing SFT, which do not generalize well when correlating with the task information during the inference time and hence acts as a confounder.

Language	Lang+Task LT-SFT	Lang + Task SC-SFT
aym	0.58	0.572
bzd	0.432	0.4053
cni	0.472	0.4067
hch	0.3973	0.4027
oto	0.4358	0.4037
Average	**0.463**	0.4381

Table 4: Comparison of Accuracy on custom datasets for South American languages, where Lang LT-SFT and Lang SC-SFT are trained and combined with Task LT-SFT and Task SC-SFT, respectively

6 PROPOSALS & FUTURE DIRECTIONS

Our survey and analysis indicate various scopes of improvement across various aspects of the modelling stages for a model. Hence, our goal is to improve the LT-SFT algorithm by pursuing one or more of the following directions:

- **Improving Cross-Lingual Alignment**: An underlying assumption in LT-SFT and most cross lingual transfer techniques is that the learned representation in different languages are aligned: there is meaningful similarity of multilingual representations across languages. Following various techniques that test representational alignment of MLLMs that are summarized in [25], we plan to conduct an experiment, where we shall first obtain the representation of same sentence in English translated in multiple languages from the learnt LT-SFT model, and observe the similarity score between the similar sentence and dis-similar sentence. This would indicate if model is indeed confounded amongst the languages. If so, we can pursue our idea to employ contrastive learning based techniques for debiasing the language SFTs.
  A very interesting analysis by [26] hypothesizes that different languages occupy different linear subspaces in the the shared representation spaces generated by the pre-training of MLLMs. We think we can use their analysis to derive certain representation specific regularization terms for language SFT training such that it aids the model to learn more richer representations of the low resource languages by marking similarities across the languages that similar features with the target language.
- **Adaptive pruning of SFTs**: The parameter selection strategy for SFT in [12] may miss out hierarchical information across tasks or languages (depending on overlap in the parameter sets of the SFTs obtained, which can be erroneous if SFT isn't a true representative of underlying task). This could be a refinement of our idea of Ada-SFT where we learn different masks/drop probabilities for each transformer block (this can go even more fine grained). Similar to [27], one can also create block specific masks OR an MLP takes input the pretrained values and post-fine-tuned values, and returns a binary mask. This is even easier in case of low rank matrices. This then ensures a loss-driver learning of masks (and in turn the SFTs) which might lead to learning more robust and better performing sub-networks.

7 CONCLUSION

In this work we propose a novel unstructured masking strategy for cross-lingual task transfer where sparsity is controllable and setting it to a some set patterns results in performance improvements. We supplement our study by comparing our method with extending a previous work demonstrating efficacy of using random lottery tickets.Our observations show promising results and outlines a novel direction for analysing the parameter space of large language model. While a high-data regime certainly helps in improving the models, more careful and sophisticated approaches are required for handling the low-resource regime, especially when dealing with endangered languages with minimal representation on the web. We hope our work serves as a next step in the path towards making language models more accessible to wider communities across the world.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023. [Online]. Available: https://arxiv.org/abs/1706.03762

[2] A. F. Aji, G. I. Winata, F. Koto, S. Cahyawijaya, A. Romadhony, R. Mahendra, K. Kurniawan, D. Moeljadi, R. E. Prasojo, T. Baldwin, J. H. Lau, and S. Ruder, "One country, 700+ languages: Nlp challenges for underrepresented languages and dialects in indonesia," 2022. [Online]. Available: https://arxiv.org/abs/2203.13357

[3] J. Devlin, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[4] A. Conneau, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2019.

[5] L. Xue, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[6] G. I. Winata, A. Madotto, Z. Lin, R. Liu, J. Yosinski, and P. Fung, "Language models are few-shot multilingual learners," 2021. [Online]. Available: https://arxiv.org/abs/2109.07684

[7] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury, "The state and fate of linguistic diversity and inclusion in the nlp world," 2021. [Online]. Available: https://arxiv.org/abs/2004.09095

[8] S. Khanuja, S. Gowriraj, L. Dery, and G. Neubig, "DeMuX: Data-efficient multilingual learning," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 7423–7436. [Online]. Available: https://aclanthology.org/2024.naacl-long.412

[9] J. Pfeiffer, I. Vulić, I. Gurevych, and S. Ruder, "MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 7654–7673. [Online]. Available: https://aclanthology.org/2020.emnlp-main.617

[10] A. Ansell, E. M. Ponti, J. Pfeiffer, S. Ruder, G. Glavaš, I. Vulić, and A. Korhonen, "MAD-G: Multilingual adapter generation for efficient cross-lingual transfer," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 4762–4781. [Online]. Available: https://aclanthology.org/2021.findings-emnlp.410

[11] J. Pfeiffer, F. Piccinno, M. Nicosia, X. Wang, M. Reid, and S. Ruder, "mmt5: Modular multilingual pre-training solves source language hallucinations," 2023. [Online]. Available: https://arxiv.org/abs/2305.14224

[12] A. Ansell, E. M. Ponti, A. Korhonen, and I. Vulić, "Composable sparse fine-tuning for cross-lingual transfer," 2023. [Online]. Available: https://arxiv.org/abs/2110.07560

[13] E. B. Zaken, S. Ravfogel, and Y. Goldberg, "Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models," 2022. [Online]. Available: https://arxiv.org/abs/2106.10199

[14] M. Artetxe, S. Ruder, and D. Yogatama, "On the cross-lingual transferability of monolingual representations," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020. [Online]. Available: http://dx.doi.org/10.18653/v1/2020.acl-main.421

[15] L. Bandarkar, B. Muller, P. Yuvraj, R. Hou, N. Singhal, H. Lv, and B. Liu, "Layer swapping for zero-shot cross-lingual transfer in large language models," 2024. [Online]. Available: https://arxiv.org/abs/2410.01335

[16] T. A. Chang, C. Arnett, Z. Tu, and B. K. Bergen, "When is multilinguality a curse? language modeling for 250 high- and low-resource languages," 2023. [Online]. Available: https://arxiv.org/abs/2311.09205

[17] A. Lauscher, V. Ravishankar, I. Vulić, and G. Glavaš, "From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 4483–4499. [Online]. Available: https://aclanthology.org/2020.emnlp-main.363

[18] A. Ansell, I. Vulić, H. Sterz, A. Korhonen, and E. M. Ponti, "Scaling sparse fine-tuning to large language models," *arXiv preprint arXiv:2401.16405*, 2024.

[19] J. Xu and J. Zhang, "Random masking finds winning tickets for parameter efficient fine-tuning," 2024. [Online]. Available: https://arxiv.org/abs/2405.02596

[20] A. Ansell, "Composable sparse fine-tuning for cross lingual transfer code," https://github.com/cambridgeltl/composable-sft/.

[21] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," *arXiv preprint arXiv:1803.03635*, 2018.

[22] A. Ebrahimi, M. Mager, A. Oncevay, V. Chaudhary, L. Chiruzzo, A. Fan, J. Ortega, R. Ramos, A. Rios, I. Meza-Ruiz *et al.*, "Americasnli: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages," *arXiv preprint arXiv:2104.08726*, 2021.

[23] A. Williams, N. Nangia, and S. R. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," *arXiv preprint arXiv:1704.05426*, 2017.

[24] G. I. Winata, A. F. Aji, S. Cahyawijaya, R. Mahendra, F. Koto, A. Romadhony, K. Kurniawan, D. Moeljadi, R. E. Prasojo, P. Fung *et al.*, "Nusax: Multilingual parallel sentiment dataset for 10 indonesian local languages," *arXiv preprint arXiv:2205.15960*, 2022.

[25] K. Hämmerl, J. Libovický, and A. Fraser, "Understanding cross-lingual alignment – a survey," 2024. [Online]. Available: https://arxiv.org/abs/2404.06228

[26] T. A. Chang, Z. Tu, and B. K. Bergen, "The geometry of multilingual language model representations," 2022. [Online]. Available: https://arxiv.org/abs/2205.10964

[27] C. Zheng, B. Zong, W. Cheng, D. Song, J. Ni, W. Yu, H. Chen, and W. Wang, "Robust graph representation learning via neural sparsification," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 11 458–11 468. [Online]. Available: https://proceedings.mlr.press/v119/zheng20d.html