Fast Adaptation and Robust Quantization for Multi-Modal Foundation Models from Associative Memory: A Case Study in SpeechLM

Shang Wu [*1] **Yen-Ju Lu** [*2] **Haozheng Luo** [*1] **Jerry Yao-Chieh Hu** [1] **Jiayi Wang** [1] **Najim Dehak** [2] **Jesus Villalba** [2] **Han Liu** [1]

Abstract

We present a preliminary investigation into the outlier problem in the multi-modal foundation model, focusing on SpeechLM. Specifically, we consider SpeechLM models that use a pretrained Language Model (LM) as the backbone and are fine-tuned on multi-modal data (speech and text). There is an outlier problem in pretrained LLMs and multi-modal inputs in SpeechLM. By adopting a principled approach inspired by associative memory models to address the outlier problem, we achieve significant improvements in *faster* low-rank adaptation, *more accurate* cross-modal fine-tuning, and *more robust* post-training quantization. Methodologically, we implement an outlier-efficient Hopfield layer to replace the conventional transformer attention mechanism. This adjustment effectively removes outliers, improving the performance in multi-modal adaptation and inference with a quantized model. Our proposed framework yields an average performance improvement of **7.98%** in cross-modal fine-tuning and **67.85%** in quantization, significantly outperforming standard frameworks in these respects.

1. Introduction

We propose to utilize an outlier-efficient Hopfield layer to tackle the outlier problem in pretrained LLMs and multi-modal fine-tuning in the current SpeechLM framework. This approach leads to faster adaptation, better multi-modal fine-tuning, and more robust post-training quantization. SpeechLM employs pretrained language models to enhance speech recognition and synthesis, revolutionizing natural language understanding and generation (Nguyen et al., 2024). Adapting LLMs to speech efficiently and effectively remains challenging (Mehrish et al., 2023) due to the inherent differences between text and speech modalities. This process often requires extensive *full(-parameter)* fine-tuning and resources (Zhang et al., 2024). Standard fine-tuning and quantization techniques, like low-rank adaptation and post-training quantization (PTQ), result in significant performance degradation when applied to the SpeechLM framework (Latif et al., 2023). We observe that the main cause of this degradation is the challenges posed by outliers. These outliers come from transformer-based models (Clark et al., 2019; Kovaleva et al., 2019) and multi-modal input in the SpeechLM framework (Wei et al., 2023; 2022).

We replace conventional transformer attention mechanisms with an outlier-efficient Hopfield layer (Hu et al., 2024a), known for its robust associative memory capabilities (Hu et al., 2024b; Xu et al., 2024a; Wu et al., 2024a;b; Hu et al., 2023; Ramsauer et al., 2020). The Outlier-Efficient Hopfield (`OutEffHop`) layer effectively identifies and filters out outliers that typically occur during the pretraining and low-rank adaptation (LoRA) phases (Hu et al., 2024a).

This strategic modification achieves a "triple win" for the model's optimization and application: speeding up the adaptation processes, boosting the accuracy of low-rank adaptation training, and strengthening the model's robustness during post-training quantization. Overall, our proposed multi-modal SpeechLM framework enhances both performance and versatility across a range of speech-related tasks.

2. Methodology

This section introduces the proposed method, consisting of the SpeechLM system, the Outlier-Efficient architecture, and deployment of PTQ methods on SpeechLM system.

2.1. Proposed Method on SpeechLM System

We implement a straightforward design for the SpeechLM architecture, building on the approaches detailed in (Nguyen et al., 2024; Maiti et al., 2024). Our design employs a

[*]Equal contribution [1]Northwestern University, Evanston, USA. [2]Johns Hopkins University, Baltimore, USA.. Correspondence to: Shang Wu <swu@u.northwestern.edu>, Yen-Ju Lu <ylu125@jhu.edu>, Haozheng Luo <hluo@u.northwestern.edu>, Jerry Yao-Chieh Hu <jhu@u.northwestern.edu>, Jiayi Wang <JiayiWang2020@u.northwestern.edu>, Najim Dehak <ndehak3@jhu.edu>, Jesus Villalba <jvillal7@jhu.edu>, Han Liu <hanliu@northwestern.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

HuBERT (Hsu et al., 2021) model combined with k-means clustering to transform speech into discrete tokens $s_i \in \mathcal{V}_{\text{sp}}$, which are then merged with text characters $t_i \in \mathcal{V}_{\text{txt}}$ to create a unified vocabulary $\mathcal{V}_{\text{joint}}$. The joint probability of speech and text tokens J is calculated as:

$$p(J) = \prod_{i=1}^{l} p(j_i \mid j_1, \cdots, j_{i-1}).$$

In our architecture, the input token J is processed through an embedding layer, followed by L layers of a transformer decoder. The final layer produces a probability distribution across the unified vocabulary $\mathcal{V}_{\text{joint}}$. Training of this model is conducted autoregressively using ground truth word as input, where at each timestep i, the model uses the actual previous output instead of its own prediction to produce a predicted distribution as:

$$\widehat{p}^{i} = \text{Ours}(j_1, \ldots, j_{i-1}).$$

The cross-entropy loss is calculated as:

$$L_{\text{CE}}(p_i, \widehat{p}^{i}) = -\sum_{c=1}^{|\mathcal{V}_{\text{joint}}|} p_i(c) \log \widehat{p}^{i}(c).$$

2.2. Outlier Efficient Architecture

We discuss the outliers challenge in current transformer-based language model (e.g, OPT) and elaborate on our proposed outlier-efficient method.

Outliers Challenge in Transformer Architecture. As reported by Bondarenko et al. (2024); Hu et al. (2024a), no-op tokens (Clark et al., 2019) that have small value vectors often receive disproportionately large attention weights. This behavior increases the computational and memory requirements during training and lowers the robustness of model quantization. A detailed discussion of this outlier challenge is presented in Appendix B.

Outlier-Efficient Method. The Outlier-Efficient Method targets the reduction of outlier effects during model pretraining (Hu et al., 2024a), fine-tuning (Chen et al., 2024), and deployment phases (Hu et al., 2024a; Bondarenko et al., 2024; Xiao et al., 2023a). Utilizing the Softmax_1 function within the attention mechanism is a proven strategy for managing outliers effectively (Hu et al., 2024a; Miller, 2023) and demonstrated in the equation below:

$$\text{Softmax}_1(a_i) = \frac{\exp(a_i)}{1 + \sum_{j=1}^{m} \exp(a_j)},$$

for $a = (a_1, \ldots, a_m) \in \mathbb{R}^m$ and m is the number of sequence length in multi-modality data. In practical applications, similar to implementations in frameworks like PyTorch (Paszke et al., 2019), a normalization step adjusts the input vector $a \in \mathbb{R}^L$ by subtracting its maximum value prior to applying the Softmax_1 function. This adjustment ensures numerical stability of the Softmax_1 activation.

2.3. Post-Training Quantization on SpeechLM System

We deploy the post-training quantization methods on the large language model (LLM) in SpeechLM system. The other modules in SpeechLM system, such as speech decoders and speech tokenizers, are not quantized since current PTQ methods are not designed for these modules. The deployment of quantization on the LLM system in the SpeechLM system can speed up the inference latency as shown in Table 4.

3. Experimental Studies

In this section, we conduct a set of experiments to validate the effectiveness of our proposed framework. Specifically, we evaluate the performance of our method against state-of-the-art (SOTA) methods outlined in (Maiti et al., 2024).

Models. Following Maiti et al. (2024), we validate our method with 2 different size of the Open Pretrained Transformer (OPT) pretrained models, OPT-125m and OPT-350m. We use the same BPE tokenizer as (Maiti et al., 2024) for four different tasks, including textLM, speechLM, ASR, and TTS. For OPT models equipped with the proposed method, we follow the same pre-training procedure as (Hu et al., 2024a).

Datasets. We use four datasets: LibriLight (LL) (Kahn et al., 2020), Librispeech (Panayotov et al., 2015), LibriTTS (LT) (Zen et al., 2019), VCTK (VC) (Veaux et al., 2017) for textLM, speechLM, ASR, and TTS. For **textLM**, we use Librispeech (Panayotov et al., 2015), working with 40 million text utterances. We use the speech data in LibriLight (LL) (Kahn et al., 2020), featuring 60,000 hours of audiobook recordings from 7,000 different speakers, totaling 12 million utterances for **SpeechLM**. For ASR, we use English Multilingual Librispeech (MLS) (Pratap et al., 2020) dataset. (Zen et al., 2019), VCTK (VC) (Veaux et al., 2017) are used for **TTS** (Text-to-Speech) tasks. We experiment with those datasets for OPT-125m and OPT-350m.

Evaluation Metrics. For speech and text generation, we evaluate models using perplexity (PPL) for those with the same vocabulary size. In automatic speech recognition (ASR) tasks, we measure performance with the Word Error Rate (WER). For text-to-speech (TTS) tasks, Hifi-gan (Kong et al., 2020) serves as the vocoder, and we assess intelligibility using the character error rate (CER) from whisper decoding results. All these metrics aim for lower scores to indicate better performance. For the accuracy metric, the higher means better performance and we give more details in the ablation study (Appendix E). Besides, we calculate and compare the average performance drop across all tasks for each method to assess their impact.

Table 1. **Comparing Ours with Vanilla Method in a Post-Training Quantization (PTQ) Setting.** We conduct experiments on proposed method with vanilla method across three quantization methods (SmoothQuant, AffineQuant, OmmiQuant) and two quantization configurations (Weight-8bit-Activation-8bit and Weight-4bit-Activation-4bit). The evaluation metrics include Text Perplexity (PPL), SpeechLM PPL, Word Error Rate (WER) in Automatic Speech Recognition (ASR), and Character Error Rate (CER) in Text-to-Speech (TTS). We also measure the average performance drop after quantization to assess the efficiency of proposed method in the PTQ setting. In most configurations, proposed method results in a smaller performance drop after quantization compared to vanilla method.

Model	Method	#Bits	Quantization Method	TextLM PPL (↓)	SpeechLM PPL (↓)	ASR WER (↓)	TTS CER (↓)	Avg Performance Drop (↓)
OPT-125m	Vanilla	16W/16A	-	22.56	59.42	12.40	12.08	-
		8W/8A	SmoothQuant	22.63	59.53	12.46	12.38	0.85%
		8W/8A	AffineQuant	22.61	59.52	12.42	12.37	0.74%
		8W/8A	OmmiQuant	22.62	59.53	12.44	12.38	0.81%
		4W/4A	SmoothQuant	45.23	96.87	52.31	48.79	197.31%
		4W/4A	AffineQuant	31.25	80.19	29.44	28.34	86.37%
		4W/4A	OmmiQuant	31.28	80.21	31.98	29.55	94.04%
	Ours	16W/16A	-	22.70	59.45	12.61	12.11	-
		8W/8A	SmoothQuant	22.71	59.49	12.64	12.18	0.23%
		8W/8A	AffineQuant	22.71	59.48	12.62	12.13	**0.08%**
		8W/8A	OmmiQuant	22.71	59.49	12.63	12.14	0.13%
		4W/4A	SmoothQuant	37.14	84.55	35.32	36.73	112.05%
		4W/4A	AffineQuant	26.11	68.42	14.33	15.71	**18.52%**
		4W/4A	OmmiQuant	26.12	68.63	14.53	16.01	19.63%
OPT-350m	Vanilla	16W/16A	-	14.01	45.08	17.76	11.59	-
		8W/8A	SmoothQuant	14.08	45.10	17.82	12.25	1.64%
		8W/8A	AffineQuant	14.04	45.08	17.78	12.18	1.35%
		8W/8A	OmmiQuant	14.05	45.09	17.80	12.20	1.45%
		4W/4A	SmoothQuant	38.51	87.34	67.87	47.28	214.62%
		4W/4A	AffineQuant	26.92	64.37	43.21	29.81	108.86%
		4W/4A	OmmiQuant	26.99	64.40	43.38	30.15	109.98%
	Ours	16/16A	-	14.04	45.38	17.81	12.53	-
		8W/8A	SmoothQuant	14.04	45.45	17.82	12.59	0.17%
		8W/8A	AffineQuant	14.04	45.44	17.81	12.55	**0.07%**
		8W/8A	OmmiQuant	14.05	45.47	17.81	12.57	0.15%
		4W/4A	SmoothQuant	24.77	62.61	37.57	32.45	96.08%
		4W/4A	AffineQuant	22.54	51.13	22.17	17.24	**33.83%**
		4W/4A	OmmiQuant	22.59	51.26	23.01	18.11	36.90%

3.1. Post-Training Quantization (PTQ)

To assess the performance of our method, we replace the standard attention layer in OPT models (Zhang et al., 2022) with the $\mathrm{Softmax}_1$ activation function (Vaswani et al., 2017). We utilize pretrained OPT model checkpoints modified with the proposed method (Hu et al., 2024a) and fine-tune them at full rank following the approach described in (Maiti et al., 2024). Our evaluation involves testing the models on datasets using FP16 (16-bit floating point) and conducting post-training quantization (PTQ) to measure the performance impact.

Baselines. Following Maiti et al. (2024), we validate our method with 3 different quantization methods: SmoothQuant (Xiao et al., 2023a), AffineQuant (Ma et al., 2024), and OmniQuant (Shao et al., 2023). We employ the hyperparameters specified in the original studies for each quantization method. For SmoothQuant, we implement the guidelines detailed in (Xiao et al., 2023a). The parameters for AffineQuant are applied as described in (Ma et al., 2024). Regarding OmniQuant, we utilize the hyperparameters outlined in (Shao et al., 2023).

Results. Referring to Table 1, our framework demonstrates superior performance over the standard training framework under W4A4 and W8A8 post-training quantization conditions, employing state-of-the-art PTQ methods. Specifically, with both weights and activations quantized to 8 bits (W8A8), our framework shows a negligible average performance decline of only **0.08%** on OPT-125m and **0.07%** on OPT-350m. Moreover, in the W4A4 setting, the standard framework suffers a significant performance drop—over 197% on OPT-125m and 214% on OPT-350m. Our framework reduces this impact, with declines of only **18.52%** on OPT-125m and **36.61%** on OPT-350m.

3.2. Low-Rank Adaptation Methods

We evaluate our framework with Low-Rank Adaptation (LoRA) methods, designed to increase fine-tuning efficiency using fewer parameters. Our method undergoes comparison with the standard version across various LoRA approaches.

LoRA Methods. We compare our method with the vanilla method across 2 different LoRA methods: Lora (Hu et al., 2021) and QLoRA (Dettmers et al., 2024). For the full fine-funing method, we fine-tune the model with full-rank using mixed-precision training. For the LoRA method, following Hu et al. (2021), we fine-tune the model with low-rank adaptations using a rank of 128 and an alpha value of 256. For the QLoRA method, following Dettmers et al. (2024),

Table 2. **Comparing Ours with Vanilla Method in a Low-Rank Adaptation Setting.** We conduct experiments on proposed method with vanilla method across two Low-Rank Adaptation methods (LoRA, QLoRA). The evaluation metrics include Text Perplexity (PPL), SpeechLM PPL, Word Error Rate (WER) in Automatic Speech Recognition (ASR), and Character Error Rate (CER) in Text-to-Speech (TTS). We also measure the average performance drop after low-rank adaptation to assess the performance of proposed method in the low-rank adaptation setting. In most configurations, proposed method results in better low-rank adaptation performance compared to vanilla method.

Model	Method	Quantization Method	TextLM PPL (↓)	SpeechLM PPL (↓)	ASR WER (↓)	TTS CER (↓)	Average Performance Drop (↓)
OPT-125m	Vanilla	Full	22.56	59.42	12.40	12.08	-
		Lora	25.69	62.16	12.39	15.47	11.61%
		QLora	25.97	62.43	12.86	15.02	12.06%
	Ours	Full	22.58	59.46	12.61	12.11	-
		Lora	25.77	62.23	12.56	11.80	**3.96%**
		QLora	25.77	62.23	13.42	12.46	**7.03%**
OPT-350m	Vanilla	Full	14.01	45.08	17.76	11.59	-
		Lora	16.84	50.36	20.57	23.27	37.08%
		QLora	16.17	48.99	24.06	26.38	46.79%
	Ours	Full	14.04	45.38	17.81	12.53	-
		Lora	16.17	49.44	21.78	23.57	**33.61%**
		QLora	16.01	48.39	24.42	22.88	**35.10%**

we fine-tune the model with quantized low-rank adaptations, maintaining the same rank and alpha value as specified in LoRA, but using Int8 (Dettmers et al., 2022a) quantization instead of 4-bit NormalFloat (NF4) (Dettmers et al., 2024).

Table 3. **Comparison of Different Ranks Using LoRA.** We compare the validation accuracy (detailed in Appendix E) between full fine-tuning with LoRA in different ranks (128, 256, 512).

Method	Fine-Tuning Method	Rank	Val Acc (%)
Vanilla	Full Fine-Tuning	N/A	**30.5**
Ours	Full Fine-Tuning	N/A	30.2
Vanilla	LoRA	512	27.6
Ours	LoRA	512	**27.8**
Vanilla	LoRA	256	28.1
Ours	LoRA	256	**28.9**
Vanilla	LoRA	128	27.5
Ours	LoRA	128	**27.5**

Results. In Table 2, our results confirm the effectiveness of proposed method in low-rank adaptation, showing substantial performance improvements in most configurations. Specifically, proposed method achieves an average performance gain of **7.98%** over the standard framework for low-rank adaptation tasks, with notable success in boosting the OPT-350m model's performance. Comparing to OPT-125m vanilla model, our proposed method exhibits smaller performance declines in LoRA and QLoRA. This trend is consistent with findings in (Hu et al., 2024a) that larger models like the OPT-350m are more affected by significant outliers.

Training Curve and Inference Comparison. To assess the performance of our proposed method versus the conventional (vanilla) method, we monitored the training curves for both approaches. As shown in Figure 1, our method delivers superior performance compared to the vanilla method within the same number of training epochs.

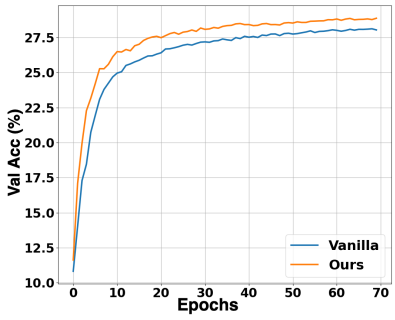

Figure 1. **Validation Accuracy vs. Training Epochs between Ours and Vanilla.** We visualize the accuracy versus epochs curve on both proposed framework and vanilla framework. We can observe that the proposed framework can always achieve higher accuracy under the same epoch across the whole training process.

We evaluate the end-to-end latency of the SpeechLM system by comparing our proposed method with the vanilla method, using validation data with a batch size of 1. The results, presented in Table 4, demonstrate that our method does not increase the overall latency of the SpeechLM system compared to the vanilla method in both the 16W/16A and 8W/8A settings. Our method reducing the performance drop when PTQ methods are applied to the entire SpeechLM system without adding extra latency.

Table 4. **Comparing Ours with Vanilla Method in Inference Latency.** We perform experiments comparing our proposed method with the vanilla method regarding inference latency for FP16 and 8-bit SmoothQuant. We evaluate the inference latency on TextLM and SpeechLM tasks. In most configurations, our method demonstrates latency similar to the vanilla method.

Model	Method	Quantization Method	#Bits	Latency in TextLM (ms)	Latency in SpeechLM (ms)
OPT-125m	Vanilla	N/A	16W/16A	15.88	15.93
		SmoothQuant	8W/8A	12.31	12.48
	Ours	N/A	16W/16A	16.12	16.21
		SmoothQuant	8W/8A	12.44	12.52
OPT-350m	Vanilla	N/A	16W/16A	16.89	18.32
		SmoothQuant	8W/8A	13.88	14.56
	Ours	N/A	16W/16A	17.31	19.82
		SmoothQuant	8W/8A	14.11	14.94

4. Discussion and Conclusion

We present proposed method, an outlier-robust multi-modal foundation model for speech-text tasks, designed to address the computational challenges posed by outlier effects in modality fusion and cross-modality adaptation of SpeechLM. Our solution not only mitigates the impact of outliers in transformer-based models but also enhances both low-rank adaptation and post-training quantization performance. Experimentally, the proposed method achieves an average performance improvement of **7.98%** in cross-modal fine-tuning (Section 3.2) and **67.85%** in quantization (Section 3.1), compared to existing methods.

Broader Impact

We believe this methodology offers an opportunity to enhance the fine-tuning and inference processes of foundation models, including low-rank adaptation and post-training quantization, by leveraging insights from associative memory models. Our solution also enables large foundation models to perform edge computing and facilitates model fine-tuning without requiring extensive resources. However, this approach might intensify biases present in the training data, potentially resulting in unfair or discriminatory outcomes for underrepresented groups.

Acknowledgments

The authors would would like to thank to Jing Liu for enlightening discussions on related topics. Also, the authors would like to thank the anonymous reviewers and program chairs for constructive comments.

HL is partially supported by NIH R01LM1372201. This research was supported in part through the computational resources and staff contributions provided for the Quest high performance computing facility at Northwestern University, which is jointly supported by the Office of the Provost, the Office of Research, and Northwestern University Information Technology. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

Josh Alman and Zhao Song. How to capture higher-order correlations? generalizing matrix softmax attention to kronecker computation. *arXiv preprint arXiv:2310.04064*, 2023a.

Josh Alman and Zhao Song. Fast attention requires bounded entries. In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)*, 2023b.

Josh Alman and Zhao Song. The fine-grained complexity of gradient computation for training large language models. *arXiv preprint arXiv:2402.04497*, 2024.

Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, et al. Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. *arXiv preprint arXiv:2110.07205*, 2021.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Understanding and overcoming the challenges of efficient transformer quantization, 2021.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Quantizable transformers: Removing outliers by helping attention heads do nothing. *Advances in Neural Information Processing Systems (NeurIPS)*, 36, 2023.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Quantizable transformers: Removing outliers by helping attention heads do nothing. *Advances in Neural Information Processing Systems*, 36, 2024.

Jiun-Man Chen, Yu-Hsuan Chao, Yu-Jie Wang, Ming-Der Shieh, Chih-Chung Hsu, and Wei-Fen Lin. Quanttune: Optimizing model quantization with adaptive outlier-driven fine tuning, 2024.

Zhehuai Chen, Yu Zhang, Andrew Rosenberg, Bhuvana Ramabhadran, Pedro Moreno, Ankur Bapna, and Heiga Zen. Maestro: Matched speech text representations through modality matching. *arXiv preprint arXiv:2204.03409*, 2022.

Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250. IEEE, 2021.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*, 2019.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in Neural Information Processing Systems*, 35:30318–30332, 2022a.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3.int8(): 8-bit matrix multiplication for transformers at scale. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 30318–30332. Curran Associates, Inc., 2022b.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

Yeqi Gao, Zhao Song, Weixin Wang, and Junze Yin. A fast optimization view: Reformulating single layer attention in llm based on tensor and svm trick, and solving it in matrix multiplication time. *arXiv preprint arXiv:2309.07418*, 2023.

Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. In *Low-Power Computer Vision*, pages 291–326. Chapman and Hall/CRC, 2022.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023.

Jiuxiang Gu, Chenyang Li, Yingyu Liang, Zhenmei Shi, Zhao Song, and Tianyi Zhou. Fourier circuits in neural networks: Unlocking the potential of large language models in mathematical reasoning and modular arithmetic. *arXiv preprint arXiv:2402.09469*, 2024a.

Jiuxiang Gu, Yingyu Liang, Heshan Liu, Zhenmei Shi, Zhao Song, and Junze Yin. Conv-basis: A new paradigm for efficient attention inference and gradient computation in transformers. *arXiv preprint arXiv:2405.05219*, 2024b.

Jiuxiang Gu, Yingyu Liang, Zhenmei Shi, Zhao Song, and Yufa Zhou. Tensor attention training: Provably efficient learning of higher-order transformers. *arXiv preprint arXiv:2405.16411*, 2024c.

Michael Hassid, Tal Remez, Tu Anh Nguyen, Itai Gat, Alexis Conneau, Felix Kreuk, Jade Copet, Alexandre Defossez, Gabriel Synnaeve, Emmanuel Dupoux, et al. Textually pretrained speech language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Tomoki Hayashi and Shinji Watanabe. Discretalk: Text-to-speech as a machine translation problem. *arXiv preprint arXiv:2005.05525*, 2020.

Mark Horowitz. 1.1 computing's energy problem (and what we can do about it). In *2014 IEEE international solid-state circuits conference digest of technical papers (ISSCC)*, pages 10–14. IEEE, 2014.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Jerry Yao-Chieh Hu, Pei-Hsuan Chang, Robin Luo, Hong-Yu Chen, Weijian Li, Wei-Po Wang, and Han Liu. Outlier-efficient hopfield layers for large transformer-based models. In *Forty-first International Conference on Machine Learning (ICML)*, 2024a.

Jerry Yao-Chieh Hu, Thomas Lin, Zhao Song, and Han Liu. On computational limits of modern hopfield models: A fine-grained complexity analysis. In *Forty-first International Conference on Machine Learning (ICML)*, 2024b.

Jerry Yao-Chieh Hu, Maojiang Su, En-Jui Kuo, Zhao Song, and Han Liu. Computational limits of low-rank adaptation (lora) for transformer-based models. *arXiv preprint arXiv:2406.03136*, 2024c.

Xijie Huang, Zechun Liu, Shih-Yang Liu, and Kwang-Ting Cheng. Rolora: Fine-tuning rotated outlier-free llms for effective weight-activation quantization. *arXiv preprint arXiv:2407.08044*, 2024.

Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

Jacob Kahn, Morgane Riviere, Weiyi Zheng, Evgeny Kharitonov, Qiantong Xu, Pierre-Emmanuel Mazaré, Julien Karadayi, Vitaliy Liptchinsky, Ronan Collobert, Christian Fuegen, et al. Libri-light: A benchmark for asr with limited or no supervision. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673. IEEE, 2020.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms, 2020.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033, 2020.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. Revealing the dark secrets of bert. *arXiv preprint arXiv:1908.08593*, 2019.

Siddique Latif, Moazzam Shoukat, Fahad Shamshad, Muhammad Usama, Heriberto Cuayáhuitl, and Björn W Schuller. Sparks of large audio models: A survey and outlook. *arXiv preprint arXiv:2308.12792*, 2023.

Ann Lee, Hongyu Gong, Paul-Ambroise Duquenne, Holger Schwenk, Peng-Jen Chen, Changhan Wang, Sravya Popuri, Yossi Adi, Juan Pino, Jiatao Gu, et al. Textless speech-to-speech translation on real data. *arXiv preprint arXiv:2112.08352*, 2021.

Yixiao Li, Yifan Yu, Chen Liang, Pengcheng He, Nikos Karampatziakis, Weizhu Chen, and Tuo Zhao. Loftq: Lora-fine-tuning-aware quantization for large language models, 2023.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

Ke-Han Lu, Zhehuai Chen, Szu-Wei Fu, He Huang, Boris Ginsburg, Yu-Chiang Frank Wang, and Hung-yi Lee. Desta: Enhancing speech language models through descriptive speech-text alignment. *arXiv preprint arXiv:2406.18871*, 2024.

Haozheng Luo, Ruiyang Qin, Chenwei Xu, Guo Ye, and Zening Luo. Open-ended multi-modal relational reasoning for video question answering. In *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 363–369. IEEE, 2023a.

Yukui Luo, Nuo Xu, Hongwu Peng, Chenghong Wang, Shijin Duan, Kaleel Mahmood, Wujie Wen, Caiwen Ding, and Xiaolin Xu. Aq2pnn: Enabling two-party privacy-preserving deep neural network inference with adaptive quantization. In *Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 628–640, 2023b.

Yuexiao Ma, Huixia Li, Xiawu Zheng, Feng Ling, Xuefeng Xiao, Rui Wang, Shilei Wen, Fei Chao, and Rongrong Ji. Affinequant: Affine transformation quantization for large language models. *arXiv preprint arXiv:2403.12544*, 2024.

Soumi Maiti, Yifan Peng, Shukjae Choi, Jee-weon Jung, Xuankai Chang, and Shinji Watanabe. Voxtlm: Unified decoder-only models for consolidating speech recognition, synthesis and speech, text continuation tasks. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13326–13330. IEEE, 2024.

M. Marchesi, G. Orlandi, F. Piazza, and A. Uncini. Fast neural networks without multipliers. *IEEE Transactions on Neural Networks*, 4(1):53–62, 1993. doi: 10.1109/72.182695.

Ambuj Mehrish, Navonil Majumder, Rishabh Bharadwaj, Rada Mihalcea, and Soujanya Poria. A review of deep learning techniques for speech processing. *Information Fusion*, page 101869, 2023.

Cristian Meo, Ksenia Sycheva, Anirudh Goyal, and Justin Dauwels. Bayesian-lora: Lora based parameter efficient fine-tuning using optimal quantization levels and rank values trough differentiable bayesian gates. *arXiv preprint arXiv:2406.13046*, 2024.

Evan Miller. Blog post: Attention is off by one, 2023. URL https://www.evanmiller.org/attention-is-off-by-one.html. Accessed: August 4, 2023.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R Costa-Jussa, Maha Elbayad, Sravya Popuri, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, et al. Spirit-lm: Interleaved spoken and written language model. *arXiv preprint arXiv:2402.05755*, 2024.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, and Ronan Collobert. Mls: A large-scale multilingual dataset for speech research. *arXiv preprint arXiv:2012.03411*, 2020.

Qian qian Dong, Zhiying Huang, Qiao Tian, Chen Xu, Tom Ko, Siyuan Feng, Tang Li, Kexin Wang, Xuxin Cheng, Fengpeng Yue, et al. Polyvoice: Language models for speech to speech translation. In *The Twelfth International Conference on Learning Representations*, 2023.

Ruiyang Qin, Dancheng Liu, Zheyu Yan, Zhaoxuan Tan, Zixuan Pan, Zhenge Jia, Meng Jiang, Ahmed Abbasi, Jinjun Xiong, and Yiyu Shi. Empirical guidelines for deploying llms onto resource-constrained edge devices. *arXiv preprint arXiv:2406.03777*, 2024a.

Ruiyang Qin, Zheyu Yan, Dewen Zeng, Zhenge Jia, Dancheng Liu, Jianbo Liu, Zhi Zheng, Ningyuan Cao,

Kai Ni, Jinjun Xiong, et al. Robust implementation of retrieval-augmented generation on edge-based computing-in-memory architectures. *arXiv preprint arXiv:2405.04700*, 2024b.

Hubert Ramsauer, Bernhard Schafl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlovic, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Karan Samel, Zelin Zhao, Binghong Chen, Kuan Wang, Robin Luo, and Le Song. How to design sample and computationally efficient vqa models. *arXiv preprint arXiv:2103.11537*, 2021.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. Omniquant: Omnidirectionally calibrated quantization for large language models. *arXiv preprint arXiv:2308.13137*, 2023.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*, 2024.

Chuan Zhang Tang and Hon Keung Kwan. Multilayer feedforward neural networks with single powers-of-two weights. *IEEE Transactions on Signal Processing*, 41(8): 2724–2727, 1993.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Christophe Veaux, Junichi Yamagishi, and Kirsten MacDonald. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. 2017.

Mingqiu Wang, Wei Han, Izhak Shafran, Zelin Wu, Chung-Cheng Chiu, Yuan Cao, Nanxin Chen, Yu Zhang, Hagen Soltau, Paul K Rubenstein, et al. Slm: Bridge the thin gap between speech and text foundation models. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE, 2023a.

Neng Wang, Hongyang Yang, and Christina Dan Wang. Fingpt: Instruction tuning benchmark for open-source large language models in financial datasets. *arXiv preprint arXiv:2310.04793*, 2023b.

Xiuying Wei, Yunchen Zhang, Xiangguo Zhang, Ruihao Gong, Shanghang Zhang, Qi Zhang, Fengwei Yu, and Xianglong Liu. Outlier suppression: Pushing the limit of low-bit transformer language models. *Advances in Neural Information Processing Systems*, 35:17402–17414, 2022.

Xiuying Wei, Yunchen Zhang, Yuhang Li, Xiangguo Zhang, Ruihao Gong, Jinyang Guo, and Xianglong Liu. Outlier suppression+: Accurate quantization of large language models by equivalent and optimal shifting and scaling. *arXiv preprint arXiv:2304.09145*, 2023.

Dennis Wu, Jerry Yao-Chieh Hu, Teng-Yun Hsiao, and Han Liu. Uniform memory retrieval with larger capacity for modern hopfield models. In *Forty-first International Conference on Machine Learning (ICML)*, 2024a.

Dennis Wu, Jerry Yao-Chieh Hu, Weijian Li, Bo-Yu Chen, and Han Liu. STanhop: Sparse tandem hopfield model for memory-enhanced time series prediction. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024b.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance, 2023.

Xinhua Wu, Haoyu He, Yanchao Wang, and Qi Wang. Pretrained mobility transformer: A foundation model for human mobility. *arXiv preprint arXiv:2406.02578*, 2024c.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099. PMLR, 2023a.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023b.

Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. Parameter-efficient fine-tuning for pre-trained vision models: A survey. *arXiv preprint arXiv:2402.02242*, 2024.

Chenwei Xu, Yu-Chao Huang, Jerry Yao-Chieh Hu, Weijian Li, Ammar Gilani, Hsi-Sheng Goan, and Han Liu. Bishop: Bi-directional cellular learning for tabular data with generalized sparse modern hopfield model. In *Forty-first International Conference on Machine Learning (ICML)*, 2024a.

Zhuoyan Xu, Zhenmei Shi, and Yingyu Liang. Do large language models have compositional ability? an investigation into limitations and scalability. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024b.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*, 2023.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Ziqiang Zhang, Sanyuan Chen, Long Zhou, Yu Wu, Shuo Ren, Shujie Liu, Zhuoyuan Yao, Xun Gong, Lirong Dai, Jinyu Li, et al. Speechlm: Enhanced speech pre-training with unpaired textual data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

Tong Zhou, Jiahui Zhao, Yukui Luo, Xi Xie, Wujie Wen, Caiwen Ding, and Xiaolin Xu. Adapi: Facilitating dnn model adaptivity for efficient private inference in edge computing. *arXiv preprint arXiv:2407.05633*, 2024a.

Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.

Zhihan Zhou, Weimin Wu, Harrison Ho, Jiayi Wang, Lizhen Shi, Ramana V Davuluri, Zhong Wang, and Han Liu. Dnabert-s: Learning species-aware dna embedding with genome foundation models. *arXiv preprint arXiv:2402.08777*, 2024b.

Appendix

A. Related Work

Discrete Speech Representation. Recent advances in Self-Supervised Learning (SSL) for speech, exemplified by models such as HuBERT (Hsu et al., 2021) and w2v-BERT (Chung et al., 2021), enhance our ability to extract meaningful representations from raw audio data. These models generate semantic tokens by clustering learned features, effectively capturing the linguistic content of speech. This process transforms speech into pseudo-text, which is highly beneficial for speech-based natural language understanding and generation applications. Specifically, models like HuBERT transform continuous speech features into discrete tokens representing phonetic or sub-word units, thereby improving the accuracy and efficiency of high-level speech processing tasks such as text-to-speech (TTS) (Hayashi and Watanabe, 2020), speech-to-speech translation (S2ST) (Lee et al., 2021), and automatic speech recognition (ASR) (Park et al., 2019).

Speech and Text LMs. As foundation models advance, multi-modality (Lu et al., 2024; Liu et al., 2024; Girdhar et al., 2023; Luo et al., 2023a; Wang et al., 2023a; Samel et al., 2021) gains prominence, especially in the integration of speech and text. This area is now a major focus within the field of foundation models. Jointing modeling of speech and text becomes a central area of research. Early methods (e.g., (Ao et al., 2021; Chen et al., 2022)) utilize separate encoders and decoders for speech and text, incorporating alignment losses to facilitate cross-modal transfer. However, more recent approaches shift towards unified models capable of handling multiple tasks simultaneously. For instance, SpeechGPT (Zhang et al., 2023) merges audio generation with text language models, while PolyVoice (qian Dong et al., 2023) applies speech language modeling to speech-to-speech translation. Similarly, SpiritLM (Nguyen et al., 2024) is noted for its capabilities in both speech and expressive speech generation, and is further adaptable to related speech tasks. Voxtlm (Maiti et al., 2024) is versatile, supporting speech/text generation as well as automatic speech recognition and text-to-speech. In our work, we start with a textually pretrained OPT model (Zhang et al., 2022) for better initialization, inspired by (Maiti et al., 2024; Hassid et al., 2024), and utilize various speech tokens to ensure the full reproducibility of our findings.

Low-Rank Adaptation and Post Training Quantization. Low-Rank Adaptation (Xin et al., 2024; Huang et al., 2024; Dettmers et al., 2024; Li et al., 2023; Hu et al., 2021) and Post Training Quantization (PTQ) (Qin et al., 2024a; Xu et al., 2024b; Gu et al., 2024a; Luo et al., 2023b; Gholami et al., 2022; Horowitz, 2014; Tang and Kwan, 1993; Marchesi et al., 1993) are essential for reducing the memory footprint and latency of large foundation models (Bommasani et al., 2021), particularly those based on extensive transformer architectures. These models are crucial not only in machine learning but also across various scientific fields such as finance (Wang et al., 2023b; Wu et al., 2023), genomics (Zhou et al., 2024b; 2023; Ji et al., 2021), human mobility (Wu et al., 2024c), and speech processing (Maiti et al., 2024). Despite their effectiveness, these resource-intensive models necessitate techniques like Low-Rank Adaptation and PTQ for deployment on resource-constrained edge devices. Significant contributions are made in the area of Low-Rank Adaptation (Meo et al., 2024; Qin et al., 2024b; Dettmers et al., 2024; Zhou et al., 2024a; Li et al., 2023; Hu et al., 2021) and PTQ (Ma et al., 2024; Shao et al., 2023; Xiao et al., 2023a; Luo et al., 2023b). However, they typically do not address the outlier problem during the adaptation and quantization processes, as highlighted by (Hu et al., 2024a). Moreover, Hu et al. (2024c) suggest that LoRA adaptor weights might lead to performance and efficiency degradation due to their additive nature. To tackle these, we incorporate the Outlier Efficiency Layer (`OutEffHop`), specifically designed to manage outliers during both the Low-Rank Adaptation and the quantization phases, thereby enhancing model robustness and reliability.

Outlier-Efficient Methods. The Outlier-Efficient Method aims to mitigate the effects of outliers during the pre-training, fine-tuning, and deployment phases of model development (Hu et al., 2024a; Chen et al., 2024; Bondarenko et al., 2024; Xiao et al., 2023a). In the deployment phase, quantization reduces the computational demands of large models through low-bit precision computing. However, outliers, creating disproportionately large attention weights, often compromise the efficacy of quantizing transformer-based models (Bondarenko et al., 2023; 2021). To counter this, Wei et al. (2022) modify LayerNorm to enable outlier-free quantization of activation tensors and introduce Token-Wise Clipping to optimize token-specific clipping ranges. Additionally, Dettmers et al. (2022b) use varying precision levels for quantizing outlier features, while Meo et al. (2024) incorporate a Bayesian approach with a prior distribution on quantization levels to help manage outliers. Despite these efforts, as outliers stem from the $\mathrm{Softmax}$ function, these methods do not address the root cause. Prior research (Hu et al., 2024a; Bondarenko et al., 2023) identifies no-op tokens as a primary issue: tokens that have

small value vectors often receive disproportionately large attention weights. Hu et al. (2024a) interpret the outlier effect in modern Hopfield networks as inefficient rare memory retrieval. They propose replacing the standard transformer layer with an outlier-efficient Hopfield layer to address this issue.

Theories of Outliers in Transformer Attention Heads. Recent theoretical results also highlight the benefits of outlier removal from attention heads in large transformer-based foundation models. Alman and Song (2023a;b) show that efficient transformers, including vanilla and tensor versions, require bounded attention weights through precise reduction methods. Hu et al. (2024b) indicate that efficient dense associative memory models (i.e., modern Hopfield models) and their corresponding networks also require bounded query and key patterns for sub-quadratic time complexity using fine-grained reduction techniques. Additionally, Hu et al. (2024c) theoretically show that the existence of outliers hamper the efficiency and performance of LoRA fine-tuning. Further, Hu et al. (2024c); Gu et al. (2024b;c); Alman and Song (2024); Gao et al. (2023) show that bounded weight matrices are essential for the efficient training of transformer-based models.

B. Outliers Challenge in Transformer Architecture

Clark et al. (2019) and Kovaleva et al. (2019) show that in BERT models, certain tokens like delimiters and punctuation marks are allocated disproportionately high attention weights. Similarly, Kobayashi et al. (2020) note that tokens with small value vectors also receive significantly larger attention weights. According to Bondarenko et al. (2024); Hu et al. (2024a), these tokens, despite their low informational value, command high attention probabilities, resulting in a no-update operation. This phenomenon not only increases computational and memory requirements during training but also leads to marked performance drops during model quantization.

To see this, let $X = [x_1, \ldots, x_m] \in \mathbb{R}^{d \times m}$ denote the input and recall the attention mechanism

$$\mathrm{Attention}(X) = \mathrm{Softmax}(QK^\top/\sqrt{d})V = A. \tag{B.1}$$

We focus on the part of transformer right after the attention mechanism, model residual

$$\mathrm{Output} = \mathrm{Residual}(A + X). \tag{B.2}$$

If the input X already has enough information and does not need to be updated, the transformer should not update the input X. As a result, the attention mechanism does not pay attention to the input X and the attention weight should be 0. This is known as the no-update situation: the output of Equation (B.2) should same as input X. However, the attention mechanism does not always work as expected. The attention mechanism focus tokens with large values (as in V) receive near-zero attention weights (as in $\mathrm{Softmax}(QK^\top)$), while tokens with small values receive large attention probability. By normalize the natural of the $\mathrm{Softmax}$ function, the operation focus on its input $QK^\top$ to have the wide range. This is fundamental source of the outliers: there must be some tokens causing the wide-range of the $QK^\top$, namely the outliers. Those findings also suggests in several recently research works (Sun et al., 2024; Hu et al., 2024a; Xiao et al., 2023b).

C. Computational Resource

All experiments are conducted using four NVIDIA A100 GPUs, each with 80GB of memory, and a 24-core Intel(R) Xeon(R) Gold 6338 CPU operating at 2.00GHz. Our experimental code is developed in PyTorch and utilize the Hugging Face Transformers Library for execution.

D. Visualization of Outliers Challenge

As shown in Figure 2, we use visualization to highlight the challenges posed by outliers in transformer-based models during the fine-tuning period. We visualize the model's hidden representations in the last hidden layers during the LoRA fine-tuning process. In the figure, deeper shades of red indicate higher values of attention probability, value, and weight. Conversely, deeper shades of blue represent lower values. This color coding helps illustrate the concentration of attention and computational focus within the model. In the vanilla model, we observe that the attention probability is distributed across various tokens rather than being concentrated on specific, significant tokens. This dispersion can cause the model to expend effort on unnecessary tokens during fine-tuning, leading to performance degradation and resources inefficiency. In contrast, the `OutEffHop`-enhanced model shows a more focused attention distribution, which helps reduce the computational effort required for fine-tuning by concentrating on the significant tokens. Additionally, we find that the attention weight in

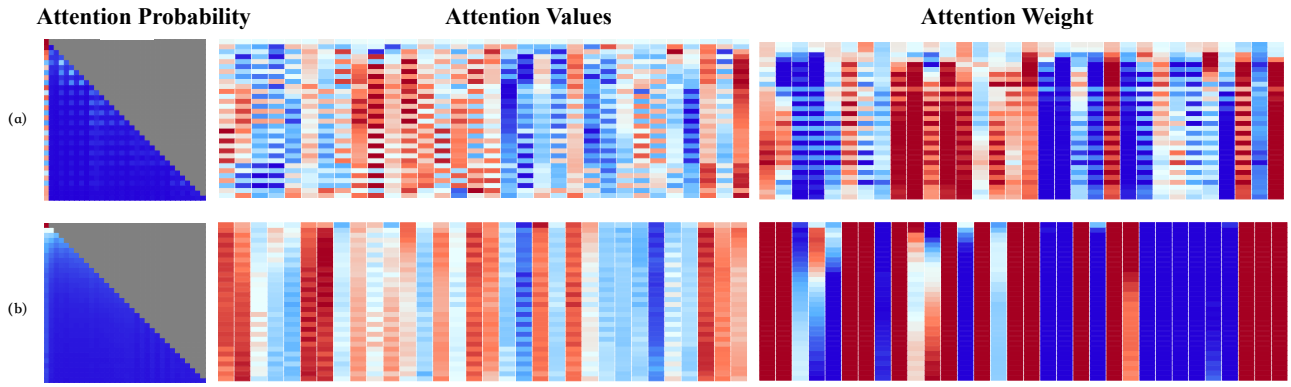

Figure 2. **Visualization of Attention Probability, Value and Weight in LoRA Funetuning.** We present a visualization of the attention probability, value, and weight for a cross-modality speech sample processed by the OPT-125m model. The visualization includes two scenarios: (a) the vanilla OPT-125m model, and (b) the `OutEffHop`-enhanced OPT-125m model (Hu et al., 2024a). Additionally, we visualize the model's hidden representations in the last hidden layers during the LoRA fine-tuning process and scale up all heatmaps from range 0 (blue) to 1 (red). In the vanilla model, we observe that the attention probability is distributed across various tokens rather than being concentrated on specific, significant tokens. This dispersion causes the model to expend effort on unnecessary tokens during fine-tuning, leading to performance degradation and resources inefficiency. In contrast, the `OutEffHop`-enhanced model shows a more focused attention distribution, which helps in reducing the computational effort required for fine-tuning by concentrating on the significant tokens. See Figure 1 for numerical verification.

`OutEffHop` is higher for tokens with high attention values. This indicates that the model does not spend extra computational resources on less significant tokens, allowing the fine-tuning process to converge more efficiently.

E. Additional Numerical Experiments

Influence of Adaptor Rank. We conducte a comprehensive analysis to assess the efficacy of our proposed method using Low-rank Adaptation (LoRA) across different ranks, comparing it to the conventional (vanilla) approach. The results, detailed in Table 3, show that our method consistently outperformed the vanilla approach at all tested ranks, with a rank of 256 delivering optimal performance. This lead us to standardize on a rank of 256 for all subsequent LoRA experiments. Further investigation indicates that increasing the rank beyond 256 does not lead to further performance improvements. This is attributed to the fact that a higher rank introduces additional trainable parameters, which can improve learning capacity but also demand significantly more training epochs for effective convergence. This extended training process introduces inefficiencies and practical limitations that may outweigh the benefits of a higher parameter count, making a rank of 256 the most effective choice for balancing performance enhancement and computational efficiency.