RoSA: Enhancing Parameter-Efficient Fine-Tuning via RoPE-aware Selective Adaptation in Large Language Models

Dayan Pan[1,3], **Jingyuan Wang**[1,2,3*], **Yilong Zhou**[1,3], **Jiawei Cheng**[1,3,4], **Pengyue Jia**[4], **Xiangyu Zhao**[4*]

[1]School of Computer Science and Engineering, Beihang University, Beijing, China
[2]School of Economics and Management, Beihang University, Beijing, China
[3]MOE Engineering Research Center of Advanced Computer Application Technology, Beihang University, China
[4]Department of Data Science, City University of Hong Kong, Hong Kong, China
{dayan, jywang, yilongzhou, JarvisC}@buaa.edu.cn, jia.pengyue@my.cityu.edu.hk, xianzhao@cityu.edu.hk

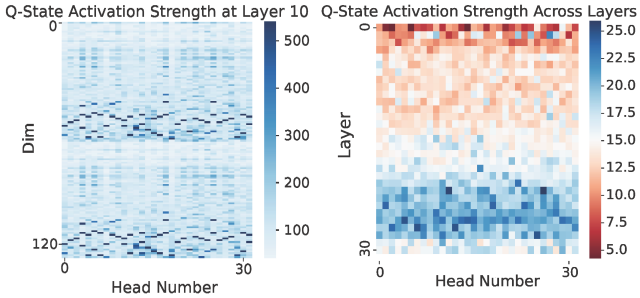
Abstract

Fine-tuning large language models is essential for task-specific adaptation, yet it remains computationally prohibitive. Parameter-Efficient Fine-Tuning (PEFT) methods have emerged as a solution, but current approaches typically ignore the distinct roles of model components and the heterogeneous importance across layers, thereby limiting adaptation efficiency. Motivated by the observation that Rotary Position Embeddings (RoPE) induce critical activations in the low-frequency dimensions of attention states, we propose RoPE-aware Selective Adaptation (RoSA), a novel PEFT framework that allocates trainable parameters in a more targeted and effective manner. RoSA comprises a RoPE-aware Attention Enhancement (RoAE) module, which selectively enhances the low-frequency components of RoPE-influenced attention states, and a Dynamic Layer Selection (DLS) strategy that adaptively identifies and updates the most critical layers based on LayerNorm gradient norms. By combining dimension-wise enhancement with layer-wise adaptation, RoSA achieves more targeted and efficient fine-tuning. Extensive experiments on fifteen commonsense and arithmetic benchmarks demonstrate that RoSA outperforms mainstream PEFT methods under comparable trainable parameters.

Introduction

Large Language Models (LLMs) have achieved remarkable success across a wide range of natural language processing (NLP) tasks, becoming a foundational infrastructure in numerous real-world applications (Cheng et al. 2025; Yu et al. 2025). However, deploying these large-scale models often requires fine-tuning to align models with specific task requirements. Traditional fine-tuning methods, such as full-parameter fine-tuning, are extremely resource-intensive, severely constraining their broader applicability. Consequently, exploring Parameter-Efficient Fine-Tuning (PEFT) methods, which aim to substantially reduce fine-tuning costs without compromising model performance, has emerged as a key research focus in the LLM community (Ding et al. 2023; Li et al. 2025; Han et al. 2025b; Liu et al. 2024c,d).

*Corresponding authors.



(a) Across Head Dimensions (b) Across Layers

Figure 1: Q-state activation strength visualizations in LLaMA-2-7B. We compute the average L2 norm per attention head to quantify activation strength. Stronger activations are concentrated in high-indexed (*i.e.,* low-RoPE frequency) dimensions and vary across layers, highlighting both dimension-wise and layer-wise heterogeneity.

Recent PEFT methods typically aim to adapt LLMs to specific downstream tasks by fine-tuning only a small fraction of parameters, significantly reducing computational cost (Wang et al. 2025b; Liu et al. 2024a). For example, mainstream PEFT methods such as P-tuning (Liu et al. 2021), LoRA (Hu et al. 2021), DoRA (Liu et al. 2024b), and HyCAM (Pan et al. 2025) introduce lightweight and trainable adaptation modules into the pre-trained model, keeping most of the original model parameters frozen.

Despite advancements, existing PEFT methods exhibit two critical limitations: **(1) Component-Heterogeneity Neglect:** Current methods largely neglect the intrinsic functional roles of LLM components (Zhang et al. 2023). For instance, LoRA inserts low-rank matrices into the linear layers of attention and feed-forward blocks, enabling adaptation with minimal trainable parameters. However, such designs are applied uniformly across modules without analyzing their distinct functional roles. **(2) Layer-Heterogeneity Neglect:** Existing approaches often overlook the diversity across layers. However, LLMs capture syntax in lower layers, semantics in higher layers (Voita, Sennrich, and Titov 2019). Most PEFT methods apply uniform adaptation schemes across all layers, limiting the potential efficiency and effectiveness of parameter allocation.

Our approach is motivated by a key observation regarding LLM architectures: different components exhibit distinct roles and activation behaviors (Xue et al. 2025a,b; Wang et al. 2025a). Recent studies suggest that Feed-Forward Networks (FFN) act as repositories for storing factual knowledge, while Multi-Head Attention (MHA) modules function primarily for knowledge retrieval and contextual routing (Geva et al. 2021). A key component within the MHA module is the Rotary Position Embedding (RoPE) (Su et al. 2024), which plays a critical role in contextual understanding by encoding positional information into attention mechanisms. RoPE achieves this by applying pair-wise complex rotations to the Query (Q) and Key (K) state tensors of attention mechanism and the sinusoidal frequency increases geometrically across successive dimension pairs.

This frequency-based encoding introduces unique activation patterns. As shown in Fig.1(a), there are obvious distinctions in Q-state activations across different dimensional channels. Specifically, low-frequency components (corresponding to higher-indexed dimensions within each half of the attention states) exhibit denser and more intense activations, while high-frequency shows sparser activations. Analyses confirm that these prominent low-frequency activations are crucial for contextual understanding (Barbero et al. 2024; Jin et al. 2025). Furthermore, Fig.1(b) reveals that this activation intensity is also highly heterogeneous across different layers, suggesting their contributions are not equal. These findings highlight that targeting these critical low-frequency components and the varying importance across layers for fine-tuning hold significant potential for enhancing both model performance and parameter efficiency.

Building on this, we propose a novel parameter-efficient fine-tuning method called RoPE-aware Selective Adaptation (RoSA). Specifically, RoSA integrates two complementary modules: (1) *a RoPE-aware Attention Enhancement (RoAE)* module, explicitly designed to adaptively enhance the distinctive low-frequency components within query/key states influenced by the RoPE mechanism, thereby enhancing the model's contextual understanding capabilities with high parameter efficiency. (2) a *Dynamic Layer Selection (DLS)* strategy, enabling RoSA to dynamically identify and adapt only the most critical layers during fine-tuning. Specifically, layer importance is quantified by computing the gradient norm of Layer Normalization parameters, serving as a reliable proxy for determining each layer's contribution to model performance. By simultaneously leveraging RoPE's inherent structural characteristics and dynamically allocating fine-tuning resources to layers that matter most, RoSA substantially improves parameter efficiency and model effectiveness compared to existing PEFT techniques. The main contributions of this paper are summarized as follows:

- To our knowledge, among PEFT works, we are the first to explicitly consider the distinctive low-frequency attention components induced by RoPE and propose RoAE, a RoPE-aware PEFT module that performs targeted enhancement of these functionally key dimensions. This adaptation effectively strengthens contextual understanding capabilities in a highly parameter-efficient manner.
- We introduce RoSA, a comprehensive PEFT framework that combines the RoAE module with a Dynamic Layer Selection (DLS) strategy. Specifically, DLS adaptively identifies and selectively updates the most impactful layers based on gradient norms of Layer Normalization parameters. Thus, RoSA optimally allocates parameters both dimension-wise and layer-wise according to their functional importance, enhancing overall efficiency.
- Extensive experiments on fifteen public benchmark datasets, using three backbone models and covering commonsense and arithmetic QA tasks, demonstrate that RoSA significantly outperforms existing mainstream PEFT methods under comparable trainable parameter scales, validating both its efficiency and effectiveness.

Preliminaries

This section reviews the key components of LLMs and the RoPE mechanism, which form the basis of our method.

LLM Architecture

Modern LLMs, such as the LLaMA, are primarily built upon the decoder-only Transformer architecture (Vaswani et al. 2017), which has been widely adopted across diverse representation learning settings (Yang et al. 2025; Jiang et al. 2023b,a; Han et al. 2025a). This architecture consists of a stack of identical Transformer blocks, each containing two primary components: a Multi-Head Self-Attention (MHSA) module and a Feed-Forward Network (FFN) module. The MHSA module allows the model to weigh the importance of different tokens in the input sequence, capturing complex contextual relationships. To incorporate crucial information about token order, which self-attention itself lacks, these models integrate positional encodings. Specifically, modern LLMs heavily adopt the Rotary Position Embedding (RoPE) (Su et al. 2024) as a relative positional encoding mechanism, which directly injects relative positional information into the attention computation and plays a crucial role in the model's ability to generalize over long contexts. The FFN, typically composed of two linear layers with a non-linear activation function, is responsible for feature transformation and is believed to be a key repository of factual and commonsense knowledge stored within the model's parameters (Geva et al. 2021). A residual connection (He et al. 2016) is applied around each of the two sub-modules, followed by a Layer Normalization step. Most LLMs utilize Pre-LN for enhanced training stability, where normalization is applied directly to the input of each sub-module. In this design, LayerNorm acts as a bridge between residual stream and subsequent attention or FFN modules, modulating the information flow across modules and layers.

Rotary Position Embedding (RoPE)

As mentioned in the previous section, the original self-attention mechanism is inherently permutation-invariant, meaning that the order of input tokens does not affect the output. Therefore, an external mechanism is required to encode token positions. While early models use additive, learned absolute position embeddings, modern LLMs widely adopt Rotary Position Embedding (RoPE) (Su et al.

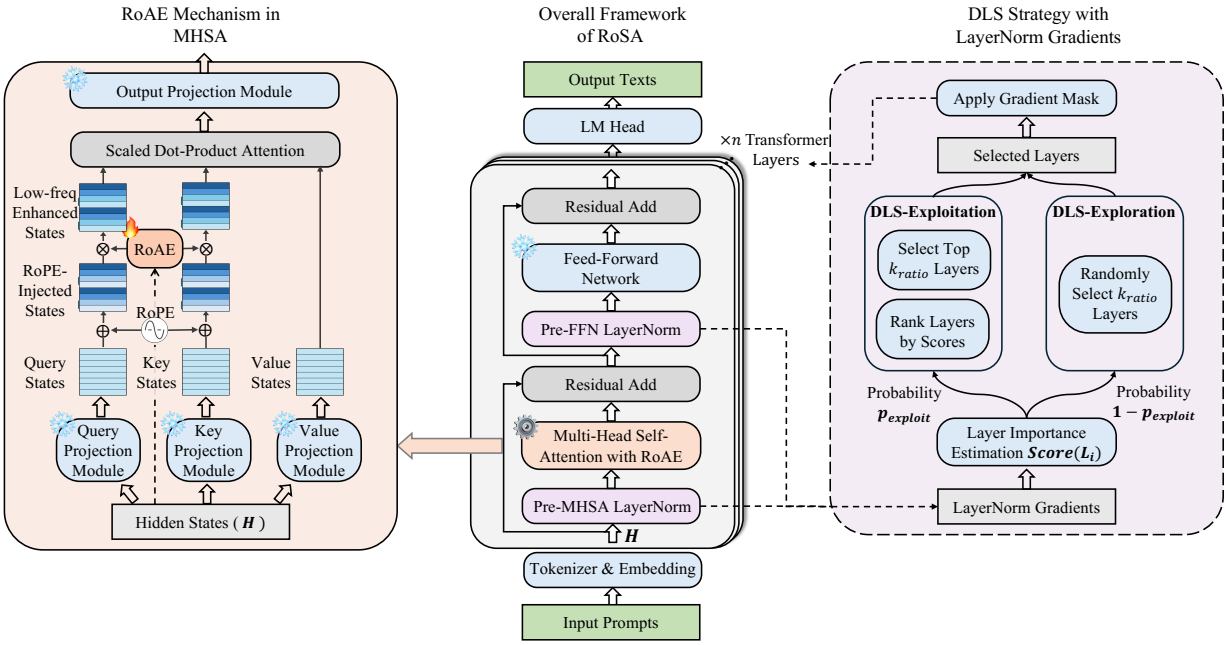


Figure 2: The architecture of RoSA. RoSA consists of two key modules: RoPE-aware Attention Enhancement (RoAE), which selectively enhances low-frequency components of RoPE-influenced Q/K states, and Dynamic Layer Selection (DLS), which dynamically selects important layers for update. Enabling targeted, efficient adaptation both frequency-wise and layer-wise.

2024) due to its effectiveness and efficiency in encoding relative positional information, especially for long sequences.

RoPE injects positional information by applying a rotational transformation directly to the Query (q) and Key (k) vectors in each attention head. Specifically, given a vector $\mathbf{z} \in \mathbb{R}^d$, where d is even, RoPE splits it into two halves: a *real* part $\mathbf{z}^{\text{real}}$ and an *imaginary* part $\mathbf{z}^{\text{imag}}$, each of dimension $d/2$. Then, for each index i, RoPE treats $(\mathbf{z}_i^{\text{real}}, \mathbf{z}_i^{\text{imag}})$ as a complex-valued component and applies a 2D rotation:

$$\text{RoPE}(\mathbf{z}_i^{\text{real}}, \mathbf{z}_i^{\text{imag}}) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix} \begin{bmatrix} \mathbf{z}_i^{\text{real}} \\ \mathbf{z}_i^{\text{imag}} \end{bmatrix}, \quad (1)$$

where $\theta_i = t \cdot \omega^{-2i/d}$, t is the token position index, and ω is a base frequency constant (commonly set to 10,000). This operation is equivalent to applying a complex-valued sinusoidal rotation, enabling relative positional relationships to be encoded directly into the attention mechanism. Since each rotation is applied to the corresponding dimensions in the two halves of the vector, both halves share the same rotation frequency θ_i. As observed in Fig.1(a), the activation patterns exhibit similarity, highlighting the impact of RoPE on the attention mechanism across dimensions.

As θ_i decreases geometrically with the index i, low-indexed dimensions encode high-frequency positional patterns, while the high-indexed dimensions encode low-frequency, smoother components. These low-frequency components often produce stronger and denser activations, and are crucial for long-range dependency modeling. These observations suggest that the frequency structure induced by RoPE provides a meaningful basis for improving PEFT methods. In this work, we explicitly target the low-frequency components of RoPE-influenced attention states, aiming to enhance parameter efficiency in a more targeted manner.

Method

In this section, we first provide an overview of the RoSA framework, then describe its two core components in detail, and finally present the overall algorithm.

Framework Overview

Existing PEFT methods often overlook two key aspects of LLMs: (*i*) the frequency-specific structure introduced by RoPE, and (*ii*) the layer-wise importance heterogeneity during adaptation. This motivates us to design a more targeted and adaptive fine-tuning strategy. To address these challenges, we propose RoPE-aware Selective Adaptation (RoSA). The core idea is to achieve a more targeted and efficient fine-tuning through a dual-level adaptation strategy, targeting critical low-frequency dimensions within layers and selecting the most important layers across the model.

As illustrated in Fig.2, RoSA achieves this through two main components. First, the RoPE-aware Attention Enhancement (RoAE) module selectively enhancing the low-frequency components of RoPE-influenced attention states, which play a critical role in contextual understanding. Further, the Dynamic Layer Selection (DLS) module identifies and adapts the most important layers during fine-tuning based on a gradient importance metric. By combining frequency-wise and layer-wise selective adaptation, RoSA achieves a more effective and efficient adaptation process.

RoPE-aware Attention Enhancement (RoAE)

Based on the observation that the low-frequency dimensions of RoPE-rotated attention states play a critical role in model-

ing long-range dependencies and contextual semantics (Barbero et al. 2024; Jin et al. 2025). However, conventional PEFT methods do not explicitly consider this frequency structure, instead applying generic adaptations across all dimensions. This limits their efficiency and effectiveness. To address this, we introduce the RoPE-aware Attention Enhancement (RoAE) module, which selectively enhances the low-frequency components within the Query (Q) and Key (K) attention states in a lightweight and targeted manner.

Low-Frequency Components Selection: Given the hidden states $\mathbf{H} \in \mathbb{R}^{b\times l\times d}$ as input to Transformer, where b is the batch size, l is the sequence length, and d is the hidden dimension. After applying the linear projections to obtain the query and key tensors, these are reshaped into multi-head with shape $[b, h, l, d_h]$, where h is the number of attention heads and $d_h = d/h$ is the dimension per head. RoPE first splits each head vector into real $\mathbf{z}_{\text{real}}$ and imaginary $\mathbf{z}_{\text{imag}}$ halves, then applies a sinusoidal rotation to every resulting complex pair.

To extract the low-frequency components, we follow the structure of RoPE and split each head vector into two halves of size $d_h/2$. From each half, we take the last $(d_h \cdot r_{\text{low}})/2$ dimensions and concatenate them to form a d_{low}-dimensional vector, denoted as $\mathbf{z}_{\text{low}}$. Here, $r_{\text{low}} \in (0, 1)$ is a hyperparameter controlling the ratio of the targeted low-frequency components. This extracted vector captures the critical low-frequency components of the RoPE-influenced Q/K head, serving as the target for enhancement.

Adaptation Signal Generation: To enhance the extracted low-frequency components in a targeted way, we first generate a context-aware adaptation signal $\mathbf{S}$. Specifically, the hidden state is passed through a trainable linear projection, $\mathbf{W}_{\text{proj}}$, followed by a non-linear activation (SiLU) (Elfwing, Uchibe, and Doya 2018) to introduce non-linearity:

$$\tilde{\mathbf{S}} = \text{SiLU}(\mathbf{H}\mathbf{W}_{\text{proj}}), \quad \mathbf{W}_{\text{proj}} \in \mathbb{R}^{d\times(h\cdot d_{\text{low}})}, \tag{2}$$

where $\tilde{\mathbf{S}} \in \mathbb{R}^{b\times l\times(h\cdot d_{\text{low}})}$. Similarly, we then reshape the projected tensors to the multi-head shape $\mathbf{S} \in \mathbb{R}^{b\times h\times l\times d_{\text{low}}}$.

Notably, to improve parameter efficiency, the projection module $\mathbf{W}_{\text{proj}}$ is implemented using a low-rank decomposition ($\mathbf{W}_{\text{proj}} = \mathbf{BA}$), adding only a small number of trainable parameters. Further, this design remains compatible and can be flexibly replaced by other emerging PEFT methods.

In typical settings, we use the same adaptation signal $\mathbf{S}$ for both query and key projections. To ensure compatibility with modern architectures employing Grouped-Query Attention (GQA) (Ainslie et al. 2023), where the number of query and key heads, denoted by h_q and h_k, may differ, we apply an additional projection module to align the dimensions:

$$\tilde{\mathbf{S}}^{(K)} = \tilde{\mathbf{S}}^{(Q)} \cdot \mathbf{W}_{\text{GQA}}, \quad \mathbf{W}_{\text{GQA}} \in \mathbb{R}^{(h_q\cdot d_{\text{low}})\times(h_k\cdot d_{\text{low}})}, \tag{3}$$

ensuring compatibility across varying attention configs, thereby enabling RoAE to support GQA-enabled models.

Targeted Enhancement Application: After obtaining the adaptation signal $\mathbf{S}$, the final step is to apply it to the targeted low-frequency components. Recall that in the previous step, we extracted the low-frequency vectors $\mathbf{z}_{\text{low}}$ of each head. Denoting the extracted low-frequency components for all attention heads as $\mathbf{Z} \in \mathbb{R}^{b\times h\times l\times d_{\text{low}}}$, we perform the enhancement via an element-wise multiply modulation:

$$\mathbf{Z}^* = \mathbf{Z} + \mathbf{Z} \odot (\alpha \cdot \mathbf{S}), \tag{4}$$

here α is a scaling factor controlling the adaptation strength.

Finally, the enhanced low-frequency tensors $\mathbf{Z}^*$ are reintegrated into their original positions of the attention head states, replacing the corresponding low-frequency dimensions. The attention mechanism then proceeds with these selectively enhanced query and key representations, allowing the model to better leverage RoPE's critical frequency structure for improved contextual understanding abilities.

In summary, the RoAE module introduces a targeted and efficient PEFT paradigm. Its core innovation lies in its mechanism-aware design, which targets the critical components of RoPE-influenced attention states. Furthermore, the enhancement is context-aware, as the adaptation signal is dynamically generated from the input states to provide token-specific modulations. By achieving this with high parameter efficiency and maintaining compatibility across diverse architectures, RoAE establishes a more flexible and effective method for adapting LLMs into specific tasks.

Dynamic Layer Selection (DLS)

While the RoAE module provides a targeted, mechanism-aware approach to adapting parameters within one layer, LLMs exhibit considerable heterogeneity across different layers, with lower layers primarily capturing syntactic features and higher layers encoding abstract semantic and contextual knowledge (Voita, Sennrich, and Titov 2019). Applying it uniformly across all layers, like common PEFT methods, overlooks the layer-wise importance heterogeneity. To address this, we propose Dynamic Layer Selection (DLS) strategy, a method designed to dynamically select and adapt the most important layers, improving parameter utilization efficiency throughout the fine-tuning process.

Layer Importance Estimation: The core of DLS is to accurately estimate the importance of each layer with respect to the fine-tuning objective. We propose to use the gradient norm of Layer Normalization (LayerNorm) parameters as an efficient proxy for this task. Because LayerNorm directly controls information flow between Transformer submodules and layers. A large gradient for this parameter indicates that it is necessary for the model to significantly change the output distribution of this layer to minimize the loss.

In the common-adopted Pre-LN architecture, LayerNorm modules are placed before the self-attention and before the FFN module. Formally, for the i-th Transformer layer L_i, its importance score is calculated by aggregating the L_2 norms of the gradients from the LayerNorm parameters:

$$\text{Score}(L_i) = \sqrt{\|\nabla\mathbf{\Theta}_{i,\text{attn}}\|_2^2 + \|\nabla\mathbf{\Theta}_{i,\text{ffn}}\|_2^2} \tag{5}$$

where $\mathbf{\Theta}_{i,\text{attn}}$ and $\mathbf{\Theta}_{i,\text{ffn}}$ represent the learnable parameters for the two LayerNorm modules in the i-th layer. In practice, we periodically compute these importance scores for all layers, providing an informative metric to guide selection.

Dynamic Selection and Gradient Masking: The selection procedure is activated periodically at an interval of u steps after an initial warmup phase. At each activation, DLS employs a strategy that balances exploitation and exploration to choose a subset of layers for updates, specifically:

- **Exploitation:** With a high probability p_{exploit}, we rank all layers based on their scores and select the top-k layers for training, where k is determined by a predefined ratio k_{ratio}.
- **Exploration:** Conversely, with a probability of $1-p_{\text{exploit}}$, we randomly select k layers to ensure that all layers have a chance to adapt, thus reducing the risk of local optima.

Once the set of selected layers $\mathcal{L}_{\mathcal{S}}$ is determined, a gradient mask is applied. Specifically, the gradients of parameters in all non-selected layers are set to 0 to prevent updating:

$$\nabla L_i \leftarrow \mathbf{0}, \quad \text{if} \quad i \notin \mathcal{L}_{\mathcal{S}}. \tag{6}$$

In summary, DLS reduces unnecessary parameter updates by dynamically identifying and adapting only the most critical layers, leading to improved efficiency and potentially superior downstream task performance. It is noteworthy that DLS is model-agnostic and can be easily integrated into existing PEFT pipelines. Combined with RoAE, which enables selective adaptation over important frequency components, DLS completes the RoSA framework by jointly targeting both dimension-level and layer-level adaptation.

Overall Algorithm

RoSA integrates the RoAE and DLS modules into the standard causal language modeling framework, where the model is trained using cross-entropy loss between predicted and target tokens. These modules operate jointly, enabling targeted adaptation both across frequency dimensions and model layers, achieving effective and efficient fine-tuning.

The full training procedure is summarized in Algorithm 1, which outlines how RoSA applies frequency-aware enhancements via RoAE and dynamically selects critical layers for update via DLS. Thus, RoSA optimally allocates parameters both dimension-wise and layer-wise according to their functional importance, enhancing overall efficiency. Importantly, RoSA can be seamlessly integrated into existing PEFT frameworks or combined with other fine-tuning techniques due to its modular and adaptive design.

Algorithm 1: RoPE-aware Selective Adaptation (RoSA)

Input: Pretrained LLM model $\mathcal{M}$, dataset $\mathcal{D}$, RoAE hyperparameters (α, r_{low}), DLS hyperparameters (k_{ratio}, p_{exploit}, u), learning rate η, warmup steps T_{warmup}.

```
Initialize RoAE modules with α and r_low;
Set only RoSA-related parameters Θ_RoSA as trainable;
for each training step t do
    Sample a batch of data from D;
    Compute forward pass with RoAE enhanced attention states (Eq. 2-4);
    Compute loss and perform backward for gradients;
    if t > T_warmup and t mod u == 0 then
        Calculate layer importance Score(L_i) (Eq. 5);
        With probability p_exploit, select the top k_ratio fraction of layers (DLS-Exploitation); otherwise, randomly select k_ratio fraction of layers (DLS-Exploration);
    end if
    Mask gradients in non-selected layers (Eq. 6);
    Update parameters of active layers using optimizer with learning rate η;
end for
```

Experiments

To comprehensively evaluate the performance of our proposed RoSA, we conduct extensive experiments guided by the following key research questions (RQs):

- **RQ1:** How does RoSA perform compared to state-of-the-art PEFT methods across different backbone LLMs and downstream tasks?
- **RQ2:** How does RoSA demonstrate scalability performance with backbone LLMs of different parameter sizes?
- **RQ3:** What are the contributions of each component within RoSA (RoAE and DLS) to its overall performance?
- **RQ4:** How do RoSA's key hyperparameters affect its overall performance?

We first introduce the experimental setup and then systematically address each of the above research questions.

Experimental Setup

Datasets We follow LLM-Adapters (Hu et al. 2023) and evaluate RoSA on two distinct tasks: Commonsense QA and Arithmetic QA. Specifically, we fine-tune models using `Commonsense15K` and `Math10K`, which are constructed from multiple data sources. For the *Commonsense* task, we evaluate on eight diverse benchmarks : BoolQ, PIQA, SIQA, ARC-Challenge, ARC-Easy, OBQA, HellaSwag, and WinoGrande. Further, we assess performance of the *Arithmetic* task on seven benchmarks: MultiArith, GSM8K, AddSub, AQuA, SingleEq, SVAMP, and MAWPS. We report accuracy on each benchmark as the evaluation metric.

Backbone Models We select three powerful and widely-used LLMs as backbone models to validate the generalization of RoSA: Qwen2.5-7B (Bai et al. 2023), Llama-3.1-8B (Dubey et al. 2024), and Gemma2-9B (Team et al. 2024).

Baseline Methods We evaluate our approach against a comprehensive set of recent and diverse PEFT methods. Specifically, we compare several low-rank methods and their variants, including the basic **LoRA** (Hu et al. 2021), its weights decomposing successor **DoRA** (Liu et al. 2024b), dynamically rank-allocating **AdaLoRA** (Zhang et al. 2023), and shared low-rank matrices **VERA** (Kopiczko et al. 2023). Methods leveraging more complex structured matrices, such as the orthogonality-enforcing **BOFT** (Liu et al. 2023), the circular-convolution-based **C3A** (Chen et al. 2024), and the block-affine-transformation-based **BONE** (Kang and Yin 2024) are also introduced. Finally, a simple and effective method **LN Tuning** (Zhao et al. 2023) is included, which only tunes the model's Layer Normalization parameters.

Implementation Details All experiments are conducted on NVIDIA GeForce RTX 3090 with PyTorch and Trans-

Backbone LLM	Baseline	# Param (%)	BoolQ	PIQA	SIQA	ARC-C	ARC-E	OBQA	HellaSwag	WinoGrande	micro-avg(%)↑
Qwen 2.5 7B	LoRA	0.527	66.9	86.8	76.7	88.2	93.9	87.2	89.7	72.2	84.3
	DoRA	0.546	68.3	87.4	77.2	89.4	95.2	88.0	90.0	70.4	84.9
	AdaLoRA	0.396	69.7	87.4	77.9	88.9	**95.7**	89.4	**90.6**	72.6	85.6
	BOFT	0.023	68.5	86.0	76.1	87.5	94.6	82.4	86.1	65.3	82.4
	VERA	0.018	55.4	83.7	74.1	85.1	93.6	77.2	82.2	64.1	77.9
	C3A	0.665	69.5	87.0	77.5	88.9	95.2	86.6	89.9	71.6	85.0
	BONE	0.291	67.6	84.9	76.8	85.2	94.3	87.4	88.3	**77.9**	83.9
	LN Tuning	0.001	62.5	86.0	73.3	85.0	93.3	77.2	80.9	62.1	78.4
	RoSA (ours)	0.261	**70.5**	**88.0**	**79.1**	**90.1**	95.3	**89.6**	**90.6**	73.7	**85.9***
Llama 3.1 8B	LoRA	0.520	**71.7**	86.8	75.5	83.1	92.7	82.4	88.6	68.8	83.7
	DoRA	0.537	71.5	86.9	75.8	83.2	92.5	82.2	88.5	70.0	83.8
	AdaLoRA	0.390	71.1	86.2	74.7	**83.6**	92.6	82.8	87.2	70.8	83.0
	BOFT	0.028	70.5	85.5	72.4	80.0	91.9	79.0	82.4	62.5	79.7
	VERA	0.017	68.8	82.9	68.4	77.6	91.4	77.4	75.2	57.4	75.2
	C3A	0.674	71.6	**87.7**	76.2	83.1	92.6	**84.4**	88.3	70.6	83.9
	BONE	0.274	64.7	78.4	74.2	72.1	86.8	78.2	81.8	70.3	77.6
	LN Tuning	0.003	70.1	84.6	70.9	80.2	91.8	78.8	80.6	61.8	78.6
	RoSA (ours)	0.329	**71.7**	87.1	**76.4**	83.3	**92.8**	83.6	**89.0**	**74.8**	**84.4***
Gemma 2 9B	LoRA	0.581	69.3	88.0	77.8	**88.0**	**95.5**	87.4	89.8	77.4	85.4
	DoRA	0.601	70.0	87.3	78.1	86.1	94.3	87.0	89.4	76.8	85.0
	AdaLoRA	0.437	72.3	88.2	77.4	87.5	**95.5**	86.2	89.0	73.4	85.1
	BOFT	0.029	65.2	83.2	72.4	81.7	91.1	75.0	80.3	62.1	77.7
	VERA	0.020	65.2	79.8	66.0	73.8	85.8	61.8	70.5	56.1	70.9
	C3A	0.699	70.7	87.7	77.7	86.9	94.5	86.8	**90.4**	75.3	85.5
	BONE	0.319	60.3	75.3	66.3	69.0	83.7	74.0	67.3	64.3	68.7
	LN Tuning	0.007	61.2	78.1	66.1	73.2	85.0	65.0	71.9	55.1	70.7
	RoSA (ours)	0.363	**74.0**	**88.3**	**78.5**	87.8	**95.5**	**87.8**	90.0	**77.5**	**86.2***

Table 1: Performance comparison of RoSA and baseline methods on the Commonsense QA task across three backbone LLMs. * indicates the statistically significant improvements (*i.e.,* two-sided t-test with $p < 0.05$) over the best baseline. RoSA consistently achieves the highest average performance under comparable parameter budgets.

Baseline	Qwen2.5 0.5B	Qwen2.5 1.5B	Qwen2.5 3B	Qwen2.5 7B
AdaLoRA	53.5	75.1	81.1	85.6
C3A	53.1	74.9	81.2	85.0
RoSA (ours)	**53.7**	**75.5**	**82.0**	**85.9**

Table 2: Average Commonsense QA accuracy of RoSA, AdaLoRA, and C3A on varying sizes Qwen2.5 (0.5 to 7B).

formers at [1]. We use AdamW optimizer with a learning rate of 1e-3. Hyperparameters are as follows: low-freq dimension ratio r_{low}: 0.25, scaling factor α: 0.1, low-rank projection dimension: 128, layer selection ratio k_{ratio}: 0.5, selection interval u: 40 steps and exploitation probability p_{exploit}: 0.8.

Overall Performance (RQ1, 2)

To answer RQ1, we compare RoSA against all baselines on two distinct tasks: Commonsense and Arithmetic QA. The results are summarized in Table 1 and Table 3, respectively.

As shown in Table 1, RoSA consistently achieves the best performance across all three backbone models, maintaining relatively low trainable parameters. This confirms that the low-frequency components introduced by RoPE play a crucial role in improving the model's contextual understanding. Among LoRA variants, AdaLoRA's dynamic rank allocation yields better performance, aligning with the principles of dynamic selection of DLS module. Methods like C3A, which employ novel adapter designs, also show competitive results, highlighting the potential of more complex structured matrices for improving parameter efficiency. Additionally, LN Tuning, a simple and effective method, performs well with minimal trainable parameters, further supporting the use of LayerNorm as an importance proxy in DLS.

[1] https://github.com/Applied-Machine-Learning-Lab/RoSA

To validate RoSA's capabilities, we also conduct a focused comparison on the Arithmetic QA task, specifically using the Qwen2.5-7B model due to space constraints. The results, summarized in Table 3, are consistent with those observed in the Commonsense task, where RoSA still achieves the best performance among all methods.

To further answer RQ2, we investigate how RoSA's performance scales with model size. We evaluate four Qwen2.5 variants (0.5B, 1.5B, 3B, and 7B) on the Commonsense QA task, comparing against two strong baselines, AdaLoRA and C3A. As shown in Table 2, all methods improve with larger models, but RoSA consistently maintains a clear advantage across scales, highlighting its robustness and scalability.

Ablation and Hyperparameter Analysis (RQ3, 4)

We then perform ablation and hyperparameter studies to analyze RoSA components and sensitivity to hyperparameters. All results in this section are reported as average performance on the Commonsense QA task with Qwen2.5-7B.

Ablation Study: We first conduct an ablation study comparing the full RoSA framework against several variants to evaluate the contributions of its components, as shown in Table. 4. The full **RoSA** model includes both RoAE and DLS. We first examine the **RoSA-RoAEonly** variant by disabling DLS for evaluating the impact of layer selection. We further investigate several RoAE replacement and modification variants, all retaining DLS: (i) **RoSA-RoAE0.5**, which sets the low-freq dimension ratio r_{low} to 0.5 while keeping all other settings unchanged, (ii) **RoSA-Lr128**, which applies standard LoRA on Q/K with all other configs identical to RoSA, and (iii) **RoSA-Lr64**, which uses LoRA with a similar number of trainable parameters as RoSA. These variants

Baseline	# Param (%)	MultiArith	GSM8K	AddSub	AQuA	SingleEq	SVAMP	MAWPS	micro-avg(%)↑
LoRA	0.527	93.0	68.7	88.8	33.8	88.9	79.2	88.2	77.7
DoRA	0.546	92.3	70.0	88.6	34.6	88.5	79.6	87.3	78.1
AdaLoRA	0.396	90.0	68.8	85.3	33.8	85.6	78.9	84.0	76.3
BOFT	0.023	89.6	67.8	82.5	31.1	86.2	75.2	80.2	74.6
VERA	0.018	72.5	63.7	80.7	31.1	80.3	74.2	83.1	70.0
C3A	0.665	**95.3**	67.1	90.3	**35.4**	**90.1**	82.1	89.4	78.7
BONE	0.291	92.8	66.6	89.6	33.4	88.3	82.1	89.0	77.8
LN Tuning	0.001	79.6	63.6	72.1	34.2	75.3	68.1	70.1	67.7
RoSA (ours)	0.261	94.3	**71.3**	**92.1**	35.0	**90.1**	**82.2**	**92.0**	**80.1***

Table 3: Evaluation of RoSA and baseline methods on the Arithmetic QA task using the Qwen2.5-7B model. RoSA achieves the highest average accuracy across all benchmarks, demonstrating its generalization to mathematical tasks.

(a) Ablation results		(b) Sensitivity results	
Variant	micro-avg↑	k_{ratio}	micro-avg↑
RoSA	**85.9**	0.10	80.3
RoAEonly (w/o DLS)	84.8	0.25	82.2
RoAE0.5 (w/ DLS/RoAE)	85.6	0.50	**85.9**
RoSA-Lr128 (w/o RoAE)	83.9	0.75	85.2
RoSA-Lr64 (w/o RoAE)	80.7	1.00	84.8

Table 4: Ablation and sensitivity analysis on layer selection ratio k_{ratio} on Commonsense tasks using Qwen2.5-7B.

also provide an implicit analysis of the effect of r_{low}, allowing us to compare targeted adaptation on varying frequency ranges. Overall, the results indicate that each component of RoSA contributes to performance, and focusing adaptation on a compact low-frequency subspace is more effective.

Sensitivity of DLS: To further evaluate the DLS module, we analyze the sensitivity of the layer selection ratio k_{ratio}, which controls the proportion of layers updated during fine-tuning. We vary k_{ratio} over a range of values. As summarized in Table. 4, RoSA performs best when $k_{\text{ratio}} \approx 0.5$. Increasing this ratio slightly degrades performance, suggesting that selectively updating fewer layers leads to more efficient optimization and enhances overall model performance.

Related Work

Parameter-Efficient Fine-Tuning

Parameter-Efficient Fine-Tuning (PEFT) aims to adapt LLMs to downstream tasks by tuning only a small subset of parameters, significantly reducing computational and memory costs. Adapter-based methods insert small trainable modules, enabling effective task adaptation with minimal parameters (Houlsby et al. 2019). Low-rank methods like LoRA (Hu et al. 2021) and its variants, including DoRA (Liu et al. 2024b), AdaLoRA (Zhang et al. 2023), and VERA (Kopiczko et al. 2023), inject trainable low-rank matrices into pretrained weights to achieve efficient adaptation. Advanced structured-matrix methods, such as C3A (Chen et al. 2024) and BONE (Kang and Yin 2024), introduce circular convolution or block affine into PEFT, further enhancing parameter efficiency through structured constraints. These efforts complement broader work on model efficiency, including compression and distillation techniques (Wang et al. 2023b), as well as domain-specific sequence modeling frameworks (Wang, Lin, and Li 2025) and efficient decision-making systems (Cong et al. 2021). However, most existing methods apply adaptation uniformly across model components, often neglecting their distinct functional roles.

Analysis of LLM Internals

Understanding the internal mechanics of LLMs is a growing research area that provides crucial insights for developing more principled and efficient methods. Early research shows that each FFN can be seen as a key-value memory (Geva et al. 2021). Recent work provides evidence that attention mechanisms are crucial for retrieving relevant context and enabling dynamic reasoning (Dong et al. 2025; Zhang et al. 2025), whereas the FFN layers are responsible for memorizing task-specific or factual content. RoPE in particular has been discussed in recent studies, inducing strong and dense activations in the low-frequency dimensions of attention states, and these activations are crucial for the LLMs' contextual understanding capabilities (Jin et al. 2025; Barbero et al. 2024). The frequency-structured behavior of attention has also been examined in wavelet-based or efficient attention training frameworks (Wang et al. 2023a; Fu et al. 2025). Meanwhile, analyses of layer-wise behavior reveal that not all layers are equally important (Belinkov et al. 2018), a trend also echoed in broader structure-aware neural modeling literature (Ji et al. 2022; Hettige et al. 2024; Wang et al. 2022). These findings underscore that different submodules contribute unique and complementary functions in LLMs, motivating our RoSA method.

Conclusion

In this work, we introduce RoPE-aware Selective Adaptation (RoSA), a novel PEFT framework for LLMs. RoSA explicitly leverages the frequency structure induced by RoPE by introducing a RoPE-aware Attention Enhancement (RoAE) module, which selectively enhances low-frequency attention components. Alongside, the Dynamic Layer Selection (DLS) strategy dynamically identifies and updates the most important layers based on LayerNorm gradients. This dual-level design enables more effective and targeted use of trainable parameters both within and across layers. Extensive experiments on fifteen commonsense and arithmetic QA datasets, covering multiple LLM families and model sizes, demonstrate that RoSA consistently outperforms baseline PEFT methods under comparable trainable parameters.

Acknowledgments

Jingyuan Wang's work was partially supported by the National Natural Science Foundation of China (No. 72171013, 72222022, 72242101), and the Fundamental Research Funds for the Central Universities (JKF-2025017226182). This research was partially supported by National Natural Science Foundation of China (No.62502404), Hong Kong Research Grants Council (Research Impact Fund No.R1015-23, Collaborative Research Fund No.C1043-24GF, General Research Fund No.11218325), Institute of Digital Medicine of City University of Hong Kong (No.9229503), Huawei (Huawei Innovation Research Program), Tencent (CCF-Tencent Open Fund, Tencent Rhino-Bird Focused Research Program), Alibaba (CCF-Alimama Tech Kangaroo Fund No. 2024002), Didi (CCF-Didi Gaia Scholars Research Fund), Kuaishou, and Bytedance.

References

Ainslie, J.; Lee-Thorp, J.; De Jong, M.; Zemlyanskiy, Y.; Lebrón, F.; and Sanghai, S. 2023. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Barbero, F.; Vitvitskyi, A.; Perivolaropoulos, C.; Pascanu, R.; and Veličković, P. 2024. Round and round we go! what makes rotary positional encodings useful? *arXiv preprint arXiv:2410.06205*.

Belinkov, Y.; Màrquez, L.; Sajjad, H.; Durrani, N.; Dalvi, F.; and Glass, J. 2018. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. *arXiv preprint arXiv:1801.07772*.

Chen, A.; Cheng, J.; Liu, Z.; Gao, Z.; Tsung, F.; Li, Y.; and Li, J. 2024. Parameter-efficient fine-tuning via circular convolution. *arXiv preprint arXiv:2407.19342*.

Cheng, J.; Wang, J.; Zhang, Y.; Ji, J.; Zhu, Y.; Zhang, Z.; and Zhao, X. 2025. Poi-enhancer: An llm-based semantic enhancement framework for poi representation learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, 11509–11517.

Cong, L. W.; Tang, K.; Wang, J.; and Zhang, Y. 2021. AlphaPortfolio: Direct construction through deep reinforcement learning and interpretable AI. *Available at SSRN 3554486*.

Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.-M.; Chen, W.; et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature machine intelligence*, 5(3): 220–235.

Dong, Y.; Noci, L.; Khodak, M.; and Li, M. 2025. Attention Retrieves, MLP Memorizes: Disentangling Trainable Components in the Transformer. *arXiv preprint arXiv:2506.01115*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv e-prints*, arXiv–2407.

Elfwing, S.; Uchibe, E.; and Doya, K. 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107.

Fu, Z.; Song, W.; Wang, Y.; Wu, X.; Zheng, Y.; Zhang, Y.; Xu, D.; Wei, X.; Xu, T.; and Zhao, X. 2025. Sliding Window Attention Training for Efficient Large Language Models. *arXiv preprint arXiv:2502.18845*.

Geva, M.; Schuster, R.; Berant, J.; and Levy, O. 2021. Transformer Feed-Forward Layers Are Key-Value Memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 5484–5495.

Han, C.; Wang, J.; Wang, Y.; Yu, X.; Lin, H.; Li, C.; and Wu, J. 2025a. Bridging traffic state and trajectory for dynamic road network and trajectory representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 11763–11771.

Han, X.; Zhao, Z.; Wang, W.; Wang, M.; Liu, Z.; Chang, Y.; and Zhao, X. 2025b. Data Efficient Adaptation in Large Language Models via Continuous Low-Rank Fine-Tuning. *arXiv preprint arXiv:2509.18942*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hettige, K. H.; Ji, J.; Xiang, S.; Long, C.; Cong, G.; and Wang, J. 2024. Airphynet: Harnessing physics-guided neural networks for air quality prediction. *arXiv preprint arXiv:2402.03784*.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hu, Z.; Wang, L.; Lan, Y.; Xu, W.; Lim, E.-P.; Bing, L.; Xu, X.; Poria, S.; and Lee, R. K.-W. 2023. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*.

Ji, J.; Wang, J.; Jiang, Z.; Jiang, J.; and Zhang, H. 2022. STDEN: Towards physics-guided neural networks for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, 4048–4056.

Jiang, J.; Han, C.; Zhao, W. X.; and Wang, J. 2023a. Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 4365–4373.

Jiang, J.; Pan, D.; Ren, H.; Jiang, X.; Li, C.; and Wang, J. 2023b. Self-supervised trajectory representation learning with temporal regularities and travel semantics. In *2023 IEEE 39th international conference on data engineering (ICDE)*, 843–855. IEEE.

Jin, M.; Mei, K.; Xu, W.; Sun, M.; Tang, R.; Du, M.; Liu, Z.; and Zhang, Y. 2025. Massive Values in Self-Attention Modules are the Key to Contextual Knowledge Understanding. *arXiv preprint arXiv:2502.01563*.

Kang, J.; and Yin, Q. 2024. Balancing LoRA Performance and Efficiency with Simple Shard Sharing. *arXiv preprint arXiv:2409.15371*.

Kopiczko et al. 2023. Vera: Vector-based random matrix adaptation. *arXiv preprint arXiv:2310.11454*.

Li, Z.; Hongjun, L.; Wang, J.; and Tang, K. 2025. Approximation to Smooth Functions by Low-Rank Swish Networks. In *Forty-second International Conference on Machine Learning*.

Liu, Q.; Wu, X.; Zhao, X.; Zhu, Y.; Xu, D.; Tian, F.; and Zheng, Y. 2024a. When moe meets llms: Parameter efficient fine-tuning for multi-task medical applications. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Liu, S.-Y.; Wang, C.-Y.; Yin, H.; Molchanov, P.; Wang, Y.-C. F.; Cheng, K.-T.; and Chen, M.-H. 2024b. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*.

Liu, W.; Qiu, Z.; Feng, Y.; Xiu, Y.; Xue, Y.; Yu, L.; Feng, H.; Liu, Z.; Heo, J.; Peng, S.; et al. 2023. Parameter-efficient orthogonal finetuning via butterfly factorization. *arXiv preprint arXiv:2311.06243*.

Liu, X.; Ji, K.; Fu, Y.; Tam, W. L.; Du, Z.; Yang, Z.; and Tang, J. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*.

Liu, Z.; Li, Z.; Wang, J.; and He, Y. 2024c. Full bayesian significance testing for neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 8841–8849.

Liu, Z.; Wang, J.; Li, Z.; and He, Y. 2024d. Full bayesian significance testing for neural networks in traffic forecasting. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*.

Pan, D.; Fu, Z.; Wang, J.; Han, X.; Zhu, Y.; and Zhao, X. 2025. Contextual Attention Modulation: Towards Efficient Multi-Task Adaptation in Large Language Models. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, 2273–2283.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Team, G.; Riviere, M.; Pathak, S.; Sessa, P. G.; Hardin, C.; Bhupatiraju, S.; Hussenot, L.; Mesnard, T.; Shahriari, B.; Ramé, A.; et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Voita, E.; Sennrich, R.; and Titov, I. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. *arXiv preprint arXiv:1909.01380*.

Wang, J.; Ji, J.; Jiang, Z.; and Sun, L. 2022. Traffic flow prediction based on spatiotemporal potential energy fields. *IEEE Transactions on Knowledge and Data Engineering*, 35(9): 9073–9087.

Wang, J.; Lin, Y.; and Li, Y. 2025. GTG: Generalizable Trajectory Generation Model for Urban Mobility. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 834–842.

Wang, J.; Yang, C.; Jiang, X.; and Wu, J. 2023a. WHEN: A Wavelet-DTW hybrid attention network for heterogeneous time series analysis. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 2361–2373.

Wang, M.; Chu, J.; Xie, S.; Zang, X.; Zhao, Y.; Zhong, W.; and Zhao, X. 2025a. Put Teacher in Student's Shoes: Cross-Distillation for Ultra-compact Model Compression Framework. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, 4975–4985.

Wang, M.; Zhao, X.; Guo, R.; and Wang, J. 2025b. Metalora: Tensor-enhanced adaptive low-rank fine-tuning. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, 4680–4684. IEEE.

Wang, M.; Zhao, Y.; Liu, J.; Chen, J.; Zhuang, C.; Gu, J.; Guo, R.; and Zhao, X. 2023b. Large multimodal model compression via efficient pruning and distillation at AntGroup. *arXiv preprint arXiv:2312.05795*.

Xue, J.; Sun, S.; Liu, M.; Wang, Y.; Liu, Z.; and Wang, J. 2025a. Learnable Sparse Customization in Heterogeneous Edge Computing. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, 58–71. IEEE Computer Society.

Xue, J.; Sun, S.; Liu, M.; Wang, Y.; Meng, X.; Wang, J.; Zhang, J.; and Xu, K. 2025b. Burst-Sensitive Traffic Forecast Via Multi-Property Personalized Fusion in Federated Learning. *IEEE Transactions on Mobile Computing*.

Yang, Y.; Wang, J.; Yu, X.; and Tang, Y. 2025. HygMap: Representing All Types of Map Entities via Heterogeneous Hypergraph. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*.

Yu, X.; Wang, J.; Yang, Y.; Huang, Q.; and Qu, K. 2025. BIGCity: A universal spatiotemporal model for unified trajectory and traffic state data analysis. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, 4455–4469. IEEE.

Zhang, Q.; Chen, M.; Bukharin, A.; Karampatziakis, N.; He, P.; Cheng, Y.; Chen, W.; and Zhao, T. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*.

Zhang, W.; Li, X.; Dong, K.; Wang, Y.; Jia, P.; Li, X.; Zhang, Y.; Xu, D.; Du, Z.; Guo, H.; et al. 2025. Process vs. Outcome Reward: Which is Better for Agentic RAG Reinforcement Learning. *arXiv preprint arXiv:2505.14069*.

Zhao, B.; Tu, H.; Wei, C.; Mei, J.; and Xie, C. 2023. Tuning layernorm in attention: Towards efficient multi-modal llm finetuning. *arXiv preprint arXiv:2312.11420*.