Adapt and Prune Strategy for Multilingual Speech Foundational Model on Low-resourced Languages

Hyeon Soo Kim[*], **Chunghyeon Cho**[*], **Hyejin Won**[*] and **Kyung Ho Park**[†]
SOCAR AI Research, Seoul, Republic of Korea
{lucci, yoplait, cheese, kp}@socar.kr

Abstract

While foundational speech models such as Whisper demonstrate state-of-the-art performance across various benchmarks, it necessitates an adaptation process for specific downstream tasks, particularly in low-resourced languages. Classical full fine-tuning (FFT) successfully adapts the model to downstream tasks, but requires computational resources proportional to the extensive model size. Parameter-efficient fine-tuning (PEFT) methods introduced to address this issue effectively adapt a given model with less trainable parameters, but demand higher inference complexities for the increased number of overall parameters. In response to these issues, we propose **PEPSI**—a **P**arameter-**E**fficient ada**P**tation for the Speech foundat**I**onal model. Our PEPSI integrates a compact adapter module into the decoder layers of the foundational model and removes neurons irrelevant to the downstream task. Through experiments, we showcase that PEPSI achieves performance surpassing PEFT methods and comparable to FFT, while significantly reducing trainable and inference parameters to utilize Whisper on low-resourced languages that require additional adaptation.

1 Introduction

Recent advancements in speech foundational models pre-trained on large-scale, multilingual data have facilitated the resolution of speech recognition tasks to human standards in a wide array of languages. However, such models, including the recently introduced Whisper (Radford et al., 2023) and Universal Speech Model(USM) (Zhang et al., 2023), tend to exhibit suboptimal performance in languages like *Swahili* or *Malayalam* that cover only a small portion of their pre-training data. A prevalent strategy to address this limitation involves adapting these models to the target language of interest (Singh et al., 2023). Full fine-tuning (FFT) involves updating all the parameters within the model, demanding substantial computational resources. Parameter Efficient Fine-Tuning (PEFT) methods, proposed to reduce the training costs required for FFT, introduce additional small-scale, trainable parameters referred to as adapters into the model's architecture (Houlsby et al., 2019; Liu et al., 2021). These techniques, such as Low-Rank Adaption (Hu et al., 2021), update only the adapter parameters while freezing the backbone model. While significantly reducing the computational resources for training, such methods hold drawbacks of increasing the parameter number during inference.

Another avenue to mitigate computational costs involves model compression and pruning. These approaches propose methods to reduce the model size by eliminating specific neurons from model weight matrices (LeCun et al., 1989). These sub-networks are identified by assessing magnitude changes before and after training the model, removing neurons with low weight magnitudes as they are considered less crucial (Han et al., 2015; Frankle and Carbin, 2018). Although these pruning methods succeeded in reducing the weight of foundational models, the resulting task performances were not adequate for practical utilization.

1.1 Main Idea and Its Novelty

Building upon previous research by (Wang et al., 2020; Houston and Kirchhoff, 2023), which uncovered the existence of language-specific parameters and multilingual interference within Large Language Models (LLMs), we propose that a similar phenomenon may also be present in the foundational speech recognition model, Whisper. We hypothesize that not all neurons are essential for addressing ASR tasks in a specific target language. Hence, eliminating these non-essential neurons could alleviate computational load while maintain-

[*] Equal Contribution
[†] Corresponding author

Proceedings of the The 3rd Workshop on Multi-lingual Representation Learning (MRL), pages 85–94

ing task performance. Furthermore, we postulate that not all layers are language-dependent and question whether incorporating adapters into the text-related layers (decoders) could enhance predicting text token outputs.

In this context, we introduce **PEPSI**, a Parameter-Efficient adaPtation for the Speech foundatIonal model, designed to address ASR tasks for a specific language. We adopt the established PEFT method introduced in Hu et al. (2021) to align the foundational model's knowledge with the target language. Subsequently, we maintain the LoRA adapter attached to the Whisper and remove language-irrelevant neurons.

We emphasize the novelty of our work. While prior studies have focused on pruning models followed by fine-tuning or simultaneous pruning and fine-tuning, we take a step further by identifying language-relevant parameters and retaining adapter-friendly neurons to enable efficient adaptation. Unlike previous research that concentrated on showcasing Whisper's capabilities or enhancing its performance during adaptation, our study addresses the practical concern of reducing computation overhead during adaptation, an aspect that has received limited attention.

Secondly, we identify that the language-relevant components of Whisper are associated with text-related decoders, rather than speech-related encoders. Building on this insight, we pioneer the application of the LoRA adapter to Whisper, exclusively integrating adapters at decoder layers. This is in contrast to prior adapter studies that focused on incorporating adapters throughout all layers of the parent model. Lastly, we introduce PEPSI as an innovative approach that combines LoRA and model pruning to achieve a streamlined utilization of Whisper. Notably, our experimental focus centers on Whisper, the only available open-sourced model that achieves state-of-the-art performance. Through experiments, we confirm the effectiveness of our approach in adapting the Whisper model to a target language or a specific domain that are low-resourced. PEPSI outperforms LoRA and matches FFT, but with significantly less active parameters.

1.2 Key Contributions

- We discover language-specific networks within Whisper, which can be solely utilized to perform comparably to FFT with significant parameter reduction.
- From analyzing the effect of LoRA on different layers, we demonstrate that ASR task relies heavily on text decoder layers, especially on the attention heads.
- Upon the above findings, we propose PEPSI, a novel paradigm to adapt multilingual speech foundational models to a target language.
- We conduct experiments on 5 low-resourced languages to demonstrate that our approach outperforms the commonly used LoRA and matches FFT while reducing the number of parameters up to 50% on specific languages.

2 Related Works

2.1 Automatic Speech Recognition

Automatic Speech Recognition (ASR), or Speech to Text (STT), transcribes a given audio into text. Previous ASR systems utilize RNNs and CNNs as backbone networks to improve performance (Hannun et al., 2014; Schneider et al., 2019). Further research demonstrated that Transformer architecture achieves a competitive recognition rate compared to prior models (Baevski et al., 2019). Recent works following the Scaling Laws (Kaplan et al., 2020) of the NLP domain demonstrated that the same applies to the speech domain; large speech models pre-trained on web-scale data can solve ASR tasks at human standards. An example is Whisper, which effectively addresses the challenge of weakly supervised pre-training by utilizing a large amount of labeled audio data collected from the web. Nevertheless, such models demand high computational complexity and latency due to the scale of their parameters. To address this concern, researchers explore methods to lightly fine-tune the large model to mitigate the cost associated with full fine-tuning large parameter models (Shao et al., 2023; Gong et al., 2023). We share the same goal with the full fine-tuning scheme, but our approach employs distinct methods.

2.2 Parameter-Efficient Fine-Tuning

Several studies have been proposed to rectify the limitations of full fine-tuning when applied to downstream tasks in Pre-trained Language Models(PLMs). Liu et al. (2021) and Li and Liang (2021) optimize the input word embedding by transforming it into a trainable continuous prompt embedding vector. In work by Houlsby et al. (2019), the bottleneck adapter with a transformer-based

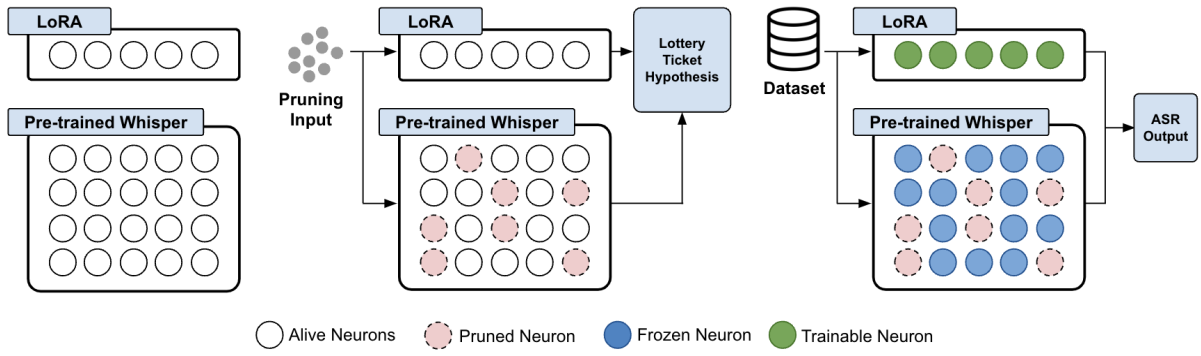


Figure 1: The three steps of PEPSI: (**Left**): Attaching LoRA onto the Whisper model. (**Middle**): Pruning the Whisper neurons irrelevant to the target language; LTH is applied with the pruning input dataset in the target language. (**Right**): Adapting the new language-specific model onto the target dataset.

model was proposed to improve diverse text classification tasks. To concurrently accommodate multiple linguistic target tasks, Bapna and Firat (2019) adds small task-specific adapter layers into the frozen language model. Hu et al. (2021) proposed LoRA, which is trainable low-rank decomposition matrics within PLMs to diminish the trainable parameters for downstream tasks. Our approach adopts a similar strategy to LoRA, utilizing an injected adapter layer. However, while LoRA integrates attention layers into the language model, we enhance the STT performance by integrating a compact adapter module into the decoder.

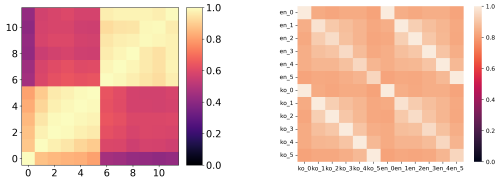
2.3 Pruning

The pruning technique implicates removing unnecessary weights from neural networks, reducing the number of parameters while minimizing the decrease in performance. LeCun et al. (1989) first introduced the pruning technique using second derivatives. Recently, Han et al. (2015) and Frankle and Carbin (2018) showed that by repeatedly removing weights with low magnitudes, the size of image networks can be significantly reduced. In addition, there are various pruning heuristics, such as activations (Hu et al., 2016), redundancy (Mariet and Sra, 2015), per-layer second derivatives (Dong et al., 2017), and energy/computation efficiency (Yang et al., 2017).

The Lottery Ticket Hypothesis (LTH) (Frankle and Carbin, 2018) goes against the shared wisdom of pruning after training (Han et al., 2015). LTH demonstrates the existence of subnetworks that reach similar performance comparable to the original network and are independently trainable from scratch.LTH has been studied in many fields. Early follow-up efforts have been researched in vision tasks (Frankle et al., 2020; Renda et al., 2020). Then, with the emergence of studies proving LTH is applicable in NLP and RL tasks (Renda et al., 2020; Yu et al., 2019), its scope extends. In particular, it is shown that LTH can be applied in Transformer architecture, commonly used as large models in NLP downstream (Chen et al., 2020). Furthermore, the first research, *Audio Lottery*, proposed applying LTH in speech tasks appeared (Ding et al., 2021). Although we share a common topic and scope, the difference lies in that while *Audio Lottery* pruned a model for a single language, we applied the LTH to a multilingual model, Whisper (Radford et al., 2023). Additionally, in contrast to conventional research that conducts pruning on the entire model, our approach involves using a pruning technique that improves the performance of models with adapters attached.

3 Discovering Language-specific Neurons

As preliminary analyses, we investigate the existence of language-specific neurons within Whisper and whether using only these neurons damages the ASR performance on the target language. We conducted two experiments on the widely utilized ASR dataset *Commonvoice 13* (Ardila et al., 2020). We selected 5 languages (i.e., *Korean, Malayalam, Japanese, Swahili, Chinese*) that cover only a small portion in the pre-training data of Whisper, and compared with *English*, a language that covers the most portion.



(a) CKA(ko vs. en) (b) NeuronSim

Figure 2: Visualized representation similarity between different language tokens. Note that (b) is conducted on Whisper's decoder module. Both experiments were conducted using $\text{Whisper}_{\text{Tiny}}$ as the base model.

3.1 Does language token influence the network?

Setup In this study, we investigate the impact of language tokens on both representation and activation patterns within the Whisper model. The prompt utilized in Whisper is as follows: $< |sot| >< |language| >< |task| >< |notimestamps| >$, where $< |language| >$ corresponds to the language token of interest. We alter the language tokens as $< |ko| >$ for *Korean* and $< |en| >$ for *English*, then quantitatively assess the influence of its variations. We employ Centered Kernel Alignment (CKA) (Kornblith et al., 2019) and NeuronSim (Wu et al., 2020) to analyze activation patterns. CKA evaluates representation similarity between layers, producing a score from 0 to 1, while NeuronSim quantifies neuron activation similarity on a scale from 0 to 1, where 0 indicates dissimilarity. It is noteworthy that CKA focuses on representation similarity, whereas NeuronSim concentrates on neuron activation similarity, distinguishing between these two concepts.

Results Figure 2 shows that different patterns are discovered by changing the decoder input of the model under the same audio signal conditions. Comparing the heatmaps of similarity layers, (a) CKA exhibits high level of similarity, whereas (b) NeuronSim reveals a discernible block-diagonal heatmap. We attribute this phenomenon to the Whisper's representation varies depending on the decoder input language. Building upon prior research, we can deduce that two models may have similar representations but different individual neurons (Wu et al., 2020).

	pruned on	Alive params %		
		100.0%	81.0%	65.7%
$\text{Whisper}_{\text{Small}}$	Korean	10.5	10.2	12.9
	Malayalam	10.5	10.8	15.2

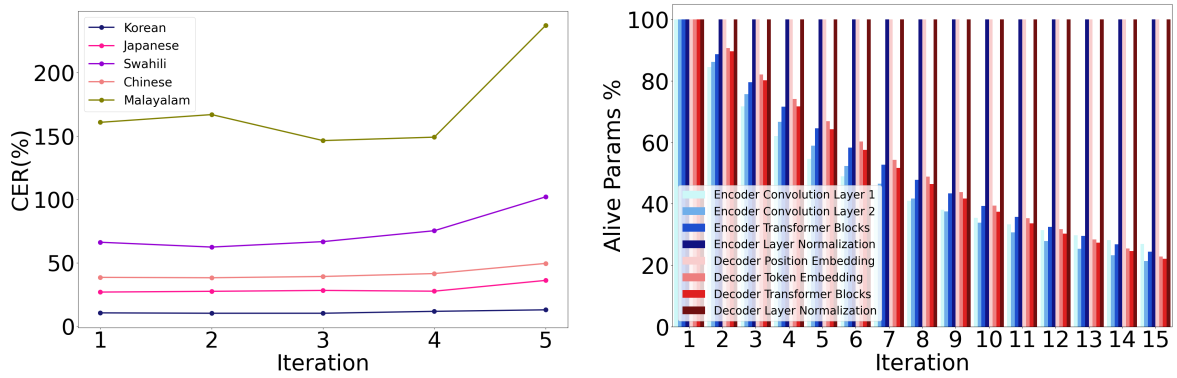
Table 1: Zero-shot CER (%) results on *Korean* when pruned with each language. The 100.0% is the unpruned Whisper model.

3.2 Impact of Pruning Language-irrelevant Neurons

Setup The previous experiment confirmed that each language's parameters are activated differently in Whisper. Therefore, we identify crucial parameters for the specific language and determine if achieving reasonable performance compared with the original model is possible using only these significant parameters. We use $\text{Whisper}_{\text{Small}}$ as our backbone model. We employ iterative weight magnitude pruning (IMP), a widely used algorithm in previous LTH literature (Frankle and Carbin, 2018; Renda et al., 2020; Ding et al., 2021), to detect subnetworks. To identify subnetworks, IMP carries out the following three steps: (1) Train an unpruned model to completion on a dataset $\mathcal{D}$; (2) Remove a portion of unimportant weights with the globally smallest magnitudes; (3) Rewind model weights to θ ($\theta = \theta_{pre}$, the weights from a pre-trained model; or $\theta = \theta_t$, the weights from t training step) and fine-tune the subnetworks to converge. Steps (2) and (3) typically require iterative repetition to discover highly competitive winning tickets. In all experiments, we set $s_i\% = (1 - 0.9^i) \times 100\%$, where i is the number of iterations and s_i is the remaining weights after pruning. We conducted three experiments to identify parameters that operate differently for each language in Whisper.

3.2.1 Results

Language-specific Subnetworks We use LTH to determine if we can identify significant parameters for specific languages in the Whisper model. We pruned the model separately for *Korean* and *Malayalam*, low-resource languages in *Common-voice*. After identifying subnetworks for each language, we conducted zero-shot evaluation on *Korean*. In Table 1, we report our results on CER with $\text{Whisper}_{\text{Small}}$ model. We observe that the model pruned in *Korean* is better than that pruned by *Malayalam* in all subnetworks. Furthermore, the subnetworks exhibit reasonable performance



(a) CER curves for each language

(b) Alive parameters percentage bar chart

Figure 3: **(a) CER curves** for each language. We conduct $\text{Whisper}_{\text{Small}}$ pruned on *Korean* on the *Commonvoice* dataset. Also, we use IMP to prune the model. **(b) Alive parameters percentage bar chart** per iteration for each model layer. We prune $\text{Whisper}_{\text{Small}}$ based on *Korean*.

compared to the unpruned Whisper model. This fact demonstrates that the model pruned in *Korean* has more appropriate parameters for *Korean* data, and we can detect subnetworks for Whisper. In other words, it is evident that there are significant parameters for specific languages in Whisper, and we can identify subnetworks composed of these parameters.

Zero-Shot CER for each Languages Also, in Figure 3(a), we evaluated the zero-shot CER of the model pruned in *Korean* across 5 languages except English, which covers majority of Whisper's pre-training data. We prune the model iteratively at the same ratio to create subnetworks. Then, we calculate each language's zero-shot CER from the subnetworks found at each iteration. As a result, the best CER score is observed in Korean and shows minimal performance drop in all iterations, while other languages exhibit notable performance degradation. These results also mean that essential parameters for specific languages exist within Whisper and can be identified.

Layer-Wise Analysis of Pruning Ratios To gain a more detailed understanding of Whisper pruning, we investigated the pruning ratios for each layer. As shown in Figure 3(b), we divide the model's layers into eight distinct segments, and analyze the pruning ratios of each layer at each iteration. In Figure 3(b), we observe that no pruning occurs in *Encoder Layer Normalization, Decoder Position Embedding,* and *Decoder Layer Normalization*. Furthermore, the trend in the pruned ratio of each layer changes as the iteration progresses. Initially, the encoder convolution layers (i.e., *Encoder Convolution Layer 1* and *Encoder Convolution Layer 2*) are the dominantly pruned layers, while the decoder layers (i.e., *Decoder Token Embedding* and *Decoder Transformer Blocks*) are pruned more significantly as the iteration increases. As a result, we can deduce that subnetworks exist for specific languages, even within the encoder convolution layers responsible for processing audio. Also, we find that the transformer blocks in the decoder layers, which handle text processing, are mainly pruned.

4 Our Method: PEPSI

Upon our findings from above sections, we design and propose PEPSI, a Parameter-Efficient adaptation scheme for the Speech foundational model. We illustrate the overall architecture of our method in Figure 1. As can be seen, our method is composed of three parts. The first phase injects lightweight adapters into the Whisper model for efficient adaptation in the following steps. Next, LTH is conducted to determine the Whisper neurons relevant to a particular language and remove those irrelevant. In the last step, we align the model representation with the distribution of the target language dataset of interest by tuning the adapters injected in the model.

4.1 Injecting Adapters to Whisper

The first part of PEPSI injects a lightweight adapter in the Whisper model for efficient adaptation in the following steps. We adopt LoRA as the adapter architecture as it was shown in Hu et al. (2021) to be the most effective in their works. Whisper follows an encoder-decoder transformer architecture with an audio encoder attached with cross attention to a text decoder. The adapter is injected into the

	KO	ML	JA	SW	ZH-CN	EN
Train	192	509	7,071	34,980	29,383	1,013,968
Test	131	215	4,961	11,271	10,624	16,372

Table 2: Statistics of each language in Commonvoice 13; the abbreviations represent *Korean*, *Malayalam*, *Japanese*, *Swahili*, *Chinese* and *English*, in the respective order.

decoder attention layers following our hypothesis that the text decoder requires further adaptation than the audio encoders for an ASR task. We conduct experiments to verify this hypothesis in the sections to follow.

4.2 Model Pruning

We carry out pruning on the Whisper model parameters to ease the increase in the number of parameters brought by the addition of LoRA. Specifically, LTH is conducted on the Whisper parameters only, without pruning any of the adapter neurons and the Whisper neurons attached to the adapters. This way, the parameters and neurons of Whisper required for connecting with LoRA remains unpruned. The process of pruning follows the previous settings, where we constantly remove unimportant weights every iteration while fine-tuning the model. We prune 50% of Whisper parameters as we figure it is the maximum possible prune percentage to maintain ASR performance on a specific language.

4.3 Tuning LoRA

Through the first and second steps of Adapter Injection and Model Pruning, we obtain a language-specific Whisper model which is able to perform close to the original Whisper without training. Still, the adaptation process on the target language is required to enhance its performance. Hence, we train the pruned model but only the added LoRA adapters for computational efficiency. Low-Rank Adaptation (LoRA) enables training injected intermediate layers within a neural network by optimizing rank decomposition matrices while maintaining the pre-trained Whisper weights in a frozen state—the formulation of adapter in equation 1.

$$output = W(x) + BA(x) \tag{1}$$

where $W(\cdot)$ represents the frozen pre-trained weight, with the weight matrix denoted as $W \in \mathbb{R}^{d\times k}$, matrices $B \in \mathbb{R}^{d\times r}, A \in \mathbb{R}^{r\times k}$, and the rank $r \ll min(d, k)$.

5 Experiments

Setup We conduct experiments to test the effectiveness of our proposed method on 5 low-resourced languages and compare with the high-resourced *English*. We aim to verify 2 objectives in our experiments: 1) To prove our proposed method does indeed bring competitive ASR performances on a specific target language despite the significant reduction in the number of active parameters. 2) To confirm the proposed method eliminates unnecessary neurons for a target language, and the knowledge left in the model is transferable to other datasets of the same language.

Implementation Details Following the prior works of Choi and Park (2022), we evaluate our method on *Commonvoice*, a standard evaluation suite for multilingual ASR models. The detailed statistics of each train/test set is summarized in Table 2. As for the second objective of our experiment, we test the transferability of our pruned model by measuring the ASR performance on a separate dataset with the same language. The model is first pruned with the *Korean* dataset in *Commonvoice*, then adapted to *Clovacall* (Ha et al., 2020) dataset, a Korean speech dataset mainly containing words and phrases from contact centers.

For PEPSI, we use $\text{Whisper}_{\text{Large}}$ as our base model, and prune 50% of its parameters. LoRA is used as the adapter architecture and is added to the attention heads in the text decoder. For the LTH stage, we observe the magnitude change in the Whisper parameters by training the model for 2 epochs with a learning rate of 1e-5. During LoRA adaptation phase, we train the LoRA parameters using the target language set using a learning rate of 1e-3 using the AdamW optimizer.

Baselines We compare the results of PEPSI with the following baselines:

- **Whisper zero-shot**: We compare the ASR performance with zero-shot Whisper, and show the model is not competent to be used as-is for low-resource languages.
- **Whisper Full Fine-tuning**: To test the efficiency of our approach, we compare the number of parameters in comparison to the ASR performance with the standard Whisper FFT.
- **Whisper LoRA**: We compare the number of train/test parameters with the typical LoRA, a widely used PEFT method.

Model	# train param	# test param	KO		ML		JA		SW		ZH-CN		EN	
			CER	WER	CER	WER	CER	WER	CER	WER	CER	WER	CER	WER
whisper zero-shot	-	1.5B	6.71	22.76	102.4	117.8	17.30	96.13	36.02	83.38	25.56	98.70	5.88	11.78
whisper FFT	1.5B	1.5B	**6.12**	**20.54**	**21.67**	**67.78**	**16.88**	**80.52**	**6.72**	**27.53**	**13.56**	**69.33**	**5.78**	**11.45**
whisper LoRA	2.6M	1.5B	6.32	21.33	31.46	76.79	22.36	91.70	11.38	35.46	16.67	73.42	5.81	11.52
whisper LTH	-	0.77B	8.10	30.47	46.89	96.62	30.41	93.44	15.98	38.70	16.12	75.59	6.12	13.22
whisper LTH FT	0.77B	0.77B	7.83	28.67	33.84	84.47	28.38	92.37	14.67	34.51	15.96	83.36	5.99	12.01
OURS	2.6M	0.77B	6.28	21.39	30.96	76.54	18.91	90.31	11.95	35.02	14.03	71.71	5.84	11.52

Table 3: ASR performance comparison of our method (PEPSI) with baselines on each language dataset. We use $\text{Whisper}_{\text{Large}}$ as the base model and prune 50% of its parameters for LTH and PEPSI. The scores are written in %.

- **Whisper LTH**: We apply sole LTH on Whisper using the target language dataset to compare its efficiency with ours. The metric is measured under zero-shot settings after pruning is complete.
- **Whisper LTH FT**: To test the effect of tuning a pruned model, we adapt the Whisper LTH model with the target language dataset.

We observe the effectiveness of each method using the standard CER / WER plus the number of active parameters during training and inference, and the results are summarized in Tables 3 and 4. Note that we set the above methods as baselines as our work is mainly focused on effectively utilizing a multilingual speech foundational model on a specific target language; comparison with monolingual models (Baevski et al., 2020) are beyond the scope of our study.

5.1 Enhanced Parameter Efficiency

Observing the results in Table 3, it is foremost visible that the Whisper model itself exhibits low performance and cannot be utilized as-is for low-resourced languages such as *Malayalam* or *Swahili* while showing supreme performance on the high-resourced *English*. While the FFT scheme on Whisper yields promising results across most datasets, it requires a considerable amount of both training and inference parameters. On the contrary, LoRA achieves error rates almost as low as the FFT paradigm while only requiring the number of parameters corresponding to the adapter itself. Still, it can be observed that LoRA requires more test time parameters than the FFT during inference time. The LTH methods introduced to reduce the test time parameters generally exhibit higher error rates than the abovementioned methods. Our method, PEPSI, mitigates the drawbacks of each work by reducing both train and test time parameters while matching the performance of FFT. As can be seen in Table 3, our method achieves error rates lower than the commonly used LoRA for lower-resourced languages, and shows results comparable to FFT for low-resourced languages.

Model	# train param	# test param	pruned (Y/N)	trained on	CER
whisper zero-shot	-	1.5B	N	-	10.19
whisper FFT	1.5B	1.5B	N	Clovacall	**5.07**
whisper LoRA	2.6M	1.5B	N	Clovacall	6.71
whisper LTH	-	0.77B	Y	-	11.25
whisper LTH FT	0.77B	0.77B	Y	Clovacall	10.75
OURS	2.6M	0.77B	Y	Clovacall	6.29

Table 4: ASR Results on *Clovacall*. For pruned models, the models are pruned on *Commonvoice* Korean then trained on *Clovacall*. The scores are written in %.

5.2 Transferability on Other Datasets

Aside from the performances on *Commonvoice*, we measure the transferability of models pruned on a general speech dataset to a more specific domain with the same language of interest, such as *Clovacall*. Table 4 shows that the Whisper zero-shot shows high error rates on the *Clovacall* dataset, hinting that the domain knowledge for contact centers is not well-formed within the Whisper model itself. The FFT scheme is able to inject the domain knowledge into the model but at high computational costs. LoRA shows comparable results with low training and high inference costs, sharing the identical takeaways from the above experiment. Unlike the original Whisper model, the model pruned on *Commonvoice Korean* causes higher error rates than the original Whisper model under the same zero-shot settings. Fine-tuning the pruned model does lower the error rates, but only to a slight degree. Our method, PEPSI, while sharing the same two phases of pruning and adapting, lowers the error rates further to match that of FFT but with fewer parameters. The result suggests that the mismatching scale of the large-scale Whisper model and a low-resourced language may cause overfitting. It necessitates a more parameter-efficient training scheme such as LoRA to prevent

		# train param	CER	WER
Encoder	fc1	246K	26.77	59.01
	fc2	246K	25.21	57.40
	attn	98K	27.48	60.62
	fc1+attn	344K	27.01	58.71
	fc2+attn	344K	27.58	61.13
Decoder	fc1	246K	24.53	54.28
	fc2	246K	24.35	53.27
	attn	98K	24.11	53.98
	fc1+attn	344K	24.79	53.37
	fc2+attn	344K	24.27	54.68

Table 5: ASR performance of LoRA injected in each layer. *attn* refers to the attention layers while *fc1* and *fc2* refer to the fully connected layers. The scores are written in %.

such phenomena and compression techniques to reduce the model size to match the dataset size.

6 Ablations

6.1 Optimal Injection Point for LoRA

We excavate the optimal positioning approach for integrating the LoRA adapter throughout the Whisper. We assume the adequate adaptation location will differ from the language model to which the original LoRA is applied. In default settings, LoRA is applied to each attention layer in the model. However, we apply the adapters to each attention and MLP layer to discover the optimal injection location. We trained the model on *Commonvoice Korean*. For LoRA parameter settings, we establish the alpha at 64 and the dropout at 0.05. We summarize our results in Table 5.

We find that the components excelling in the encoder differ from those in the decoder. Injecting LoRA in the decoder significantly enhances the STT performance more than the encoder. We presume the underlying reason behind these phenomena is the architectural difference in the Whisper. In this framework, the encoder transforms input audio into a representation vector while the decoder predicts the corresponding text caption.

6.2 Trade-off between Pruned Neurons and Performance

We aim to observe the correlation between the ratio of neurons and performance in the $\text{Whisper}_{\text{Large}}$ model. By measuring the change in zero-shot CER with respect to the increase in prune percentage, we can estimate the ratio of the neurons essential to solving ASR tasks in a particular language. During inference, we apply our proposed PEPSI, which involves applying LTH to the Whisper model alongside LoRA adapters, and we assess its performance

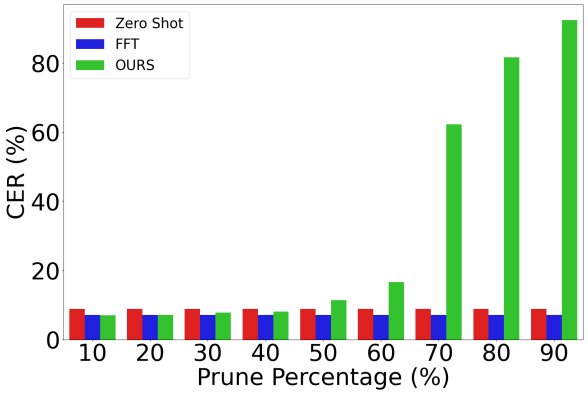


Figure 4: Change in the ASR performance of PEPSI according to the prune percentage.

using the *Commonvoice Korean*. The prune percentage is gradually incremented from 10 to 90, with a step size of 10. For each prune percentage, we conduct IMP with two epochs to obtain the pruning masks. The masks are applied to the updated weights of the Whisper+LoRA model, and the zero-shot performance is measured on the test set of each language; the results are illustrated in Figure 4.

By analyzing the overall trend between prune percentage and CER, we observe that the Whisper model can maintain its performance until approximately 50% of its neurons/parameters are pruned. We assume that 50% of the parameters are composed of the parameters heavily relevant to the target language, plus those containing the general reasoning ability the model gains from large-scale pretraining, as similarly suggested in Lu et al. (2022).

7 Conclusion

In this paper, we proposed PEPSI, a parameter-efficient adaptation strategy for the speech foundation model in low-resource language, demonstrating competitiveness with high-parameter multilingual models. The method incorporates compact adapter modules into the decoder layers of the pretrained model and then eliminates neurons irrelevant to the target language by LTH-based pruning. For adaptation, only the parameters of the added LoRA are updated for efficient tuning. We exhibit the efficiency of our approach by comparing the ASR error rates with existing Whisper baselines in *5* low-resourced languages. We expect our study to serve as a practical guideline for lightweight tuning with speech foundation models and be applied to various low-resource language research.

Limitations

Our method achieves performance surpassing the commonly used LoRA approach with fewer inference parameters. The results are comparable to the standard FFT but with significantly less computational burden. Although our proposed PEPSI exhibits promising results, several improvement avenues exist. While PEPSI applies LoRA with LTH, future works might utilize other adapter architectures or pruning methodologies. Moreover, enhancements to our PEPSI method might involve integration with other speech foundational models, such as USM (Zhang et al., 2023).

Ethics Statement

We hereby clarify that our work complies with ACL Ethics policy. As potential social harms, our method utilizes a well-pretrained Whisper model; thus, any bias or fairness issues in the original pretrained Whisper model can be carried out during our experiments on ASR. We encourage candidate researchers or any users to thoroughly examine the base model to prevent bias and fairness issues.

References

R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 4211–4215.

Alexei Baevski, Steffen Schneider, and Michael Auli. 2019. vq-wav2vec: Self-supervised learning of discrete speech representations. *arXiv preprint arXiv:1910.05453*.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. 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 1538–1548.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. 2020. The lottery ticket hypothesis for pre-trained bert networks. *Advances in neural information processing systems*, 33:15834–15846.

Kwanghee Choi and Hyung-Min Park. 2022. Distilling a pretrained language model to a multilingual asr model. *arXiv preprint arXiv:2206.12638*.

Shaojin Ding, Tianlong Chen, and Zhangyang Wang. 2021. Audio lottery: Speech recognition made ultra-lightweight, noise-robust, and transferable. In *International Conference on Learning Representations*.

Xin Dong, Shangyu Chen, and Sinno Pan. 2017. Learning to prune deep neural networks via layer-wise optimal brain surgeon. *Advances in neural information processing systems*, 30.

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

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. 2020. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR.

Yuan Gong, Sameer Khurana, Leonid Karlinsky, and James Glass. 2023. Whisper-at: Noise-robust automatic speech recognizers are also strong general audio event taggers. *arXiv preprint arXiv:2307.03183*.

Jung-Woo Ha, Kihyun Nam, Jingu Kang, Sang-Woo Lee, Sohee Yang, Hyunhoon Jung, Hyeji Kim, Eunmi Kim, Soojin Kim, Hyun Ah Kim, et al. 2020. Clovacall: Korean goal-oriented dialog speech corpus for automatic speech recognition of contact centers. *arXiv preprint arXiv:2004.09367*.

Song Han, Jeff Pool, John Tran, and William Dally. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Brady Houston and Katrin Kirchhoff. 2023. Exploration of language-specific self-attention parameters for multilingual end-to-end speech recognition. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 755–762. IEEE.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. 2016. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. *arXiv preprint arXiv:1607.03250*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR.

Yann LeCun, John Denker, and Sara Solla. 1989. Optimal brain damage. *Advances in neural information processing systems*, 2.

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.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. Gpt understands, too. *arXiv preprint arXiv:2103.10385*.

Yizhou Lu, Mingkun Huang, Xinghua Qu, Pengfei Wei, and Zejun Ma. 2022. Language adaptive cross-lingual speech representation learning with sparse sharing sub-networks. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6882–6886. IEEE.

Zelda Mariet and Suvrit Sra. 2015. Diversity networks: Neural network compression using determinantal point processes. *arXiv preprint arXiv:1511.05077*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Alex Renda, Jonathan Frankle, and Michael Carbin. 2020. Comparing rewinding and fine-tuning in neural network pruning. *arXiv preprint arXiv:2003.02389*.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

Hang Shao, Wei Wang, Bei Liu, Xun Gong, Haoyu Wang, and Yanmin Qian. 2023. Whisper-kdq: A lightweight whisper via guided knowledge distillation and quantization for efficient asr. *arXiv preprint arXiv:2305.10788*.

Abhayjeet Singh, Arjun Singh Mehta, Jai Nanavati, Jesuraja Bandekar, Karnalius Basumatary, Sandhya Badiger, Sathvik Udupa, Saurabh Kumar, Prasanta Kumar Ghosh, Priyanka Pai, et al. 2023. Model adaptation for asr in low-resource indian languages. *arXiv preprint arXiv:2307.07948*.

Zirui Wang, Zachary C Lipton, and Yulia Tsvetkov. 2020. On negative interference in multilingual models: Findings and a meta-learning treatment. *arXiv preprint arXiv:2010.03017*.

John Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2020. Similarity analysis of contextual word representation models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4638–4655.

Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze. 2017. Designing energy-efficient convolutional neural networks using energy-aware pruning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5687–5695.

Haonan Yu, Sergey Edunov, Yuandong Tian, and Ari S Morcos. 2019. Playing the lottery with rewards and multiple languages: lottery tickets in rl and nlp. *arXiv preprint arXiv:1906.02768*.

Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, et al. 2023. Google usm: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*.