KS-Lottery: Finding Certified Lottery Tickets for Multilingual Transfer in Large Language Models

Fei Yuan[1], **Chang Ma**[2], **Shuai Yuan**[1], **Qiushi Sun**[2], **Lei Li**[3]
[1] Shanghai Artificial Intelligence Laboratory
[2] The University of Hong Kong, [3] Carnegie Mellon University
yuanfei@pjlab.org.cn, cma@cs.hku.hk
syuanaf@connect.ust.hk, qiushisun@u.nus.edu, leili@cs.cmu.edu

Abstract

The lottery ticket hypothesis posits the existence of "winning tickets" within a randomly initialized neural network. Do winning tickets exist for LLMs in fine-tuning scenarios? How can we find such winning tickets? In this paper, we propose KS-Lottery, a method to identify a small subset of LLM parameters highly effective in multilingual fine-tuning. Our key idea is to use Kolmogorov-Smirnov Test to analyze the distribution shift of parameters before and after fine-tuning. We further theoretically prove that KS-Lottery can find the certified winning tickets in the embedding layer, fine-tuning on the found parameters is guaranteed to perform as well as full fine-tuning. Comparing KS-Lottery with other tuning algorithms on translation tasks, the experimental results show that KS-Lottery finds a much smaller set of parameters for fine-tuning while achieving the comparable performance as full fine-tuning LLM. Surprisingly, we find that fine-tuning 18 tokens' embedding of LLaMA suffices to reach the fine-tuning translation performance [1].

1 Introduction

Can we find an ultra-small subset of a well-trained Large Language Model (LLM; Touvron et al., 2023a,b; OpenAI, 2023; Chowdhery et al., 2022) such that fine-tuning these few parameters suffices to achieve the same performance as full tuning? The lottery tickets hypothesis (Frankle and Carbin, 2019) states that a small subnetwork (less than 10-20% of the whole model size), referred to as "winning tickets", in a large, randomly initialized neural network can achieve comparable performance to the original network with the same amount of training. Yet, the existence of winning tickets is not investigated for fine-tuning scenarios. Prior work (Aghajanyan et al., 2021) presents evidence that there are a small number of additional parameters corresponding to an intrinsic dimension (Li et al., 2018) on which fine-tuning leads to good performance. However, it remains an unsolved challenge to uncover such a small subset of fine-tuning efficient parameters within the *original* model.

Figure 1: Existence of "winning tickets". KS-Lottery identifies a small subset of embedding parameters of LLaMA-7B to maintain the translation performance of en→ca on Flores-101.

In this paper, we show that there exist key parameters - winning tickets, for transferring LLM to multiple new languages. As shown in Figure 1 (KS-Lottery), we found that as tuning as few as 18 ($18/32000 = 0.0006$) token embeddings of a well-trained LLM could achieve test performance comparable to full tuning on machine translation tasks. Based on the observation, we state the lottery ticket hypothesis for multilingual fine-tuning.

Generally, the fine-tuning process can be represented by a transition in M parameters from $\boldsymbol{\theta} = [\theta_0, \theta_1, \cdots, \theta_M]$ to $\widetilde{\boldsymbol{\theta}} = [\widetilde{\theta}_0, \widetilde{\theta}_1, \cdots, \widetilde{\theta}_M]$, where $\boldsymbol{\theta}$ and $\widetilde{\boldsymbol{\theta}}$ denote the sets of parameters that characterize the LLM $f(\cdot, \boldsymbol{\theta})$ before and after the fine-tuning process, respectively. Typically, the entire set of parameters $\boldsymbol{\theta}$ is adjusted during fine-tuning to enhance the model's ability to represent and learn the new tasks more effectively.

[1] https://github.com/CONE-MT/KS-Lottery.git

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 9077–9090
April 29 - May 4, 2025

The Fine-Tuning Lottery Ticket Hypothesis. n*A pre-trained neural network contains a **small** subset of parameters* ($\widetilde{\boldsymbol{\theta}}^D = [\widetilde{\theta}_0, \widetilde{\theta}_1, \cdots, \widetilde{\theta}_D, \theta_{D+1}, \cdots, \theta_M]$*, where* $D \ll M$*) that is initialized such that—when fine-tuned in isolation—it can match the performance of full tuning.*

In this work, we examine the fine-tuning lottery ticket hypothesis on LLMs in a multilingual transfer scenario, and explore the following inquiries:

- ***Existence of Winning Tickets:*** Is it certain that every LLM in multilingual transfer encompasses a compact subset of winning tickets? And how to quickly identify the winning tickets?
- ***Efficiency of Winning Tickets:*** How minimal can this subset be in terms of size?
- ***Interpretability of Winning Tickets:*** Do these winning tickets reflect the unique architectural characteristics of the multilingual LLM?

Identifying Winning Tickets. We propose KS-Lottery, a method to identify a winning ticket by merely fine-tuning the embedding layer of an LLM. And then fine-tune these identified tickets, keeping the remaining parameters frozen. The whole KS-Lottery consists of three steps:

1. Fine-tuning the embedding layer of an LLM $f(\cdot, \boldsymbol{\theta})$ to obtain $f(\cdot, \widetilde{\boldsymbol{\theta}}^D)$.
2. Run Kolmogorov-Smirnov Test between $\boldsymbol{\theta}$ and $\widetilde{\boldsymbol{\theta}}^D$ to select the winning ticket $\widetilde{\boldsymbol{\theta}}^D_a$.
3. Tuning the $\widetilde{\boldsymbol{\theta}}^D_a$ within $\boldsymbol{\theta}$ on downstream tasks.

The core idea of our method is to heuristically identify parameters with large distribution changes before and after fine-tuning. Here we use Kolmogorov-Smirnov Test to determine whether two sample variables stem from the same underlying distribution. Also, we simplify and accelerate the Kolmogorov-Smirnov Test process by focusing on the embedding layer, which constitutes the major change in parameters, due to the inductive bias of multilingual tasks. To this end, our approach, KS-Lottery, is a surprisingly straightforward yet effective technique for pinpointing winning tickets in LLMs. The Kolmogorov-Smirnov Test has the advantage of not assume the distribution, which is particularly useful when the data dows not conform to a normal distribution. Furthermore, a theoretical framework is developed to certify the effectiveness of our method. Inspired by randomized smoothing techniques (Zhao et al., 2021, 2022), we illustrate that parameters with Kolmogorov-Smirnov distance bounded by a small value before and after fine-tuning do not impact prediction. This provides a way for giving a certified lower bound for the performance of partial tuning on winning tickets. Our analysis also proves that KS-Lottery can find a small set of winning tickets when the original prediction model shows little uncertainty. This gives us a strong foundation for asserting that KS-Lottery can be an effective tool for finding winning tickets.

- We propose KS-Lottery, a method to identify winning tickets – an ultra-small subset of parameters that are sufficient to fine-tune on to achieve that of Full Tuning.
- Theoretically, we prove that KS-Lottery finds certified winning tickets.
- Empirically, we demonstrate that fine-tuning as few as 18 identified winning tickets (the token embedding) of LLaMA-7B using en→ca data achieves surprisingly good performance in translation tasks. This will result in a new standard in multilingual transfer of LLMs.

2 Related Work

Lottey Tickets Hypothesis. The Lottery Tickets Hypothesis suggests the presence of 'winning tickets' or beneficial subnetworks within a model(Frankle and Carbin, 2018; Malach et al., 2020). These subnetworks, discovered during pruning, are believed to be specifically suited to the learning task (Frankle and Carbin, 2018). In relation to this, Zheng et al. (2022) fins that these 'winning tickets' are more sparse in the later layers of the BERT model on GLUE tasks (Wang et al., 2018). While much research focuses on model pruning, there's also work on efficient parameter tuning (Ding et al., 2023). For example, it's shown that the performance of fine-tuned parameters can indicate both the task's inductive biases and the model's inherent structure (Ding et al., 2023).

Certified Methods for Transfer Learning. Certification is crucial in transfer learning, aiming to measure a model's generalization and capabilities (Raghunathan et al., 2018; Jia et al., 2019). Research has introduced certified robustness accuracy as a defense against adversarial attack (Raghunathan et al., 2018; Jia et al., 2019; Muravev and Petiushko, 2021; Zhao et al., 2022; Lecuyer et al., 2019). Randomized smoothing, a model-independent certification technique, assesses how input changes affect predictions (Lecuyer et al., 2019; Muravev and Petiushko, 2021). Our approach focuses on model parameter variations. Other studies have certified fairness in models (Ruoss et al., 2020; Peychev et al., 2021) and robustness against data selection (Wang et al., 2023).

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, *inter alia*) excel in English but underperform in other language. Studies have fine-tuned LLMs using monolingual or multilingual data to enhance their multilingual capabilities (Zhu et al., 2023; Li et al., 2023; Jiao et al., 2023; Cui et al., 2023; Yang et al., 2023). Embedding Tuning can activate multilingual abilities in certain languages, suggesting that the intrinsic dimension of these abilities may lie within the embedding layer (Li et al., 2018; Yuan et al., 2023b). The intrinsic dimension, the minimum parameters needed for a specific objective function, is estimated using heuristic methods and random subspace training due to computational constraints (Li et al., 2018; Aghajanyan et al., 2021).

3 Certified Winning Tickets via KS-Lottery

In this section, we first introduce KS-Lottery (Fasano and Franceschini, 1987) (§ 3.1). Then we apply KS-Lottery to find certifiable winning tickets in a multilingual transfer scenario (§ 3.2). Finally, the theoretical result of § 3.2, as well as the experimental result (§ 3.3), validate the effectiveness of KS-Lottery.

3.1 KS-Lottery

Guided by the hypothesis that parameters undergoing substantial changes during fine-tuning are crucial for making predictions (Levin et al., 2022), we exam the distribution p_i of each parameter $\boldsymbol{\theta}_i$ (with the parameter θ_{ij} of token j drawn from p_i) before and after the fine-tuning. Although various metrics exist for pinpointing essential parameters (Li et al., 2016; Dalvi et al., 2019; Meng et al., 2022), these often depend on specific cutoff values, and determining the necessary number of parameters in advance is challenging. We advocate for a new approach: *actively seeking out "lottery tickets" that are guaranteed to achieve similar fine-tuning outcomes with a high level of confidence.* This calls for a more principled approach and the Kolmogorov-Smirnov Test stands out. In this section, we introduce the test and then theoretically explain its effectivenes in the following section.

We propose a probing strategy that employs the Kolmogorov-Smirnov Test, which is a statistical method used to compare two sample distributions and determine whether they are drawn from the same underlying distribution. The Kolmogorov-Smirnov Test is an exact test, meaning that distribution does not depend on the underlying cumulative distribution function being tested. Specifically, we view the embedding of each LLM token j as a distribution i.e. $\theta_{ij}^E \sim p_i$. The cumulative distribution function (CDF) of θ_i^E and $\widetilde{\theta}_i^E$ could be denoted by $\Phi_i(\theta)$ and $\widetilde{\Phi_i}(\theta)$. The Kolmogorov-Smirnov distance between the two CDFs is $D_i = \sup_\theta |\widetilde{\Phi_i}(\theta) - \Phi_i(\theta)|$.

Now we wish to determine whether a token embedding before and after fine-tuning comes from the same distribution. Formally, we state Kolmogorov-Smirnov Test as:

Theorem 1. *(Kolmogorov-Smirnov Test, Fasano and Franceschini (1987)) The test statistic for this Kolmogorov-Smirnov Test can be defined in terms of two hypotheses:*
H_0: θ_i and $\widetilde{\theta}_i$ come from the same distribution.
H_1: two samples aren't from the same distribution.

If test T: $D_i > \tau(\alpha)$ is passed, then H_1 holds with confidence $1-\alpha$, where $D_i = \sup_\theta |\widetilde{\Phi_i}(\theta) - \Phi_i(\theta)|$, $\tau(\alpha) = c(\alpha)\sqrt{\frac{2}{d}}$, the value of $c(\alpha)$ is given in the reference table (Karson, 1968), and d is the parameter dimension.

Based on the Kolmogorov-Smirnov Test, we came to propose our method, KS-Lottery.

KS-Lottery. A parameter is designated as a "winning ticket" if it meets the criterion of rejecting the null hypothesis (no difference in the distribution of the embedding before and after fine-tuning) and the alternative hypothesis H_1 (indicating a significant distributional change) is accepted. Kolmogorov-Smirnov Test ensures that if the distribution of parameter θ does not change after fine-tuning, then $\mathbb{P}\left[D_i > \tau(\alpha)\right] < \alpha$, ensuring the majority of crucial token embeddings would be chosen by the test.

3.2 Finding 1: KS-Lottery finds certifiable winning tickets in multilingual transfer.

There exists a set of winning tickets $\boldsymbol{\theta}_a^E$ within the token embeddings $\boldsymbol{\theta}^E = [\theta_0, \theta_1, \cdots, \theta_{|V|}]$, where

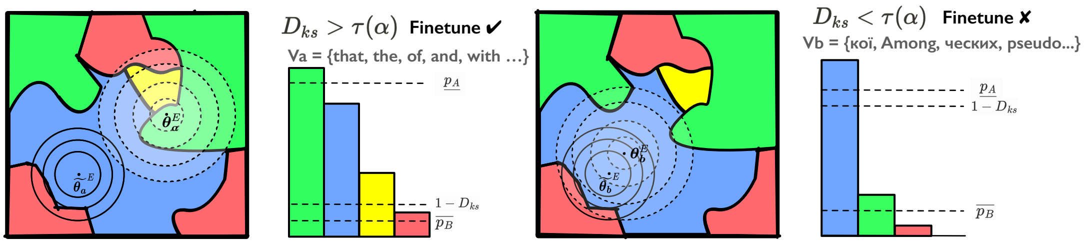

Figure 2: Illustration of $f(\cdot, \boldsymbol{\theta}_a^E, \boldsymbol{\theta}_b^E)$ in 2 dimensions. **Left:** The concentric circles are the density contours of embedding parameters before and after tuning, and the colored landscape is the decision boundaries of $f(\cdot)$. **Right:** the distribution $\mathbb{P}\left[f(x, \theta_a^E, \widetilde{\theta_b^E})\right]$ and $\mathbb{P}\left[f(x, \widetilde{\theta_a^E}, \theta_b^E)\right]$. $\underline{p_A}$ is the probability $\mathbb{P}\left[f(x, \widetilde{\theta}_a^E, \widetilde{\theta_b^E})\right]$ predicts x to be token c_A (color blue), and $\overline{p_B}$ as the probability of second most likely prediction (color red). D_{ks} denotes the Kolmogrov-Smirnov distance between distributions before and after tuning. We choose the set of token embeddings for tuning as those with little distribution overlap before and after fine-tuning, which may be critical to prediction.

$\boldsymbol{\theta}^E = [\boldsymbol{\theta}_a^E, \boldsymbol{\theta}_b^E]$, $\widetilde{\boldsymbol{\theta}}^E = [\widetilde{\boldsymbol{\theta}}_a^E, \boldsymbol{\theta}_b^E]$ is the parameters of embedding layer before and after tuning. If tuning on parameters $\boldsymbol{\theta}_a^E$ would achieve similar performance to full tuning on $\boldsymbol{\theta}^E$ downstream tasks, then we refer to $\boldsymbol{\theta}^E$ as the winning tickets. For convenience, we discuss the downstream task as a next-token prediction problem, which is central to text generation, such that for an LLM $f(\cdot, \boldsymbol{\theta}^E)$, given input context x, generate the most probable token in vocabulary set $\mathcal{Y}$.

$$g(x, \widetilde{\boldsymbol{\theta}}_a^E, \boldsymbol{\theta}_b^E) = g(x, \widetilde{\boldsymbol{\theta}}_a^E, \widetilde{\boldsymbol{\theta}}_b^E),$$
$$\text{where } g(x, \boldsymbol{\theta}_a^E, \boldsymbol{\theta}_b^E) = arg \max_{c \in \mathcal{Y}} \mathbb{P}\left[f(x, \boldsymbol{\theta}_a^E, \boldsymbol{\theta}_b^E) = c\right] \tag{1}$$

Suppose that when the LLM $f(\cdot, \boldsymbol{\theta}_a^E, \boldsymbol{\theta}_b^E)$ predicts any given input x, the most probable token c_A is returned with probability p_A. Given that small changes to parameters in a smooth subspace do not affect decision boundary (Muravev and Petiushko, 2021; Zhao et al., 2022), we could probably guarantee that training on KS-Lottery selected token embeddings could achieve the same performance as full layer Embed Tuning at high confidence. The intuition of the theory is illustrated in Figure 2. Our main theoretical results are as follows, the proofs can be found in Appendix B:

Theorem 2. *(Certified Winning Tickets) Let $f(\cdot, \boldsymbol{\theta}_a, \boldsymbol{\theta}_b) : \mathbb{R}^d \to \mathcal{Y}$ be any random or deterministic function, and let g be defined as in Equation 1. For any input x, suppose $c_A \in \mathcal{Y}$, the bounds of prediction based on random variable parameters $\widetilde{\boldsymbol{\theta}}_a$, $\widetilde{\boldsymbol{\theta}}_b$, $\underline{p_A}, \overline{p_B} \in [0, 1]$ satisfies*

$$\begin{aligned}&\mathbb{P}\left(f(x, \widetilde{\boldsymbol{\theta}}_a, \widetilde{\boldsymbol{\theta}}_b) = c_A\right)\\&\geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}\left(f(x, \widetilde{\boldsymbol{\theta}}_a, \widetilde{\boldsymbol{\theta}}_b = c)\right)\end{aligned} \tag{2}$$

If the set of parameters $\boldsymbol{\theta}_b$ satisfies

$$D(\widetilde{\theta}_b^i, \theta_b^i) < \tau(\alpha) < \frac{\underline{p_A} - \overline{p_B}}{2}, \tag{3}$$

for all $i \in V_b$, the generator partial-tuned on parameters $\boldsymbol{\theta}_a$ always return token c_A, i.e. $g(x, \widetilde{\boldsymbol{\theta}}_a^E, \boldsymbol{\theta}_b^E) = c_A$.

Remark 3. (About $\boldsymbol{\theta}_a$) Theorem 2 immediately holds for *Partial Transfer* setting as specified in § 3.3. As for *Partial Tuning* setting, we need to use the hypothesis that the value of $\widetilde{\boldsymbol{\theta}}_a$ in the Embed Tuning setting and partial-tuning setting are the same for Theorem 2 to hold. This is due to $\boldsymbol{\theta}_a$ taking major effect during Embed Tuning despite small changes in $\boldsymbol{\theta}_b$. We also show with empirical analysis in § 4.2 that $\widetilde{\boldsymbol{\theta}}_a$ in two tuning settings share the same distribution.

Remark 4. (About α) Practically, $\tau(\alpha)$ is small. In the case of fully embedding tuned LLM that has optimal performance and produces confident prediction, $1 > p_A \gg p_B > 0$, we can choose small $\alpha = \tau^{-1}(\frac{p_A - p_B}{2})$ while maintaining performance. In this case, the Kolmogorov-Smirnov Test samples fewer token embeddings for fine-tuning.

Remark 5. (About Applicability) This theorem could be used for certifying fine-tuning lottery tickets for any black-box model and input. Note that KS-Lottery could be applied to the entire model, though we only perform testing on the embedding layer for convenience and training stability.

3.3 Finding 2: 18 identified winning tickets (18 tokens) achieves remarkable performance.

There are two different ways to verify the effectiveness of winning tickets: Partial Tuning (sufficient

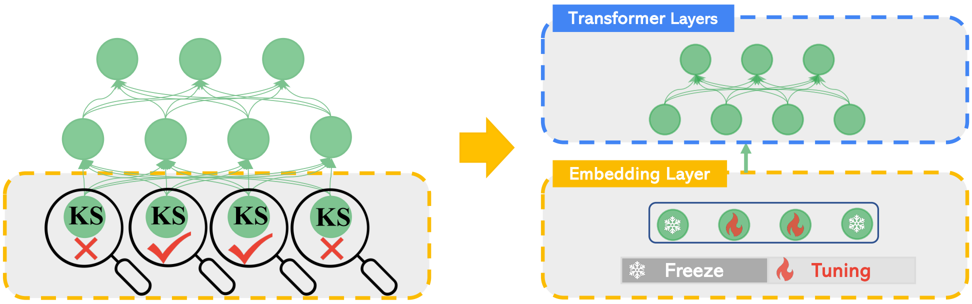

Figure 3: Overview of **Partial Training**, one application of KS-Lottery. It consists of two steps: (1) finding the winning tickets in the embedding layer by Kolmogorov-Smirnov Test; (2) one way to use these winning tickets is partial tuning these tokens ensuring other parameters keep frozen.

Table 1: Empirical verification of winning tickets with **Partial Tuning** and **Partial Transfer** with bilingual translation data. * denotes that the random results from three distinct seeds.

Method	LLaMA-7B # Token	en→ro	en→es	en→de	en→ca	Avg.	Mistral-7B # Token	en→ro	en→es	en→de	en→ca	Avg.
Original Model	-	3.5	4.8	4.8	5.7	4.6	-	13.4	11.7	14.2	16.8	14.0
Full Tuning		28.3	23.5	22.5	34.9	27.3		33.4	25.1	29.1	37.8	31.4
Embed Tuning		28.7	25.5	25.8	36.1	29.0		**34.2**	**28.3**	**33.2**	40.2	**34.0**
Random Tuning*	<18 (p-value <0.05)	0.1	0.1	0.1	0.1	0.1	<169 (p-value <0.05)	16.2	13.9	15.3	20.3	16.4
Partial Tuning		20.7	**26.7**	26.4	**37.7**	27.9		23.7	26.3	27.2	34.2	27.9
Partial Transfer		23.4	22.6	16.1	31.1	23.4		26.7	27.3	30.5	37.5	30.5
Random Tuning*	<100 (p-value <0.25)	0.1	0.1	0.1	0.1	0.1	<170 (p-value <0.25)	17.3	13.9	15.3	20.4	16.7
Partial Tuning		26.9	27.3	27.2	37.4	29.7		23.7	26.3	27.2	34.2	27.9
Partial Transfer		25.1	26.5	**30.4**	34.4	29.1		26.7	27.3	30.5	37.5	30.5
Random Tuning*	≤800	5.8	8.7	5.2	11.8	7.9	≤180	16.3	13.9	15.4	20.3	16.5
Partial Tuning		29.4	27.3	30.1	37.7	31.1		30.7	28.2	29.5	39.2	31.9
Partial Transfer		**30.2**	**26.7**	**30.4**	37.6	**31.2**		33.7	27.9	32.6	**40.3**	33.6

condition) and Partial Transfer (necessary condition). **Partial Tuning.** As shown in Figure 3, only train winning tickets, keeping the remaining parameters frozen. **Partial Transfer.** Given a model trained by Embed Tuning, we select the winning tickets from the embedding layer and use them to replace the corresponding parameters in the original model, thus curating a new model.

As shown in **Table 1**, just tuning winning tickets can achieve results on par with Embed Tuning. Meanwhile, by only modifying the winning tickets, it manages to retain $86.9\%(31.3/36.1)$ of the performance. The experimental result demonstrates that a low-dimensional subspace exists, guided by the winning tickets, that can achieve comparable performance as optimizing all parameters in the embedding-tuned model. The result empirically verifies the effectiveness of winning tickets.

4 Analysis

KS-Lottery . There are three different ways to use the winning tickets. **Partial Tuning**, **Partial Transfer** and **Frequency Tuning**, which only train the high-frequency tokens in the corpus.

Algorithm 1: Next Token Prediction Certification

Input: Sequence x, ground truth output token is y. LLM $f(x,\boldsymbol{\theta})$ that maps sequence to the probability of the next token class, with pre-trained parameters $\boldsymbol{\theta}$. Multilingual training set $\mathcal{D}$. KS-Lottery parameter α.

Output: Whether $g(x,\widetilde{\boldsymbol{\theta}}_a^E,\boldsymbol{\theta}_b^E)=y$

fine-tune LLM $f(x,\boldsymbol{\theta})$ on $\mathcal{D}$ with parameters $\widetilde{\boldsymbol{\theta}}$.

$\boldsymbol{\theta}_a^E,\boldsymbol{\theta}_b^E=$ KS-Lottery$(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}},\tau(\alpha))$

$\underline{p_A},\overline{p_B}=$ Top-2 $\left(f(x,\boldsymbol{\theta}_a^E,\boldsymbol{\theta}_b^E)\right)$

if $g(x,\widetilde{\boldsymbol{\theta}}_a^E,\widetilde{\boldsymbol{\theta}}_b^E)=y$ ***and*** $\frac{\underline{p_A}-\overline{p_B}}{2}>\tau(\alpha)$ **then**

  return True // Certification can be provided.

else

  return False // Certification cannot be provided.

end

Other Baselines. Original Model Directly using the LLaMA-7B (Touvron et al., 2023a)/Mistral-7B (Jiang et al., 2023) weight on test data, without any tuning. **Random Tuning** Only the tokens randomly selected in the embedding layer are fine-tuned, while the remaining parameters are kept frozen. **Full Tuning** is an approach to transfer learning where the weights of a pre-trained entire model are trained on new data. **Embed Tuning** Merely fine-tuning the embedding layer of a model keeping the remaining param-

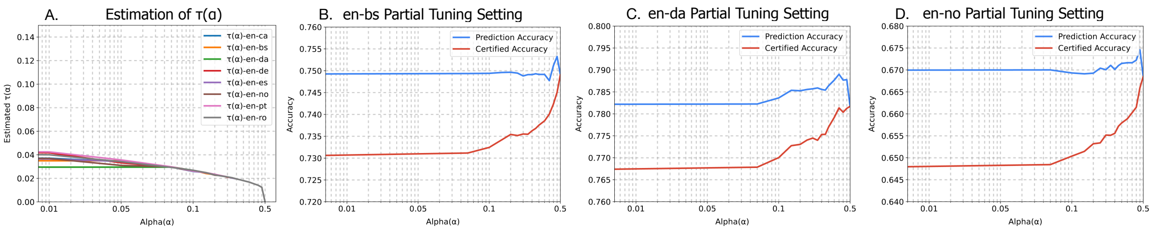

Figure 4: Certified experiment under **Partial Tuning** setting. **A**: Estimation of $\tau(\alpha)$ w.r.t different α by running Kolmogorov-Smirnov Test between the distribution of LLaMA-7B embedding and fine-tuned embedding on different datasets. **B C D**: Comparison between *Certified Accuracy* and *Empirical Prediction Accuracy* w.r.t. different α on 3 datasets. More results are shown in Appendix C.

Table 2: **Partial Tuning** of the winning ticket is a parameter-efficient method. Its performance is demonstrated by comparing it with other parameter-efficient methods on the Flores-101 devtest.

Method	# Param	en→ro	en→es	en→ca	en→no	Avg.
LLaMA w/o Tuning	7B	3.5	4.8	5.7	3.2	4.3
Full Fine-Tuning	7B	28.3	23.5	34.9	21.2	27.0
QLoRA	17M	25.8	23.5	33.5	17.2	25.0
LoRA	4.2M	**29.8**	26.4	37.3	19.6	28.3
Prefix Tuning	0.04M	17.6	20.4	20.8	11.0	17.4
Embed Tuning	131M	28.7	25.5	36.1	19.5	27.5
Partial Tuning	3.2M	27.3	**30.1**	**39.3**	**22.1**	**29.7**

eters frozen. **LoRA** (Hu et al., 2022) utilizes low-rank matrices for approximating parameter updates. **QLoRA** (Dettmers et al., 2023) reduces memory usage enough to efficiently finetune. **Prefix-Tuning** (Li and Liang, 2021) introduce a lightweight prefix module into the input layer and each transformer layer, enabling efficient training over these modules.

Training and Evaluation. To ensure a fair comparison, we apply various parameter-efficient settings on LLaMA-7B using Lego-MT (Yuan et al., 2023a) 10k data. For full tuning, training with LoRA, and Embed Tuning, we set the learning rate to $2e-5$ and the number of epochs to 3. For partial tuning, prefix-tuning, and KS-Lottery, we set the learning rate to $1e-2$ and the number of epochs to 5. All other parameters are kept consistent across all settings. We test each model on the Flores-101 (Goyal et al., 2022) devtest, which offers human-written translation pairs across 101 languages. In alignment with Flores-101, we employ the same evaluation metric, sentence piece BLEU (spBLEU) on beam size$=4$, to assess multilingual capabilities.

4.1 KS-Lottery Certification

Certified Accuracy, when using certified winning tickets Embed Tuning, is measured as the proportion of correct predictions from an embedding tuned model (reference model) that is certified to be correct at a significance level of α. The certification process follows Theorem 2 and is stated in Algorithm 1, where for each prediction based on input sequence x, we compare the probability gap between two most-likely tokens by the original LLaMA, i.e. $\frac{\underline{p_A}-\overline{p_B}}{2}$ and $\tau(\alpha)$. $\tau(\alpha)$ is a static value that could be obtained through the equation, though we use the value estimated with Scipy Kolmogorov-Smirnov test to be precise.

To check the validity of our certification method and test the empirical tightness of the certification bound, we experiment on Flores-101 devtest. A model, developed using partial tuning, and Embed Tuning, utilizes both the instruction and a partial sequence of reference tokens as input. Temporarily both settings of certification only use the first twenty token prediction results for calculation for a fair comparison and don't let overlong sentences dominate the results. It subsequently predicts the next token in the partial reference sentence. The "prediction accuracy" is ascertained by comparing the predicted token, generated by the partially tuned model, with the reference token.

Our theoretical discovery provides certification for the lottery tickets hypothesis. In Figure 4 that plots certified accuracy as a function of α, the certified accuracy always increases gradually until reaching the same value as the prediction accuracy. This is due to the decrease in $\tau(\alpha)$ as α increases and reaches 0 when $\alpha = 1$. At any α, the empirical estimation of certified accuracy is a lower bound of prediction accuracy, providing a performance guarantee for tuning on the lottery tickets.

Certification lower bound is tighter for larger α. When $\underline{p_A} \gg \overline{p_B}$, there must exist a α that satisfies the certification constraint. Since $\underline{p_A}$ and

Table 3: The efficiency and interpretability of KS-Lottery. Given an en→ca Embed Tuning bilingual model, KS-Lottery is denoted by the p-value, whereas the metrics from other evaluation methods are normalized for comparability by calculating the importance # rank/32000.

Idx	Str	Freq.	KS-Lottery	Cos	Absolute	Relative	Ratio	KL
13	\n	5	0.0000	0.0001	0.9972	0.0004	0.0004	0.0154
263	a	10	0.0000	0.0016	0.9964	0.1376	0.1376	0.0071
278	the	569.5	0.0000	0.0003	0.9508	0.0546	0.0546	0.0018
297	in	286	0.0000	0.0034	0.8807	0.9885	0.9885	0.0353
304	to	146	0.0000	0.0016	0.8860	0.9032	0.9032	0.0155
310	of	264	0.0000	0.0009	0.1571	0.9955	0.9955	0.0171
322	and	680	0.0000	0.0004	0.9913	0.0060	0.0060	0.0077
338	is	1356	0.0214	0.0005	0.1557	0.0201	0.0201	0.0023
376	"	174	0.0163	0.0000	0.0375	0.9796	0.9796	0.0001
393	that	2965	0.0175	0.0007	0.9922	0.0086	0.0086	0.0035
411	with	2102	0.0459	0.0013	0.1553	0.1493	0.1493	0.0054
29871	~	6	0.0000	0.0005	0.9936	0.9908	0.9908	0.0026
29889	.	4	0.0000	0.0002	0.9962	0.8387	0.8387	0.0068
29892	,	3	0.0000	0.0002	0.9976	0.9950	0.9950	0.0048
29896	1	9	0.0000	0.0023	0.0440	0.0500	0.0500	0.0139
29900	0	8	0.0008	0.0045	0.0728	0.9743	0.9743	0.0254
29901	:	30	0.0069	0.0003	0.9791	0.9950	0.9950	0.0008
29949	O	294	0.0406	0.0028	0.9336	0.0359	0.0359	0.0201

$\overline{p_B}$ are input and model dependent, we empirically assessed the tightness of our bound by comparing the estimated value of certified accuracy with prediction accuracy. As shown in Figure 4, the bound is tighter when $\alpha \to 1$, and the gap is larger when $\alpha \to 0$. This is due to when fewer token embeddings are chosen, the certification expects the performance to be worse, though in actual tuning even zero-shot performance is quite good due to pre-training. The gap is quite small in practical cases, the gap is about 2% on average for $\alpha = 0.05$ and around 1.5% on average for $\alpha = 0.5$. Therefore, we recommend using $\alpha \geq 0.05$ significance rate for guaranteed.

4.2 KS-Lottery Efficiency and Interpretability

Partial Tuning can serve as a parameter-efficient tuning method. We proceed to evaluate it as a method that optimizes parameter usage, in comparison with another similar method. As illustrated in Table 2, in contrast to other methods such as LoRA (Hu et al., 2022), QLoRA (Dettmers et al., 2023) and Prefix-tuning (Li and Liang, 2021), Partial Tuning eliminates the need for an additional model structure. Remarkably, our method not only matches but frequently surpasses the performance of these alternate approaches.

Comparing with other selection methods, KS-Lottery is parameter-efficient. There are other ways (more details in Appendix C) to select critical parameters, such as Cos, Absolute, Relative, Ratio, and KL. In Table 3, we selected only 18 selective tokens by KS-Lottery on en→ca bilingual data with extremely stringent requirements (p-value < 0.05). At this time, we used other evaluation methods to

Table 4: The best result from Frequency Tuning, Partial Tuning, and the result of Embed Tuning.

Setting	en→ca	en→da	en→de	en→es	en→no	Avg.
Embedding Tuning	36.1	32.7	25.8	25.5	19.5	27.9
Partial Tuning	**37.7**	**33.3**	30.1	**27.3**	**19.8**	**29.6**
Frequency Tuning	**37.7**	30.0	**30.7**	26.1	19.6	28.8

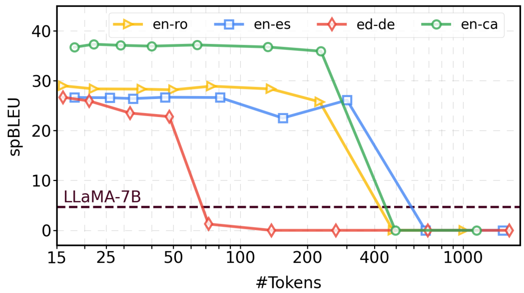

Figure 5: When selective tokens are restricted from being updated, the model's fine-tuning process for downstream tasks loses its effectiveness.

measure the importance of these tokens separately: the evaluation results of these tokens based on distribution changes (such as Cos, KL) are highly consistent with KS-Lottery; those evaluation methods based on value changes (Absolute, Relative, and Ratio) tend to give these tokens very low importance evaluation results. If we use different methods to select the top 18 important tokens for **Partial Transfer** experiments, the performance of different selective methods is 31.3 (KS-Lottery), 28.7 (Cos), 5.8 (Absolute), 5.8 (Relative), 5.8 (Ratio) and 1.6 (KL). Based on the experimatl results, we find that except for the results of Cos, which are close to the performance of KS-Lottery (but still -2.6 points), the translation performance of other methods is very poor.

Winning tickets are high-frequency tokens in the corpus. In the embedding layer, each dimension is associated with a meaningful token index. By referencing the vocabulary, we can decode the text representation corresponding to this index, as illustrated in Table 3 (Str). To examine the distinct characteristics of these tokens, we retrieve 50k sentences in ca language from the MC4 (Raffel et al., 2019) dataset, then employ LLaMA's tokenizer to segment all sentences, tallying the occurrence of each token within the corpus. Table 3 (Freq.) indicates winning tickets commonly associated with the most frequently occurring tokens in the corpus. Based on this finding, we start the training only with the high-frequency tokens in the corpus. The summary of different tuning setting, as shown

Table 5: Impact of Different α on the en→ca. For α values greater than 0.05, the verification percentage reaches 94.3%, indicating substantial but not complete verification. The empirical accuracy under these conditions is also satisfactory.

pvalue<α	Verified Percentage (% $D_i > \tau(\alpha)$)	Original Accuracy	Certified Accuracy	Empirical Accuracy
0.05	94.3	75.86	65.77	78.87
0.10	95.1	75.86	65.98	78.85
0.20	95.6	75.86	66.15	78.93
0.30	95.8	75.86	66.19	78.98
0.50	96.5	75.86	66.35	78.97
0.70	96.8	75.86	66.48	79.03
0.90	97.5	75.86	66.66	79.09
1.00	100.0	75.86	67.23	78.46

Table 6: With varying values of α, the distribution of winning tickets remains largely consistent across both Partial Tuning and Partial Transfer.

α	ro	es	de	ca	pt	da	no	bs
0.05	0.56	0.67	0.81	0.78	0.82	0.33	0.64	0.82
0.1	0.75	0.78	0.72	0.80	0.85	0.71	0.79	0.81
0.2	0.78	0.75	0.68	0.73	0.75	0.79	0.83	0.81
0.3	0.69	0.82	0.85	0.77	0.86	0.76	0.81	0.80
0.4	0.75	0.92	0.89	0.83	0.83	0.78	0.82	0.84
0.5	0.83	0.87	0.82	0.80	0.85	0.86	0.84	0.90
0.6	0.83	0.90	0.88	0.85	0.86	0.86	0.85	0.92
0.7	0.83	0.91	0.88	0.86	0.90	0.86	0.89	0.88
0.8	0.82	0.90	0.87	0.87	0.90	0.87	0.88	0.88
0.9	0.82	0.86	0.84	0.85	0.88	0.84	0.86	0.85

in Table 4, suggests the training is sufficient with frequency token tuning.

Sensitivity in the siginificance level Selection (α). Our theoretical framework provides a way for heuristically finding an effective α, as mentioned in Remark5. Since the original fully-tuned prediction model is available, we could directly predict the range of $\frac{p_A - p_B}{2}$ as $[s_{min}, s_{max}]$ for all predictions in the dataset. An ideal α would satisfy $\tau(\alpha) < s_{min}$, under this condition, every data point in the dataset is guaranteed to make the same prediction as the original prediction model.

In practice, we don't need a full certification of the dataset to make the KS-Lottery work, as this would result in quite large α values. As we can see in the en→ca example below: for α larger than 0.05, the verified percentage is an acceptable 94.3%, though not fully verified, and the empirical accuracy proves acceptable. In practice, the acceptable range of α varies between datasets, though usually α within 0.05-0.4 guarantees a verified percentage 95%.

The distribution of winning tickets under both Partial Tuning and Partial Transfer is largely identical. By conducting KS-Lottery with varying α values, we obtain different winning tickets (denoted as $\boldsymbol{\theta}_a$). Furthermore, we perform a Kolmogorov-Smirnov Test on the winning tickets tuned with partial tuning and Embed Tuning, and compute the number of tokens that exhibit a significant difference ($\widetilde{\boldsymbol{\theta}}_a'$). The ratio of unchanged tokens is calculated using the formula $1 - \widetilde{\boldsymbol{\theta}}_a' / \widetilde{\boldsymbol{\theta}}_a$. As shown in Table 6, the distribution of winning tickets before and after tuning (Partial Transfer vs Partial Tuning) remains largely consistent.

Keeping the parameters of winning tickets frozen while tuning the remaining parameters could lead to the collapse of the Embed Tuning. Our previous experiments have highlighted the significance of winning tickets in training, while it's intriguing to consider whether such an important function can be replaced by other tokens. To delve into this problem, we freeze the winning tickets, and fine-tune the reset parameters on bilingual data. As shown in Figure 5, a small amount of disabled winning tickets is acceptable for tuning. However, as the number of disabled tokens increases, the entire tuning process crashes. The process reveals a remarkable finding: for a vocabulary size of 32k, 1k is a very small, yet fewer than 1k winning tickets play a crucial role in Embed Tuning.

Applying KS-Lottery on the whole model. After training all model parameters using en→ca bilingual sentence pair from Lego-MT (Yuan et al., 2023a), we utilize the KS-Lottery to identify a small parameter set for multilingual transfer by comparing the parameters before and after tuning. Interestingly, no parameters exhibite significant changes before and after tuning. Concurrently, the study by Yuan et al. (2023b) revealed that fine-tuning specific layers, including the embedding layer, can yield results comparable to those of full-tuning. Figure 6 reveals that following fine-tuning on the Lego-MT (Yuan et al., 2023a) en→ca bilingual dataset, a subset of token embeddings exhibit significant changes in their parameters. Additionally, a small number of parameters (fewer than two for each layer with a significance level of $\alpha < 0.05$) demonstrate substantial changes within LayerNorm. However, for multilingual transfer, the impact of LayerNorm varies and is not uniform across the lower and higher layers.

5 Conclusion

This work presents a novel method, KS-Lottery, which applies the lottery ticket hypothesis to

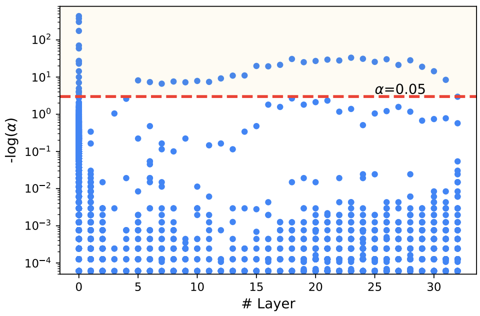

Figure 6: Applying KS-Lottery on whole LLaMA-7B which is single-layer tuned on Lego-MT en→ca 10k data. Each layer is trained in isolation and is analyzed by KS-Lottery to identify the parameters with significant changes (as indicated by scatter points above the red line). We find that within each Transformer layer, changes are primarily focused on LayerNorm, while other notable changes occur in the embedding layer.

LLMs fine-tuning. By employing the Kolmogorov-Smirnov Test, KS-Lottery first analyzes the shift in the parameter distribution before and after fine-tuning, and then identifies a small but effective subset of LLM parameters. Theoretical evidence confirms that KS-Lottery can pinpoint certified winning tickets within the embedding layer, thereby ensuring performance equivalent to full tuning. Notably, KS-Lottery surpasses other parameter-efficient tuning algorithms by identifying fewer parameters for fine-tuning while maintaining similar performance levels.

References

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7319–7328, Online. Association for Computational Linguistics.

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*.

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

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6309–6317.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

G. Fasano and A. Franceschini. 1987. A multidimensional version of the Kolmogorov–Smirnov test. *Monthly Notices of the Royal Astronomical Society*, 225(1):155–170.

Jonathan Frankle and Michael Carbin. 2018. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*.

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.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4129–4142, Hong Kong, China. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Wenxiang Jiao, Jen tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Parrot: Translating during chat using large language models tuned with human translation and feedback. In *Findings of EMNLP*.

Marvin Karson. 1968. Handbook of methods of applied statistics. volume i: Techniques of computation descriptive methods, and statistical inference. volume ii: Planning of surveys and experiments. i. m. chakravarti, r. g. laha, and j. roy, new york, john wiley; 1967, $9.00. *Journal of the American Statistical Association*, 63(323):1047–1049.

Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. 2019. Certified robustness to adversarial examples with differential privacy. *Preprint*, arXiv:1802.03471.

Roman Levin, Manli Shu, Eitan Borgnia, Furong Huang, Micah Goldblum, and Tom Goldstein. 2022. Where do models go wrong? parameter-space saliency maps for explainability. *Advances in Neural Information Processing Systems*, 35:15602–15615.

Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. 2018. Measuring the intrinsic dimension of objective landscapes. In *International Conference on Learning Representations*.

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. *Preprint*, arXiv:2305.15083.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Eran Malach, Gilad Yehudai, Shai Shalev-Schwartz, and Ohad Shamir. 2020. Proving the lottery ticket hypothesis: Pruning is all you need. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6682–6691. PMLR.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35.

Nikita Muravev and Aleksandr Petiushko. 2021. Certified robustness via randomized smoothing over multiplicative parameters of input transformations. *arXiv preprint arXiv:2106.14432*.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Momchil Peychev, Anian Ruoss, Mislav Balunovic, Maximilian Baader, and Martin T. Vechev. 2021. Latent space smoothing for individually fair representations. *CoRR*, abs/2111.13650.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. 2018. Semidefinite relaxations for certifying robustness to adversarial examples. *CoRR*, abs/1811.01057.

Anian Ruoss, Mislav Balunovic, Marc Fischer, and Martin T. Vechev. 2020. Learning certified individually fair representations. *CoRR*, abs/2002.10312.

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. *Preprint*, arXiv:2302.13971.

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*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Yudong Wang, Chang Ma, Qingxiu Dong, Lingpeng Kong, and Jingjing Xu. 2023. A challenging benchmark for low-resource learning. *arXiv preprint arXiv:2303.03840*.

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*.

Fei Yuan, Yinquan Lu, Wenhao Zhu, Lingpeng Kong, Lei Li, Yu Qiao, and Jingjing Xu. 2023a. 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.

Fei Yuan, Shuai Yuan, Zhiyong Wu, and Lei Li. 2023b. How multilingual is multilingual llm? *arXiv preprint arXiv:2311.09071*.

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. *Preprint*, arXiv:2205.01068.

Haiteng Zhao, Chang Ma, Qinyu Chen, and Zhi-Hong Deng. 2021. Domain adaptation via maximizing surrogate mutual information. *arXiv preprint arXiv:2110.12184*.

Haiteng Zhao, Chang Ma, Xinshuai Dong, Anh Tuan Luu, Zhi-Hong Deng, and Hanwang Zhang. 2022. Certified robustness against natural language attacks by causal intervention. In *International Conference on Machine Learning*, pages 26958–26970. PMLR.

Rui Zheng, Bao Rong, Yuhao Zhou, Di Liang, Sirui Wang, Wei Wu, Tao Gui, Qi Zhang, and Xuanjing Huang. 2022. Robust lottery tickets for pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2211–2224, Dublin, Ireland. Association for Computational Linguistics.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Extrapolating large language models to non-english by aligning languages. *arXiv preprint arXiv:2308.04948*.

Acknowledgments

Authors of this paper would like to thank Zixian Huang for giving many helpful comments on previous versions of this paper. We deeply express our gratitude to Shanghai AI Laboratory. This work is partially supported by the National Key R&D Program of China (NO.2022ZD01601022030).

A Limitations and Broader Impacts

Limitations. In the scope of this research, our proposed KS-Lottery mainly targets multilingual research and has been thoroughly validated in bilingual translation tasks. However, we believe that our certified theoretical framework is generalizable to other types of tasks, and we leave these explorations for future work.

Broader Impacts. This paper presents work whose goal is to advance the field of machine learning and multilingual research. We do not anticipate it will cause negative societal impacts, such as potential malicious or unintended uses.

B Proofs

Theorem 3. (Certified Winning Tickets) Let $f(\cdot, \boldsymbol{\theta}^a, \boldsymbol{\theta}^b) : \mathbb{R}^d \to \mathcal{Y}$ be any random or deterministic function, and let g be defined as in Equation 1. For any input x, suppose $c_A \in \mathcal{Y}, \underline{p_A}, \overline{p_B} \in [0,1]$ satisfies

$$\mathbb{P}\left(f(x, \widetilde{\boldsymbol{\theta}}^a, \widetilde{\boldsymbol{\theta}}^b) = c_A\right) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x+\varepsilon) = c)$$

If the set of parameters $\boldsymbol{\theta}^b$ satisfies

$$D(\widetilde{\theta}_i^b, \theta_i^b) < \tau(\alpha) < \frac{\underline{p_A} - \overline{p_B}}{2},$$

for all $i \in V_b$, the classifier partial-tuned on parameters $\boldsymbol{\theta}^a$ always return class c_A, i.e. $g(x, \widetilde{\boldsymbol{\theta}}_a^E, \boldsymbol{\theta}_b^E) = c_A$.

Proof Sketch Fix an input x, we study the change of prediction w.r.t. change in the distribution of $\boldsymbol{\theta}^b$. Let $A = \{\boldsymbol{\theta} | g(X, \widetilde{\boldsymbol{\theta}}^a, \boldsymbol{\theta}) = c_A\}$, $B = \{\boldsymbol{\theta} | g(x, \widetilde{\boldsymbol{\theta}}^a, \boldsymbol{\theta}) \neq c_A\}$. Using definition, $\mathbb{P}(\widetilde{\boldsymbol{\theta}}^b \in A) \geq \underline{p_A}$, $\mathbb{P}(\widetilde{\boldsymbol{\theta}}^b \in B) \leq \overline{p_B}$. Since $D(\widetilde{\boldsymbol{\theta}}^b, \boldsymbol{\theta}^b) = \max_i D(\widetilde{\theta}_i^b, \theta_i^b) \leq \tau(\alpha)$, the minimum overlapping cumulative probability $\mathbb{P}(\widetilde{\boldsymbol{\theta}}^b \in S \cap \boldsymbol{\theta}^b \in S) \geq 1 - \tau(\alpha)$, where S is any contour set $S = \{\boldsymbol{\theta} || \boldsymbol{\theta} - \widetilde{\mu}^b| \leq s, s \geq 0\}$.

Now we compare the probability that $f(x, \widetilde{\boldsymbol{\theta}}^a, \boldsymbol{\theta}^b)$ predicts c_A or other classes. The probability that the partial-tuned f predicts c_A is

$$\begin{aligned}
\mathbb{P}(\boldsymbol{\theta}^b \in A) &\geq \mathbb{P}(\boldsymbol{\theta}^b \in A \cap \boldsymbol{\theta}^b \in S) \\
&\geq \mathbb{P}\left[(\widetilde{\boldsymbol{\theta}}^b \in S \cap \boldsymbol{\theta}^b \in S) - (\widetilde{\boldsymbol{\theta}}^b \in S \cap \widetilde{\boldsymbol{\theta}}^b \notin A)\right] \\
&\geq (1 - \tau(\alpha)) - (1 - \underline{p_A}) = \underline{p_A} - \tau(\alpha)
\end{aligned}$$

Similarly, we can prove that $\mathbb{P}(\boldsymbol{\theta}^b \in B) \leq \overline{p_B} + \tau(\alpha)$. Thus when $\tau(\alpha) < \frac{\underline{p_A} - \overline{p_B}}{2}$ and we partial-tune all parameters $\boldsymbol{\theta}^a$ that fails to pass the KS-test, then the partial-tuned model $g(x, \widetilde{\boldsymbol{\theta}}_a^E, \boldsymbol{\theta}_b^E) = c_A$, i.e. always give the same prediction as the embedding fully-tuned model. □

C More Analysis

Table 7: Certified experiment under **Partial Tuning** setting w.r.t significance level α. When $\alpha = 1$, all token embeddings are chosen during tuning and thus certified accuracy equals to prediction accuracy.

α	**Setting**	**en→ca**	**en→bs**	**en→da**	**en→no**	**en→pt**	**en→ro**	**en→es**	**en→de**
0	Certified	0.5649	0.4592	0.5384	0.4083	0.5959	0.5416	0.5108	0.5505
	Prediction	0.7765	0.7040	0.7544	0.6476	0.7912	0.7555	0.7191	0.7644
0.05	Certified	0.7722	0.7312	0.7679	0.6484	0.7999	0.7714	0.7177	0.7313
	Prediction	0.7887	0.7494	0.7822	0.6700	0.8156	0.7852	0.7372	0.7537
0.1	Certified	0.7752	0.7325	0.7700	0.6504	0.8006	0.7736	0.7197	0.7347
	Prediction	0.7885	0.7494	0.7836	0.6693	0.8143	0.7855	0.7375	0.7551
0.5	Certified	0.7797	0.7375	0.7770	0.6579	0.8042	0.7765	0.7240	0.7482
	Prediction	0.7897	0.7492	0.7864	0.6715	0.8143	0.7847	0.7356	0.7619
1	Certified	0.7846	0.7491	0.7817	0.6685	0.8097	0.7795	0.7349	0.7525
	Prediction	0.7846	0.7491	0.7817	0.6685	0.8097	0.7795	0.7349	0.7525

The observed trend in the value of α suggests the presence of an optimal size for the set of selective tokens. In Figure 8, we examined the translation performance on the Flores-101 dataset under varying thresholds. It's important to note that by default, the threshold creates an interval that is closed on the left and open on the right. Our observations indicate that even a minimal threshold is capable of preserving approximately 80% of the performance that is achieved through comprehensive embedding fine-tuning, as highlighted in green. Interestingly, each plot exhibits the same pattern characterized by an initial increase in performance, followed by a subsequent decline. This pattern suggests the existence of an optimal point that maximizes performance.

Another Selective Methods There are five commonly used methods for determining parameters: Cos-Similarity (cos), Absolute Value (absolute), Related Value (relative), Related Ratio (ratio), and KL Divergence (KL). Each method emphasizes different aspects and requires a heuristic value p for selection.

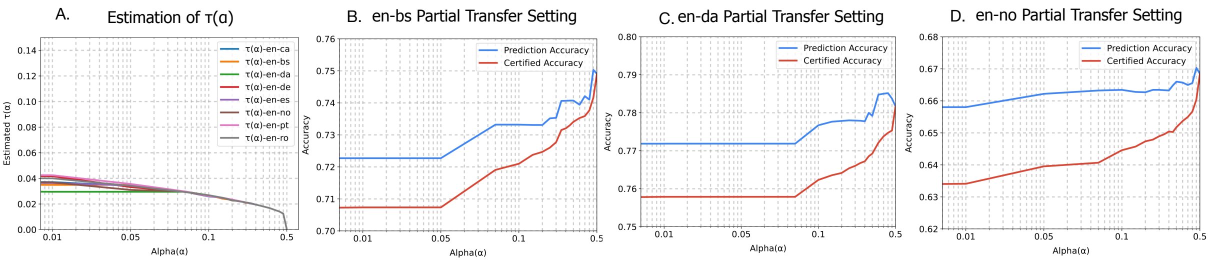

Figure 7: Certified experiment under **Partial Transfer** setting. **A**: Estimation of $\tau(\alpha)$ w.r.t different α values by running KS-Test between the distribution of LLaMA-7b embedding and fine-tuned embedding on different datasets. **B C D**: Comparison between *Certified Accuracy* and *Empirical Prediction Accuracy* w.r.t. different α values on three datasets.

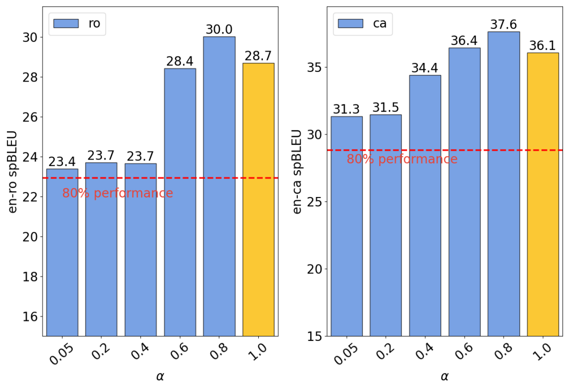

Figure 8: The translation performance of updated models with replacement method on Flores-101 with different token numbers. We observe the minimal threshold is sufficient to maintain approximately 80% of the performance achieved through embedding tuning (marked in orange). Meanwhile, for each plot, there exists a trend of initial increase followed by a decline, with the presence of an optimum point.

en→ca	en→es	en→ro	en→da	en→de	en→pt
13	**13**	**13**	**13**	**13**	**13**
263	262	**278**	263	263	263
278	263	**297**	**278**	**278**	**278**
297	278	**304**	**297**	**297**	**297**
304	297	**310**	**304**	**304**	**304**
310	304	**322**	**310**	**310**	**310**
322	310	338	**322**	**322**	**322**
338	313	366	**29871**	338	338
376	**322**	393	**29889**	363	363
393	363	**29871**	**29892**	**29871**	411
411	393	**29889**	**29896**	**29889**	**29871**
29871	411	**29892**	29900	**29892**	**29889**
29889	**29871**	**29896**		**29896**	**29892**
29892	**29889**	29900		29901	**29896**
29896	**29892**	29901		29915	29900
29900	**29896**				29901
29901	29897				29915
29949	29901				

Table 8: Across various languages, the winning tickets share some common pattern.

Cos-Similarity: This method determines the similarity between θ_i^E and $\widetilde{\theta}_i^E$ from the perspective of vector dot product. A lower cos-similarity suggests that it is the target of fine-tuning adjustment.

Absolute Value: This is calculate as $|\widetilde{\theta}_i^E - \theta_i^E|$, which is concerned with the magnitude of the change. A larger value indicates that it is the target of fine-tuning.

Related Value: This is calculated as $\widetilde{\theta}_i^E / \theta_i^E$, which focuses on the rate of change before and after fine-tuning. A larger ratio suggested that it is the target.

Related Ratio: This method considers the initial value and calculates the result as $(\widetilde{\theta}_i^E - \theta_i^E)/\theta_i^E$. A larger ratio indicates that it is the key adjustment in fine-tuning.

KL Divergence: This is a statistical distance, measuring the distance between $\widetilde{\theta}_i^E$ and θ_i^E, with a focus on the distribution itself.

C.1 When Fine-tuning on different languages, the winning tickets contain the same tokens.

The winning tickets across various languages overlap with a subset of tokens, as shown in Table 8, {index id = 13 (token="\n"), index id=278 (token='the'), index id=29871 (token='∼'), and so on}, which are frequently used and common across various languages.

D AI Assistants

We just use AI assistants for our sentence-level polishing.

E 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.