Parameter-Efficient Tuning in Large Language Models: A Survey

www.surveyx.cn

Abstract

Parameter-efficient tuning (PET) has emerged as a transformative approach in optimizing large language models (LLMs) by reducing computational costs and resource requirements while maintaining or enhancing performance. This survey comprehensively explores advancements in PET techniques, including null space optimization, which selectively adjusts a subset of parameters to enhance efficiency. The survey highlights the significance of PET in privacy-sensitive environments, such as federated learning, where data cannot be aggregated centrally. It categorizes PET methods into additive, selective, reparameterized, and hybrid approaches, detailing innovations like Adapter tuning, Prefix tuning, and Low-Rank Adaptation (LoRA). These methods demonstrate substantial improvements in model efficiency, adaptability, and performance across diverse applications, including natural language processing and computer vision. The survey also examines challenges in scalability, generalization, and ethical considerations, providing insights into future research opportunities. By addressing these challenges and leveraging methodological innovations, PET techniques promise to advance the deployment of LLMs in real-world scenarios, achieving significant memory savings and computational efficiency. The findings underscore the potential for PET to revolutionize model tuning, offering scalable, adaptable, and efficient solutions for optimizing LLMs across a wide range of applications.

1 Introduction

1.1 Concept of Parameter-Efficient Tuning

Parameter-efficient tuning (PET) enhances the efficiency of large language models (LLMs) by reducing the substantial costs associated with training and inference [1]. This approach is particularly crucial in privacy-sensitive environments, such as federated learning, where data cannot be centrally aggregated [2]. PET fine-tunes large pre-trained language models (PLMs) for specific downstream tasks while minimizing resource requirements, effectively addressing inefficiencies in traditional fine-tuning processes.

In LLMs, PET optimizes a limited set of task-specific parameters while largely keeping the pre-trained model unchanged, thereby decreasing the number of trainable parameters without sacrificing performance. This strategy is vital for real-world applications, where computational resources and time are constrained. By focusing on fewer parameters, PET significantly enhances the efficiency of PLMs and supports their implementation across diverse tasks. Recent studies indicate that PET methods can yield results comparable to full fine-tuning while optimizing far fewer parameters, making PET increasingly advantageous as model sizes grow [3, 4].

1.2 Importance and Motivation

The increasing size and complexity of LLMs impose significant computational and memory demands, presenting challenges for practical deployment [5]. Traditional fine-tuning methods require adjusting

Generated by SurveyX.cn.

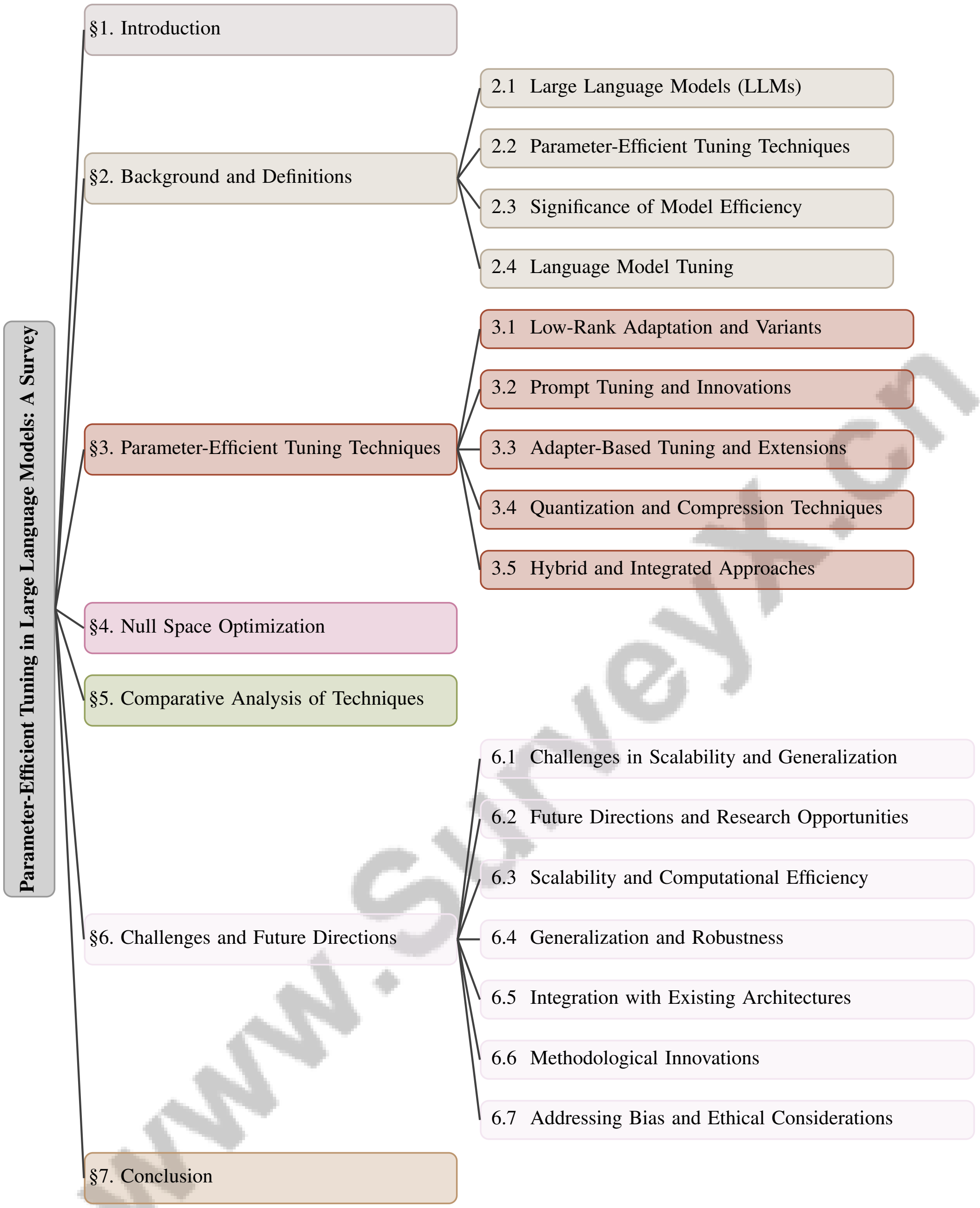

Parameter-Efficient Tuning in Large Language Models: A Survey
§1. Introduction
§2. Background and Definitions
2.1 Large Language Models (LLMs)
2.2 Parameter-Efficient Tuning Techniques
2.3 Significance of Model Efficiency
2.4 Language Model Tuning
§3. Parameter-Efficient Tuning Techniques
3.1 Low-Rank Adaptation and Variants
3.2 Prompt Tuning and Innovations
3.3 Adapter-Based Tuning and Extensions
3.4 Quantization and Compression Techniques
3.5 Hybrid and Integrated Approaches
§4. Null Space Optimization
§5. Comparative Analysis of Techniques
§6. Challenges and Future Directions
6.1 Challenges in Scalability and Generalization
6.2 Future Directions and Research Opportunities
6.3 Scalability and Computational Efficiency
6.4 Generalization and Robustness
6.5 Integration with Existing Architectures
6.6 Methodological Innovations
6.7 Addressing Bias and Ethical Considerations
§7. Conclusion

Figure 1: chapter structure

all parameters, resulting in inefficiencies and high resource demands that hinder the deployment of PLMs [6]. PET emerges as a critical strategy to alleviate these issues by optimizing a small subset of parameters, thereby lowering computational costs while preserving model performance [7]. This method is particularly essential in memory-constrained environments, saving approximately 30

The motivation for this survey is multifaceted. It aims to address inefficiencies in traditional fine-tuning methods, which necessitate impractical full parameter optimization and limit the reusability of PLMs due to task-specific output layers [8]. Additionally, the survey explores advancements in Parameter-Efficient Fine-Tuning (PEFT) for large models, focusing on optimizing the fine-tuning process to reduce computational costs while maintaining performance [9]. The limitations of existing PET methods in low-data scenarios, particularly in fine-tuning large pre-trained transformers for computer vision tasks, further emphasize the need for a comprehensive analysis and the development of robust techniques [10].

This survey systematically analyzes and advances PET techniques, aiming to maintain or enhance model performance while significantly reducing resource requirements. By examining various algorithmic advancements, including scaling laws, architectural innovations, and tuning strategies, this survey provides a thorough overview of optimizing LLM efficiency and effectiveness across applications [3, 4, 11, 12]. The exploration of PET strategies is vital for overcoming inefficiencies caused by traditional fine-tuning methods and advancing the field toward more scalable and adaptable solutions.

1.3 Survey Objectives

This survey's primary objective is to explore advancements in PET techniques, assessing their effectiveness in enhancing model efficiency and performance across various applications. A key focus is on continual learning scenarios, where methods such as Importance-aware Sparse Tuning (IST) dynamically update the most critical layers in PEFT modules based on importance scores. The survey introduces OpenDelta, an open-source library for the parameter-efficient adaptation of pre-trained models (PTMs) through plug-and-play delta tuning methods, exemplifying practical PET implementation [13].

Another important goal is to investigate scalable instruction-following capabilities in LLMs without relying on traditionally annotated data, thus improving adaptability and usability [14]. The survey also examines innovative PET techniques, particularly their effectiveness in long-document classification tasks, addressing specific challenges in natural language processing [15]. Furthermore, methods enhancing the controllability and alignment of LLMs with human preferences are explored to ensure models remain responsive to user needs [16].

Additionally, the survey investigates methods such as Adaptive Pruning and Tuning (APT), which effectively balance training and inference efficiency without compromising task performance [1]. It benchmarks various PET methods in federated learning environments, emphasizing privacy and resource constraints [2]. New methods like KronA are also developed to enhance efficiency and effectiveness in fine-tuning large PLMs [5].

The survey proposes a unified tuning method, termed plugin-tuning, which reformulates classification tasks into a common language modeling task, thereby improving efficiency and reducing parameter counts [8]. It covers a range of PEFT algorithms and their applications across diverse domains, including natural language processing and computer vision, highlighting the versatility and broad applicability of these techniques [9].

Through these objectives, the survey aims to advance the field of PET, offering strategies for scalable, adaptable, and efficient solutions in deploying large language models, ultimately achieving significant memory savings while maintaining model performance [7].

1.4 Structure of the Survey

The survey is systematically organized into several key sections, each addressing different aspects of parameter-efficient tuning (PET) in large language models (LLMs). The introductory section outlines the concept, importance, and motivation behind PET, establishing the foundation for the survey's objectives. The subsequent section delves into the background and definitions, providing a comprehensive overview of core concepts related to PET, including definitions of LLMs and related tuning techniques.

Following this, the survey explores various PET techniques, such as low-rank adaptation, prompt tuning, adapter-based tuning, and hybrid approaches, highlighting their advantages, limitations, and applications. A dedicated section on null space optimization examines its mathematical foundations and its impact on model performance.

The survey then conducts a comparative analysis of different PET techniques, evaluating their effectiveness across various applications and datasets, including their efficiency in low-resource and few-shot learning scenarios. The penultimate section addresses the challenges and future directions in PET, exploring issues related to scalability, generalization, and ethical considerations.

Finally, the conclusion summarizes key findings and insights from the survey, reiterating the significance of PET in enhancing model efficiency and performance. Each section is meticulously referenced, drawing on a wide range of scholarly sources to support the analysis and discussions presented throughout the paper.The following sections are organized as shown in Figure 1.

2 Background and Definitions

2.1 Large Language Models (LLMs)

Large Language Models (LLMs) represent a significant advancement in natural language processing, characterized by their vast parameter sizes, often reaching billions [17]. These models excel in generating and understanding human-like text, supporting a wide range of applications from text generation to complex language comprehension [1]. However, their deployment is often hindered by inefficiencies, mainly due to the necessity for task-specific output layers, which complicate fine-tuning [8].

The high computational costs and resource demands of fine-tuning LLMs present substantial challenges, limiting their adaptability across diverse tasks [9]. Despite these challenges, LLMs have revolutionized NLP, enabling tasks requiring a deep understanding of context and semantics. Establishing scaling laws for linear complexity in language models serves as a benchmark for evaluating various architectures and guiding future model development [17].

The necessity for efficient tuning methods is underscored by the significant training and inference costs of LLMs, prompting the development of innovative strategies to optimize performance while ensuring computational feasibility [1]. Addressing these challenges will improve the adaptability of LLMs across multiple domains, aligning them with user expectations and facilitating broader real-world deployment.

2.2 Parameter-Efficient Tuning Techniques

Parameter-efficient tuning (PET) techniques are crucial for optimizing LLMs, allowing adaptation to new tasks with minimal computational overhead. These methods focus on modifying a limited subset of parameters, reducing resource demands while preserving or enhancing model performance. This survey categorizes PET methods into additive, selective, reparameterized, and hybrid approaches, providing a framework for existing research [9].

Key techniques include Adapter tuning, Prefix tuning, and Low-Rank Adaptation (LoRA). Adapter tuning integrates small trainable modules within pre-trained model layers, enabling task-specific adaptations without overhauling the entire model structure. Prefix tuning optimizes continuous prefix vectors in transformer layers, allowing dynamic adjustments based on task needs [6]. LoRA reduces the number of trainable parameters through low-rank decomposition, enhancing fine-tuning efficiency [7].

Innovative methods such as Adaptive Pruning and Tuning (APT) optimize parameter selection by adaptively pruning less critical parameters, thereby increasing training and inference efficiency [1]. The MEET method employs prompt tuning and low-rank adaptation to optimize control tokens, enhancing alignment with human preferences and improving model controllability [16].

In federated learning contexts, PET methods like adapter tuning, prefix tuning, LoRA, and BitFit effectively address privacy and resource constraints [2]. The PET-GC approach exemplifies innovation by adapting modules from small language models to larger ones without gradient-based optimization [7].

The PVP framework leverages pre-training of PET modules to enhance performance in few-shot scenarios, highlighting the importance of pre-training in improving tuning methods' adaptability and efficiency [10]. Additionally, Generative Entropy-regularized Matching (GEM) reformulates supervised fine-tuning as a distribution matching problem with entropy regularization, mitigating overfitting and enhancing output diversity [18].

Advancements in PET techniques, including gradient-free approaches, significantly enhance LLM deployment. Techniques such as Adapter tuning, Prefix tuning, and LoRA enable updates to a small subset of parameters, improving adaptability and reducing resource consumption. A new gradient-free method offers up to 5.7 times memory savings compared to traditional PET methods, enhancing LLM versatility and efficiency across various real-world applications while maintaining competitive performance levels [7, 19, 8, 3, 20].

2.3 Significance of Model Efficiency

Model efficiency is critical for LLMs due to the substantial computational and financial resources required for deployment and fine-tuning [11]. Traditional fine-tuning, which involves adjusting all model parameters, incurs significant costs, especially for models with billions of parameters [21]. These demands are particularly evident in graph representation learning, emphasizing the need for more efficient methods [22].

PET techniques have emerged as a practical solution, enabling fine-tuning of large models while significantly reducing memory requirements and preserving performance. Techniques like MeZO illustrate how PET can facilitate efficient LLM adaptation, especially in low-resource settings where traditional methods may be unfeasible [23]. The high costs associated with fine-tuning large vision-language models further highlight the necessity for efficient tuning methods [24].

In federated learning scenarios, model efficiency is vital as it minimizes communication overhead and local adaptation costs for clients with limited resources, enhancing the feasibility of deploying LLMs in decentralized environments [2]. Efficient models also reduce reliance on extensive manual annotation, improving instruction-following capabilities in LLMs without incurring additional resource costs [14].

The importance of model efficiency is underscored by challenges in resource-limited scenarios, where existing methods often fail to enhance inference efficiency or lead to additional costs [1]. The irregular memory access patterns introduced by sparsity in current methods necessitate innovative approaches to optimize compute resource utilization [25]. PET techniques address these challenges by optimizing a subset of parameters, effectively reducing memory and time complexity, thus facilitating practical applications [26].

Additionally, existing benchmarks frequently lack the capacity to comprehensively evaluate LLMs across diverse optimization tasks, neglecting the sensitivity of LLMs to problem dimensions and sample sizes [27]. This highlights the need for robust assessments of PET methods, as their potential to supplant traditional fine-tuning relies on reliable performance metrics [20].

Model efficiency is essential for addressing resource allocation challenges and minimizing the environmental impact of training LLMs, enabling broader application across various fields while maintaining high performance and reducing resource consumption [11].

2.4 Language Model Tuning

Language model tuning is crucial for optimizing LLM performance on specific tasks, involving the adjustment of a pre-trained model's parameters to enhance its text understanding and generation capabilities. Techniques such as Regularized Mask Tuning (R-MT) can reveal latent knowledge within pre-trained models, improving their transferability and performance [28].

Tuning enables LLMs to be fine-tuned for downstream tasks, ensuring adaptability to new contexts without extensive retraining. This adaptability is crucial for LLMs operating across diverse domains, including natural language processing and computer vision. Effective fine-tuning minimizes the need for extensive manual adjustments while enhancing deployment in real-world applications, as evidenced by studies demonstrating the advantages of parameter-efficient tuning strategies. These strategies maintain foundational capabilities while improving task-specific performance, such as in medical knowledge retrieval and automated program repair, facilitating scalable personalization [29, 30, 31, 32].

Tuning is essential for aligning LLMs with specific user preferences, enabling the generation of outputs that are both accurate and contextually appropriate. This alignment often employs advanced techniques such as parameter-efficient tuning, allowing for the integration of multiple user preferences during training and inference. Methods like controllable generation enhance LLM responses based on varying user needs, significantly advancing AI-driven solutions' performance and safety [33, 31, 16].

AI-generated, for reference only.

This alignment is particularly critical in applications where LLMs engage with users in a natural and intuitive manner.

3 Parameter-Efficient Tuning Techniques

Category	Feature	Method
Low-Rank Adaptation and Variants	Model Efficiency Techniques	PVP[10], PET-GC[7]
Prompt Tuning and Innovations	Parameter Efficiency	PT[34], KronA[5]
	Instance and Input Sensitivity	LPT[35], APT[6]
	Multi-Modal Capabilities	VPT[36]
	Control and Alignment	MEET[16]
Adapter-Based Tuning and Extensions	Efficiency-Enhancing Techniques	ABT[37], SiRA[38], HiDe-PET[39], COLA[40]
	Layer-Specific Adaptations	PP[15], IST[41]
Quantization and Compression Techniques	Structured Optimization	EfficientQAT[42], EP[43]
	Federated and Privacy Techniques	FP[44], DPA[45]
	Pruning and Tuning	AT[46], PEQA[47], LRP[48]
Hybrid and Integrated Approaches	Adaptive Network Structures	OD[13]
	Efficient Update Mechanisms	PMLM[49], PCFT[50]
	Optimization Techniques	SI[51], DFO-LoRA[52]
	Integrated Task Management	PT[8]

Table 1: This table provides a comprehensive overview of parameter-efficient tuning techniques categorized into five main areas: Low-Rank Adaptation and Variants, Prompt Tuning and Innovations, Adapter-Based Tuning and Extensions, Quantization and Compression Techniques, and Hybrid and Integrated Approaches. Each category lists specific features and corresponding methods, highlighting recent advancements and innovations aimed at enhancing the efficiency and adaptability of large language models (LLMs). The table serves as a detailed reference for understanding the landscape of parameter-efficient tuning methods and their applications.

The advancement of parameter-efficient tuning techniques has significantly enhanced the performance and adaptability of large language models (LLMs). This section explores key strategies such as Low-Rank Adaptation (LoRA) and its variants, which have become instrumental in reducing the number of trainable parameters while maintaining model effectiveness. By understanding LoRA's principles and innovations, we gain insights into its role within the broader parameter-efficient tuning methodologies. Table 1 presents a detailed classification and summary of parameter-efficient tuning techniques, illustrating their diverse applications and innovations in optimizing large language models. Additionally, **??** illustrates the hierarchical structure of these tuning techniques, encompassing core concepts and recent advancements across various approaches, including Low-Rank Adaptation, Prompt Tuning, Adapter-Based Tuning, Quantization and Compression Techniques, and Hybrid and Integrated Approaches. Each category within the figure highlights key innovations and methodologies aimed at enhancing model adaptability, efficiency, and performance across diverse applications, thereby providing a comprehensive overview of the landscape of parameter-efficient tuning.

3.1 Low-Rank Adaptation and Variants

Low-Rank Adaptation (LoRA) is a pivotal technique within parameter-efficient tuning frameworks, aimed at mitigating the computational challenges of fine-tuning large language models (LLMs). LoRA reduces trainable parameters by decomposing weight matrices into lower-dimensional structures, preserving model performance and making it advantageous in resource-limited settings [7]. This technique efficiently adapts pre-trained models to new tasks with minimal overhead, facilitating diverse applications.

Recent enhancements in LoRA have increased its flexibility and applicability. Integrating small, task-specific language models into larger LLMs via a Bridge model optimizes task-specific learning [7]. This integration exemplifies how LoRA can synergize with other parameter-efficient methods to boost model adaptability and efficiency.

The PVP framework showcases LoRA's potential by pre-training parameter-efficient tuning modules on extensive datasets, improving performance in low-data scenarios [10]. Innovations like plugin-tuning offer a unified approach to classification tasks, reusing the same model head across different tasks without adding parameters [8]. This framework underscores the versatility of low-rank adaptation in optimizing LLMs for varied applications.

Advancements in low-rank adaptation techniques highlight significant evolution, leading to scalable and resource-efficient solutions for fine-tuning large-scale pre-trained models. Innovations such as

Sparse Mixture of Low Rank Adaptation (SiRA) and Dynamic Low-Rank Adaptation (DyLoRA) enhance performance by optimizing parameter efficiency while addressing stability and convergence issues. Frameworks like LoRAPrune demonstrate the potential for structured pruning to further reduce memory usage and improve performance [53, 54, 55, 38, 48]. As the field progresses, integrating LoRA with other parameter-efficient tuning methods offers valuable insights into optimizing LLMs across diverse applications and environments.

3.2 Prompt Tuning and Innovations

Prompt tuning has become integral to parameter-efficient tuning strategies, enabling large language models (LLMs) to adapt to various tasks with reduced computational costs. This technique optimizes continuous prompt vectors, guiding the model during inference and significantly decreasing the parameters required compared to traditional full model fine-tuning [34].

Recent advancements have expanded prompt tuning's scope and effectiveness. Late Prompt Tuning (LPT) inserts a late prompt into an intermediate model layer, generated based on hidden states, making it instance-dependent [35]. This method enhances task-related interaction by creating instance-aware prompts.

Approaches like MEET incorporate a two-step optimization process for control tokens before fine-tuning, improving alignment with diverse human preferences [16]. This highlights the importance of control tokens in refining model outputs to meet human expectations.

In cross-lingual applications, prompt tuning outperforms traditional fine-tuning, especially in decoder-based models [56]. Visual Prompt Tuning (VPT) injects learnable parameters into the input space of Transformer layers, illustrating prompt tuning's versatility in extending LLMs to multi-modal applications [36].

Integrating Kronecker products within the low-rank adaptation framework enhances representational power during fine-tuning, offering a novel prompt tuning approach [5]. Adaptive Pruning and Tuning (APT) employs a gate mechanism for fine-grained token-level and coarse-grained layer-level adjustments, contrasting with fixed-length prefixes in previous methods [6].

These prompt tuning innovations highlight its pivotal role in parameter-efficient tuning, offering scalable, adaptable solutions for optimizing LLMs across a broad application spectrum. Ongoing advancements, like Late Prompt Tuning, are expected to enhance LLM performance and efficiency. This evolution facilitates scalable, personalized applications, such as abbreviation expansion for specific communication needs, and enhances model adaptability to diverse scenarios, broadening deployment potential. Methods like prompt tuning and retrieval-augmented generation optimize LLMs with limited training data, increasing practical application value [35, 29].

3.3 Adapter-Based Tuning and Extensions

Adapter-based tuning has emerged as a pivotal technique in parameter-efficient tuning, offering a robust framework for adapting large language models (LLMs) to specific tasks without extensive retraining [37]. This method enhances pre-trained model adaptability by focusing on adapter module parameter tuning, reducing overfitting risk and improving performance.

Recent advancements in adapter-based tuning introduce innovative approaches that optimize resource utilization and model efficiency. SiRA leverages a sparse mixture of experts to enhance tuning, improving resource utilization and performance [38]. This approach exemplifies sparse architecture integration into adapter-based frameworks for greater efficiency.

COLA improves traditional methods by allowing effective multiple low-rank updates without additional computational costs [40]. This capability maintains efficiency while enhancing LLM adaptability to diverse tasks.

Techniques like prefix-propagation enhance prefix contextual relevance across layers, improving adapter-based tuning effectiveness [15]. Ensuring contextual relevance optimizes tuning and enhances performance.

AI-generated, for reference only.

In continual learning, HiDe-PET uses mainstream parameter-efficient tuning techniques to tune pre-trained models, enhancing adaptability in dynamic environments [39]. This underscores adapter-based tuning's importance in scenarios requiring continual data and task adaptation.

Dynamic layer selection methods, like IST, enhance adapter-based tuning efficiency and effectiveness. Dynamically selecting important layers improves fine-tuning compared to uniform approaches, highlighting potential for more targeted, efficient strategies [41].

These adapter-based tuning advancements reflect a trend towards more efficient, adaptable methods optimizing LLM performance across varied applications. As the field advances, incorporating techniques like sparse mixtures, dynamic layer selection, and contextual prefix propagation is expected to enhance efficiency and effectiveness, addressing challenges like slow convergence and overfitting while enabling robust natural language processing task performance [57, 54, 37, 58, 59].

To further illustrate these concepts, Figure 2 presents a figure that illustrates the hierarchical structure of adapter-based tuning methods and their extensions, highlighting key methods, innovative techniques, and challenges in the field. This visual representation enhances our understanding of the complex relationships and developments within adapter-based tuning.

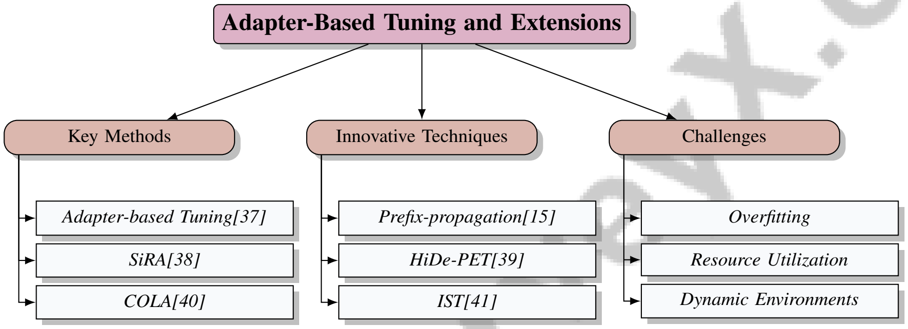


Figure 2: This figure illustrates the hierarchical structure of adapter-based tuning methods and their extensions, highlighting key methods, innovative techniques, and challenges in the field.

3.4 Quantization and Compression Techniques

Quantization and compression techniques are crucial for improving large language models' (LLMs) parameter efficiency, significantly reducing memory consumption and computational demands during fine-tuning and deployment while maintaining or enhancing performance. Methods like Parameter-Efficient and Quantization-aware Adaptation (PEQA) combine parameter-efficient fine-tuning with quantization, reducing model size and memory overhead without compromising quantization structure integrity. Efficient Quantization-Aware Training (EfficientQAT) optimizes training by minimizing accuracy loss in low-bit quantization scenarios, enabling efficient billion-parameter model training. These advancements address LLM deployment challenges in practical applications [47, 42, 11].

LoRAPrune integrates LoRA weights into pre-trained weights while pruning redundant channels and heads, reducing model size and computational requirements, facilitating parameter-efficient tuning without additional overhead [48]. EfficientQAT applies quantization-aware training to larger models, significantly reducing training time and memory usage while maintaining performance [42].

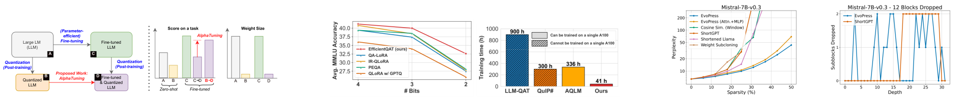
In federated learning, FedPipe incorporates model parameter quantization to minimize memory usage, essential for parameter-efficient tuning in distributed environments. This underscores quantization's importance in reducing federated learning communication and computational costs [44]. PEQA leverages quantization to decrease fine-tuning memory overhead while maintaining rapid inference performance, distinguishing itself by focusing on memory efficiency [47].

Dynamic compression techniques, like EvoPress, adjust compression levels across layers to ensure optimal accuracy while adhering to compression constraints. This dynamic approach fine-tunes compression strategies to balance performance and resource efficiency effectively [43].

Differential privacy techniques, like DPA, enhance adaptation while preserving privacy. Unlike traditional methods like DPSGD, which often drop performance, DPA maintains performance, highlighting privacy-preserving adaptation potential in parameter-efficient tuning [45].

AI-generated, for reference only.

These quantization and compression techniques represent significant parameter-efficient tuning advancements, offering scalable, adaptable solutions for optimizing LLMs across applications. Implementing innovative optimization algorithms and memory-efficient fine-tuning techniques significantly lowers LLM computational and memory requirements. This enhancement facilitates widespread real-world application, ensuring LLMs maintain high efficiency and effectiveness across diverse tasks, addressing substantial resource demand challenges [60, 47, 11].



(a) Quantization and Fine-tuning in Large Language Models: A Comparative Study[46]

(b) EfficientQAT: A Low-Resource Pre-training Model for Large Language Models[42]

(c) Comparison of Perplexity and Subblock Dropped for Various Models on Mistral-7B-v0.3 and Mistral-7B-v0.3 - 12 Blocks Dropped[43]

Figure 3: Examples of Quantization and Compression Techniques

As shown in Figure 3, parameter-efficient tuning techniques like quantization and compression are pivotal for enhancing large language models (LLMs) while reducing computational demands. Figure 3 illustrates practical applications and comparative analyses of these techniques. "Quantization and Fine-tuning in Large Language Models: A Comparative Study" explores quantization versus traditional fine-tuning efficacy, showcasing their impact on task performance and weight size. "EfficientQAT: A Low-Resource Pre-training Model for Large Language Models" highlights EfficientQAT's performance in resource-constrained environments, emphasizing the trade-off between quantization bits and accuracy. The "Comparison of Perplexity and Subblock Dropped for Various Models on Mistral-7B-v0.3 and Mistral-7B-v0.3 - 12 Blocks Dropped" provides insights into model sparsity's effect on perplexity, offering a nuanced understanding of model efficiency. These examples underscore quantization and compression techniques' significance in advancing LLMs by balancing performance with resource efficiency [46, 42, 43].

3.5 Hybrid and Integrated Approaches

Hybrid and integrated approaches to parameter-efficient tuning have emerged as pivotal strategies in enhancing the adaptability and efficiency of large language models (LLMs). These approaches synergize various tuning methodologies to create robust frameworks for optimizing model performance across diverse applications. The PORTLLM method exemplifies such a hybrid strategy by integrating fine-tuning with a lightweight update mechanism, facilitating efficient personalization of LLMs [49]. Similarly, the integration of low-rank modules with derivative-free optimization techniques offers a novel hybrid approach, optimizing parameter-efficient tuning by reducing the reliance on gradient-based updates [52].

The OpenDelta framework provides another example of hybrid integration, enabling flexible adaptation of pre-trained models (PTMs) by dynamically rerouting tensors through delta modules, which enhances the adaptability and efficiency of model updates [13]. In the realm of personalized tuning, PCFT emphasizes the importance of individualized model updates rather than a single global model, thereby minimizing communication overhead while enhancing personalization [50].

Smart-Infinity represents an innovative hybrid approach by executing parameter updates on near-storage processing devices, optimizing data transfer and enhancing the efficiency of parameter-efficient tuning processes [51]. This method exemplifies the potential of integrating hardware-level optimizations with model tuning strategies to achieve superior performance.

Plugin-tuning further illustrates the versatility of hybrid approaches by consolidating insights from various classification tasks into a unified framework, thereby improving efficiency in parameter usage and enhancing model adaptability across tasks [8]. This approach underscores the potential for hybrid methodologies to streamline the tuning process and optimize resource allocation.

Collectively, these hybrid and integrated approaches represent significant advancements in the field of parameter-efficient tuning, offering scalable and adaptable solutions for optimizing LLMs. By integrating various fine-tuning methodologies, such as full-parameter and parameter-efficient tuning, these innovative approaches significantly improve the adaptability and resilience of Large Language Models (LLMs). This enhancement not only facilitates their effective deployment in diverse real-world applications, such as medical knowledge retrieval and personalized text entry, but also addresses the challenges of optimizing LLM architectures for better performance in dynamic environments. For instance, the Med42 model achieved a noteworthy 72% accuracy on the USMLE datasets, exemplifying the potential of these techniques to advance AI-driven healthcare solutions and other practical uses [29, 60, 31].

Feature	**Low-Rank Adaptation and Variants**	**Prompt Tuning and Innovations**	**Adapter-Based Tuning and Extensions**
Parameter Reduction	Decomposes Weight Matrices	Optimizes Prompt Vectors	Adapter Module Tuning
Adaptation Technique	Bridge Model Integration	Instance-aware Prompts	Sparse Architecture Integration
Application Scope	Resource-limited Settings	Cross-lingual Applications	Continual Learning

Table 2: This table provides a comparative analysis of three prominent parameter-efficient tuning techniques: Low-Rank Adaptation and its variants, Prompt Tuning and its innovations, and Adapter-Based Tuning along with its extensions. It highlights the unique features, adaptation techniques, and application scopes of each method, demonstrating their respective strengths in optimizing large language models for diverse applications.

4 Null Space Optimization

Exploring null space optimization requires a solid grasp of its foundational principles, which are crucial for effective implementation in parameter-efficient tuning. This section examines the conceptual underpinnings that guide strategies in null space optimization, emphasizing the role of derivative-free optimization methods and collaborative fine-tuning frameworks. Understanding these concepts enhances our appreciation for the subsequent techniques and implementations that leverage these principles to improve model adaptability and efficiency.

4.1 Conceptual Foundations

The theoretical foundations of null space optimization in parameter-efficient tuning focus on enhancing model adaptability while minimizing computational demands. A core aspect of this approach is the use of derivative-free optimization methods, which bypass gradient calculations, streamlining the tuning process and proving beneficial in resource-limited environments [52].

The effectiveness of null space optimization is illustrated through the Personalized Collaborative Fine-Tuning (PCFT) framework, which capitalizes on the collaborative strengths of users with similar data distributions. This framework underscores the importance of personalized tuning strategies that align with user data characteristics, thereby improving model performance while maintaining resource efficiency [50].

In the context of parameter-efficient tuning, federated learning offers a perspective that emphasizes performance sustainability and data privacy. This aligns with the principles of null space optimization, which focuses on optimizing a select subset of parameters to achieve efficient learning outcomes without compromising privacy [2].

Robustness in parameter-efficient tuning methods is further highlighted by comprehensive evaluation frameworks that integrate various adversarial attacks and information perturbations. Such frameworks deepen our understanding of the robustness and adaptability of tuning methods, distinguishing them from prior benchmarks that lacked this depth [61].

The conceptual foundations of null space optimization revolve around strategically manipulating select parameters to enhance learning efficiency and model robustness. By employing derivative-free optimization methods, collaborative tuning strategies, and comprehensive evaluation frameworks, these techniques provide a robust foundation for optimizing large language models (LLMs) in dynamic and resource-constrained environments. They enable parameter-efficient tuning without gradient computation, significantly reducing memory usage and computational costs while maintaining performance comparable to traditional fine-tuning methods. This is particularly advantageous for adapting LLMs to various tasks, enhancing convergence speed and facilitating effective model adaptation even under limited computational resources [27, 62, 7, 63, 52].

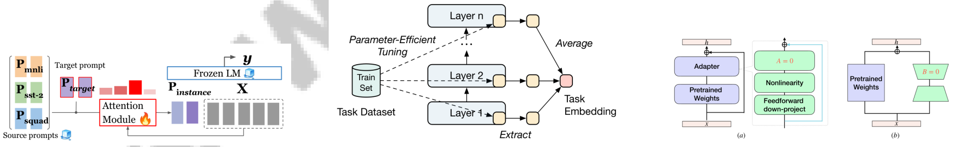
4.2 Techniques and Implementations

Null space optimization techniques in parameter-efficient tuning emphasize optimizing a subset of model parameters to enhance computational efficiency and adaptability. A key technique is late prompt tuning (LPT), which optimizes the interaction between prompts and model outputs while maintaining a shorter propagation path for task-related information. This ensures efficient capture and leveraging of task-specific nuances during the tuning process [35].

The implementation of null space optimization is exemplified by derivative-free optimization methods, which eliminate the need for gradient calculations. This not only streamlines the optimization process by reducing computational overhead but also enhances robustness in few-shot learning scenarios, as demonstrated by advancements in parameter-efficient tuning methods for large language models. Techniques such as low-rank adaptation enable significant memory savings and improved convergence speed, yielding performance comparable to traditional gradient-based optimization while minimizing resource demands [64, 27, 7, 65, 52]. This is particularly beneficial in environments where traditional gradient-based methods are impractical. By concentrating on the most impactful parameters, these techniques streamline the tuning process, allowing LLMs to adapt to new tasks with minimal resource expenditure.

Moreover, null space optimization frequently utilizes collaborative fine-tuning frameworks that strategically leverage user data distributions, enhancing model personalization and performance through parameter-efficient tuning methods. These approaches focus on fine-tuning a small subset of model parameters, improving stability and generalization capabilities while addressing training efficiency and convergence challenges associated with traditional full model fine-tuning [54, 58]. This collaborative approach aligns with null space optimization principles, ensuring parameter updates are efficient and tailored to user needs.

The techniques and implementations of null space optimization in parameter-efficient tuning highlight the significance of targeted parameter adjustments for efficient model adaptation. By employing advanced techniques like late prompt tuning and derivative-free optimization, these methods offer effective solutions for enhancing LLMs across various optimization tasks, especially in resource-limited settings. This approach leverages LLMs' extensive domain knowledge to facilitate intelligent modeling and strategic decision-making while addressing computational challenges associated with complex problems. Integrating LLMs with optimization algorithms enables iterative solution generation and evaluation, ensuring robust performance in optimizing small-scale problems and enhancing overall output quality across diverse applications [27, 60].



(a) Attention Module for Cross-Modality Prompting in Language Models[66]

(b) Parameter-Efficient Tuning for Efficient Neural Network Training[67]

(c) Adaptive Nonlinear Down-Projecting for Efficient Learning[58]

Figure 4: Examples of Techniques and Implementations

As shown in Figure 4, the examples of "Null Space Optimization; Techniques and Implementations" provide an overview of innovative methods and their applications in neural network efficiency and cross-modality prompting. The techniques include an "Attention Module for Cross-Modality Prompting in Language Models," "Parameter-Efficient Tuning for Efficient Neural Network Training," and "Adaptive Nonlinear Down-Projecting for Efficient Learning." Each technique is visualized through schematic diagrams highlighting their unique components and operational strategies. For instance, the attention module enhances model adaptability to different datasets by integrating source and target prompts through an attention mechanism. Parameter-efficient tuning is depicted as a multi-layered neural network architecture employing "Extract" and "Average" operations to refine task embeddings, optimizing the training process by efficiently leveraging network parameters. Lastly, the adaptive nonlinear down-projecting technique minimizes parameter requirements while maintaining learning efficiency through adapter layers and feedforward down-projecting layers.

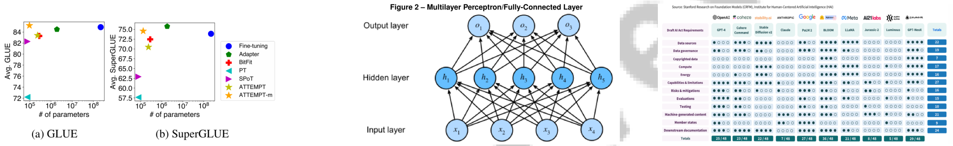
Together, these examples underscore the importance of optimizing neural network architectures to enhance performance while reducing computational demands [66, 67, 58].

4.3 Comparative Analysis with Other Techniques

Null space optimization distinguishes itself among parameter-efficient tuning techniques by optimizing a select subset of parameters, minimizing computational overhead while preserving model performance. This technique is particularly advantageous in resource-constrained scenarios, relying on derivative-free optimization methods that eliminate the need for gradient calculations [52]. In contrast, traditional gradient-based methods often demand significant computational resources, which can be impractical in such environments.

When compared to other parameter-efficient tuning methods like low-rank adaptation (LoRA) and adapter-based tuning, null space optimization offers superior computational efficiency. LoRA reduces the number of trainable parameters by decomposing weight matrices into lower-dimensional structures, which, while effective, still involves gradient-based updates [7]. Adapter-based tuning inserts small trainable modules within the model, potentially increasing architectural complexity and reducing computational efficiency [37].

Furthermore, null space optimization's emphasis on personalized and collaborative fine-tuning frameworks aligns well with federated learning scenarios, where data privacy and minimal communication overhead are critical [2]. This ensures that parameter updates are efficient and tailored to the specific needs of the user base, providing a significant advantage over more generalized tuning methods.



Source: Stanford Research on Foundation Models (CRFM), Institute for Human-Centered Artificial Intelligence (HAI)

	OpenAI	cohere	stability.ai	ANTHROPIC	Google	BigScience	Meta	AI21labs	ALEPH ALPHA	EleutherAI	
Draft AI Act Requirements	GPT-4	Cohere Command	Stable Diffusion v2	Claude	PaLM 2	BLOOM	LLaMA	Jurassic-2	Luminous	GPT-NeoX	Totals
Data sources	●○○○	●●●○	●●●●	○○○○	●●○○	●●●●	●●●●	○○○○	○○○○	●●●●	22
Data governance	●●○○	●●●○	●●○○	○○○○	●●●○	●●●●	●●○○	○○○○	○○○○	●●●○	19
Copyrighted data	○○○○	○○○○	○○○○	○○○○	○○○○	●●●○	○○○○	○○○○	○○○○	●●●●	7
Compute	○○○○	○○○○	●●●●	○○○○	○○○○	●●●●	●●●●	○○○○	●○○○	●●●●	17
Energy	○○○○	●○○○	●●●○	○○○○	○○○○	●●●●	●●●●	○○○○	○○○○	●●●●	16
Capabilities & limitations	●●●●	●●●○	●●●●	●○○○	●●●●	●●●○	●●○○	●●○○	●○○○	●●●○	27
Risks & mitigations	●●●○	●●○○	●○○○	●○○○	●●●○	●●○○	●○○○	●●○○	○○○○	●○○○	16
Evaluations	●●●●	●●○○	○○○○	○○○○	●●○○	●●●○	●●○○	○○○○	●○○○	●○○○	15
Testing	●●●○	●●○○	○○○○	○○○○	●●○○	●●○○	○○○○	●○○○	○○○○	○○○○	10
Machine-generated content	●●●○	●●●○	○○○○	●●●○	●●●○	●●●○	○○○○	●●●○	●○○○	●●○○	21
Member states	●●○○	○○○○	○○○○	●●○○	●●●●	○○○○	○○○○	○○○○	●○○○	○○○○	9
Downstream documentation	●●●○	●●●●	●●●●	○○○○	●●●●	●●●●	●●○○	○○○○	○○○○	●●●○	24
Totals	25 / 48	23 / 48	22 / 48	7 / 48	27 / 48	36 / 48	21 / 48	8 / 48	5 / 48	29 / 48	

(a) Comparison of GLUE and SuperGLUE Metrics Across Different Models and Training Strategies[66]

(b) Multilayer Perceptron/Fully-Connected Layer[68]

(c) Draft AI Act Requirements[69]

Figure 5: Examples of Comparative Analysis with Other Techniques

As shown in Figure 5, the "Null Space Optimization; Comparative Analysis with Other Techniques" section is illustrated through figures that provide a multi-faceted view of how various models and strategies perform across different metrics and requirements. The first figure offers a comparative analysis of GLUE and SuperGLUE metrics, highlighting performance distinctions across models and training strategies. The second figure showcases the architecture of a multilayer perceptron (MLP), fundamental to many optimization techniques, including null space optimization. The third figure presents a table detailing the compliance of various AI models with draft AI Act requirements, underscoring the regulatory landscape that AI models must navigate and providing context for assessing the advantages of null space optimization. Together, these figures offer a comprehensive view of the landscape within which null space optimization operates, enriching the comparative analysis with other optimization techniques [66, 68, 69].

5 Comparative Analysis of Techniques

A thorough comparative analysis of parameter-efficient tuning techniques reveals their practical implications and applications in optimizing large language models (LLMs) across various domains. This section explores specific applications and case studies that illustrate the versatility and effectiveness of these techniques, providing insights into their implementation and outcomes.

5.1 Applications and Case Studies

Parameter-efficient tuning techniques have demonstrated significant versatility in optimizing LLMs for a variety of tasks. The MEET framework consistently outperforms traditional controllable

AI-generated, for reference only.

generation methods, achieving results comparable to Direct Preference Optimization (DPO) [16]. This effectiveness enhances model controllability and alignment with human preferences.

In supervised learning, the GEM framework has shown marked improvements on datasets like UltraFeedback, particularly in tasks such as instruction-following, mathematical reasoning, and code generation [18]. This underscores GEM's capacity to enhance model performance across diverse applications.

Smart-Infinity has been utilized with model architectures such as GPT-2 and BERT, demonstrating significant training speedups [51]. This capability facilitates efficient deployment of LLMs in resource-constrained environments while maintaining performance.

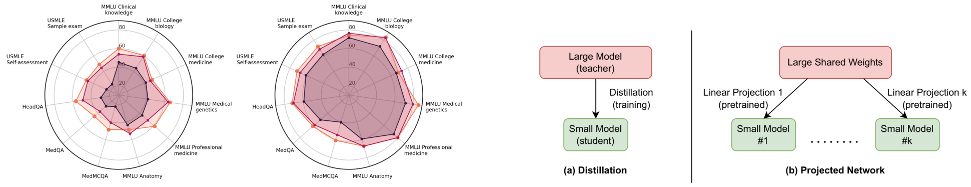
Benchmark evaluations on downstream tasks, including commonsense reasoning and information retrieval, highlight the practical applications of parameter-efficient tuning techniques in enhancing model adaptability and performance across different contexts [17]. These evaluations elucidate how these techniques can be effectively applied to optimize LLMs for specific tasks.

In federated learning, a systematic evaluation framework has been developed to assess the performance and privacy implications of parameter-efficient tuning methods [2]. This framework is pivotal for deploying these methods in decentralized environments, ensuring data privacy and minimizing communication overhead.

The KronA method, tested on the GLUE benchmark, exhibits significant accuracy and efficiency improvements compared to baseline methods like LoRA and Adapter [5]. This highlights KronA's potential for optimizing LLMs in natural language understanding tasks.

Extensive experiments on eight natural language understanding tasks from the SuperGLUE benchmark validate the effectiveness of parameter-efficient tuning methods, demonstrating comparable performance to traditional fine-tuning approaches [7]. Additionally, the APT method has been evaluated on the SuperGLUE and Named Entity Recognition datasets, showcasing its effectiveness against baseline methods [6].

Collectively, insights from various applications and case studies emphasize the versatility and efficacy of parameter-efficient tuning techniques in optimizing LLMs across a wide array of tasks. These methods not only enhance LLM performance in real-world scenarios, such as personalized text entry, but also maintain high resource efficiency. For instance, techniques like prompt-tuning and retrieval-augmented generation have shown substantial improvements in relevance and accuracy, even with limited training data. Furthermore, frameworks like FedCoLLM illustrate how these strategies can facilitate knowledge transfer between large and small language models while preserving data privacy and minimizing computational overhead, paving the way for effective AI applications in specialized fields, including healthcare [31, 70, 7, 20, 29].



(a) Comparison of Medical Knowledge Assessment Methods[31]

(b) Distillation and Projected Network[71]

Figure 6: Examples of Applications and Case Studies

As illustrated in Figure 6, the "Comparative Analysis of Techniques; Applications and Case Studies" serves as a crucial example in evaluating methodologies within specialized fields. The first figure, "Comparison of Medical Knowledge Assessment Methods," employs a radar chart to compare the effectiveness of MedQA and MedMCQA, mapping performance across various medical knowledge areas. The second figure, "Distillation and Projected Network," contrasts the distillation process, where knowledge is transferred from a large model to a smaller one, with projected networks that involve pretraining on extensive datasets followed by weight projection. Together, these figures illuminate the strengths and limitations of each method, emphasizing the importance of tailored approaches in optimizing performance within specific domains [31, 71].

5.2 Performance Evaluation Across Diverse Datasets

Benchmark	Size	Domain	Task Format	Metric
LMTCC[72]	1,000,000	Text Classification	Multiclass	F1-score
PELT[68]	1,000	Legal Text Classification	Text Classification	F1-score
Med42[31]	411,064	Medical Question Answering	Question Answering	Accuracy
FT-LLM[62]	1,000	Dialogue Generation	Causal Language Modeling	Memory Usage, Runtime
PET-DA[73]	3,000	Text Classification	Text Classification	Accuracy
FedPETuning[74]	1,000,000	Natural Language Inference	Text Classification	Accuracy, F1 Score
APR-Instruction[32]	30,000	Automated Program Repair	Instruction Generation	pass@10
LaMP[75]	37,560	Text Generation	Personalized Text Generation	Accuracy, F1

Table 3: The table provides a comprehensive overview of various benchmarks used for evaluating parameter-efficient tuning techniques across diverse domains. It includes details on the size, domain, task format, and evaluation metrics for each benchmark, highlighting their relevance in assessing performance across different natural language processing tasks. This information is crucial for understanding the adaptability and effectiveness of these tuning methods in optimizing large language models.

Evaluating performance across diverse datasets is vital for understanding the effectiveness and adaptability of parameter-efficient tuning techniques in optimizing LLMs. Table 3 presents a detailed summary of the benchmarks utilized in the evaluation of parameter-efficient tuning techniques, illustrating their application across various domains and task formats. The Y-Tuning approach has been assessed through accuracy metrics across multiple classification tasks, demonstrating improvements in training speed and memory usage, highlighting its efficiency and applicability [21]. Similarly, evaluations across 16 text classification datasets reveal the robustness of parameter-efficient methods in addressing a variety of text processing challenges [72].

The Deconfounded Causality-aware Parameter-Efficient Fine-Tuning (DCA) technique has shown superior accuracy relative to baseline models and other parameter-efficient methods across multiple benchmarks, emphasizing its potential to enhance model performance while maintaining efficiency [76]. DyLoRA's evaluation based on accuracy and F1 scores across various ranks and tasks indicates its competitive performance and notable reduction in training time, making it a viable option for efficient model tuning [55].

Comprehensive experiments comparing various large language models and parameter-efficient fine-tuning (PEFT) methods across different tasks, including commonsense reasoning and arithmetic reasoning, highlight the versatility of these techniques [41]. The Q-PEFT method has been evaluated using Recall and Hit Rate metrics on datasets such as Natural Questions, TriviaQA, SQuAD, and WebQ, demonstrating its effectiveness in enhancing retrieval-based tasks [77].

The prefix-propagation technique has been evaluated on long-document classification tasks, yielding improvements in accuracy, precision, recall, and F1 scores compared to prefix-tuning and traditional fine-tuning methods, thus validating its efficacy in complex document classification challenges [15]. Additionally, the Adaptive Pruning and Tuning (APT) method has been tested on datasets including SST2, MNLI, and SQuAD v2.0, showcasing significant enhancements in task accuracy, training time, and memory usage [1].

Evaluations across a diverse array of datasets reinforce the effectiveness and adaptability of parameter-efficient tuning techniques in optimizing LLMs for various applications. These techniques can achieve performance levels comparable to full-parameter fine-tuning while significantly enhancing resource efficiency. For instance, the Med42 model, utilizing parameter-efficient tuning, achieved an accuracy of 72

5.3 Efficiency in Low-Resource and Few-Shot Scenarios

The efficiency of parameter-efficient tuning techniques in low-resource and few-shot learning scenarios is critical, as these contexts often present challenges due to limited data availability. Techniques such as Residual Prompt Tuning have proven effective in few-shot settings by enhancing model adaptability and performance with minimal data [78]. This approach refines prompt tuning using residual connections, ensuring robust performance even with scarce data.

Adapter-based tuning has also demonstrated remarkable efficiency in low-resource and cross-lingual scenarios, outperforming traditional fine-tuning methods by optimizing a smaller set of parameters tailored to specific tasks [37]. This adaptability across diverse linguistic contexts with limited data highlights its broad applicability.

HyperT5 exemplifies an innovative approach that achieves efficiency in low-resource and few-shot scenarios by generating effective task-specific parameters through a single forward pass [63]. This method streamlines the tuning process, reducing computational overhead while maintaining high performance, making it particularly suitable for resource-constrained applications.

The PVP framework enhances parameter-efficient tuning methods' performance in few-shot scenarios, achieving state-of-the-art results on benchmarks such as VTAB-1k [10]. By pre-training tuning modules on extensive datasets, PVP improves model adaptability and efficiency, showcasing the significant potential of pre-training strategies in few-shot learning contexts.

Collectively, these techniques underscore the adaptability and robustness of parameter-efficient tuning methods in low-resource and few-shot scenarios, offering scalable solutions that maintain high performance with minimal data and computational resources. As the demand for efficient model adaptation grows, recent parameter-efficient tuning methods—such as Adapter tuning, Prefix tuning, and LoRA—provide substantial advancements in optimizing LLMs for various applications. These techniques enable the updating of only a small subset of model parameters, thereby reducing computational costs and memory usage while ensuring robust performance in complex environments. Notably, innovative approaches that eliminate the need for gradient calculations can achieve up to 5.7 times memory reduction compared to traditional methods, making LLMs more accessible and practical for real-world deployment across diverse tasks and domains [27, 7, 37, 79, 80].

6 Challenges and Future Directions

6.1 Challenges in Scalability and Generalization

Parameter-efficient tuning (PET) techniques face notable challenges in scalability and generalization, affecting their application across diverse tasks and architectures. A key issue is the dependency on pretraining dataset characteristics, which limits PET methods' adaptability to new data [49]. Gradient-based tuning methods, including derivative-free optimization, often lead to slow convergence and instability, particularly in few-shot learning, impeding efficient scaling [52]. Additionally, the high computational costs of fine-tuning large language models (LLMs) vary significantly with architecture and tasks, underscoring the need for resource-efficient approaches that maintain performance while reducing computational demands [81].

Existing methods like Multi-Query Attention and Grouped Query Attention face challenges with optimal query head grouping, potentially leading to accuracy loss and affecting model generalization [82]. Benchmarks that fail to encompass all adversarial scenarios constrain the robustness and applicability of PET techniques, highlighting the need for comprehensive evaluation frameworks [61]. Hardware dependencies, such as Smart-Infinity's reliance on specific configurations, present additional scalability challenges [51]. Dynamic adjustments of tuning parameters and pruning masks, as seen in methods like APT, also complicate convergence, impacting both scalability and generalization [1].

The computational overhead and latency introduced by current PET methods can compromise accuracy, further complicating effective scaling [5]. Moreover, the inability to apply certain methods to LLMs with unavailable weights and potential limitations with different architectures restrict the generalization of PET techniques [7]. Addressing these challenges requires developing scalable, generalizable, and resource-efficient tuning techniques capable of dynamically adjusting to varying task requirements while ensuring stable training, rapid convergence, and robust performance. Recent advancements, such as Res-Tuning, which decouples tuners from backbone architecture, exemplify innovative approaches that enhance efficiency across multiple tasks [3, 54, 83].

6.2 Future Directions and Research Opportunities

The future of parameter-efficient tuning (PET) in large language models (LLMs) presents numerous research opportunities aimed at enhancing adaptability, scalability, and efficiency across diverse tasks and domains. Developing a theoretical framework for Pre-trained Visual Parameter-efficient (PVP) tuning could facilitate broader application across tasks and datasets [10]. The unified framework of

AI-generated, for reference only.

plugin-tuning offers potential exploration in tasks like text matching, requiring specific label spaces [8]. Further research could focus on more efficient PEFT techniques and their application to emerging domains, necessitating comprehensive benchmarks to evaluate performance [9].

Integrating PET methods with reinforcement learning could significantly improve model alignment with human preferences, expanding applicability across diverse tasks, particularly in large pre-trained language models (PLMs), where traditional full fine-tuning is often impractical. By leveraging modular modifications and innovative frameworks like Vision-and-Language Parameter-Efficient Tuning (VL-PET) and Hierarchical Decomposition PET (HiDe-PET), researchers can achieve competitive performance while optimizing fewer parameters [3, 39, 4]. Exploring the influence of data distribution on scaling laws and investigating additional model architectures and configurations could provide deeper insights into LLM scalability, guiding future PET developments. Integrating Adaptive Pruning and Tuning (APT) with other parameter-efficient fine-tuning architectures could significantly enhance training and inference efficiency [1, 54, 20, 84, 48].

In federated learning, developing PET methods that effectively handle larger models and diverse data distributions remains a critical research focus, ensuring LLMs can be deployed in decentralized environments while maintaining performance and data privacy. Optimizing current PET methods and exploring their adaptability across various model architectures and tasks could significantly enhance their effectiveness and versatility [85, 3, 4]. Enhancing compatibility among diverse language model architectures and investigating additional optimization techniques for proposed methods is crucial, as these improvements can lead to more effective fine-tuning strategies that leverage the strengths of large pre-trained language models (PLMs) while minimizing resource requirements [85, 3, 4, 86]. Extending current methods to other model architectures and developing a unified framework for parameter-efficient learning approaches could provide comprehensive solutions for optimizing LLMs in diverse applications.

The proposed future directions in PET are poised to enhance the field by offering scalable, adaptable, and efficient methodologies for deploying large-scale language models. These advancements will streamline training and fine-tuning processes across diverse applications while leveraging innovative techniques such as Arbitrary PET (APET) and Vision-and-Language Parameter-Efficient Tuning (VL-PET) to optimize performance while minimizing resource requirements [3, 4, 86].

6.3 Scalability and Computational Efficiency

Scalability and computational efficiency are critical in developing and deploying parameter-efficient tuning (PET) techniques for large language models (LLMs). The GPS framework exemplifies advancements in this area by offering a parameter-free and efficient solution for prompt optimization, significantly enhancing performance while reducing computational costs [87]. However, challenges persist, particularly in scenarios involving large task schemas or data volumes, as highlighted by the limitations of the ConPET method [88]. Yang et al. demonstrate a significant reduction in training time by avoiding complex calculations within the pretrained language model, enhancing computational efficiency [89]. Despite advancements, some methods, such as those proposed by Sadraeijavaeri et al., require approximately 1000 GPU hours for training, indicating substantial computational costs associated with their implementation [90]. Similarly, the SparseStru method demands considerable computational resources during the search process, potentially limiting its accessibility for certain applications [91].

The benchmark introduced by Chen et al. provides a rigorous evaluation protocol that separates validation and test sets, addressing flaws in previous studies and offering a more accurate comparison of PET methods [20]. To achieve scalability and computational efficiency in PET for large pre-trained language models (PLMs), ongoing innovation in techniques minimizing computational overhead is essential. This includes exploring diverse PET methods, such as Arbitrary PET (APET) and Hierarchical Decomposition PET (HiDe-PET), which optimize a reduced number of tunable parameters while maintaining or enhancing model performance across various tasks [39, 92, 86, 3, 11]. By addressing these challenges, PET techniques can be more widely adopted and effectively deployed across diverse applications and environments.

6.4 Generalization and Robustness

The generalization and robustness of parameter-efficient tuning techniques are critical in determining their effectiveness across diverse applications and environments. MoLA (Multi-layer Optimization with Layer-wise Allocation) exemplifies advancements in this area by optimizing expert allocation based on specific layer requirements, enhancing performance and reducing redundancy compared to traditional methods [93]. Understanding the generalization and robustness of safety mechanisms in large language models (LLMs) is particularly important in safety-critical regions, where the potential impact of model failures can be significant [94]. The study of scaling laws provides valuable insights into LLM behavior, yet may face limitations in generalizability due to fixed datasets and limited variations in model sizes [17].

Improving the effectiveness of tuning techniques for LLMs requires accounting for various operational contexts in which these models are utilized, as different tuning strategies, such as full-parameter and parameter-efficient approaches, can significantly influence performance based on specific deployment requirements and constraints [54, 29, 31, 95]. Addressing variations in data distributions, model architectures, and application domains enables parameter-efficient tuning methods to achieve greater generalization and robustness, ensuring effectiveness in real-world scenarios and contributing to broader LLM deployment across various fields.

6.5 Integration with Existing Architectures

The integration of parameter-efficient tuning (PET) techniques with existing architectures is pivotal for enhancing the adaptability and performance of large language models (LLMs) across diverse applications. Recent advancements have demonstrated the potential of incorporating PET methods into established model frameworks, optimizing resource utilization and improving model efficiency. For instance, integrating low-rank adaptation (LoRA) with derivative-free optimization techniques offers a novel approach to parameter-efficient tuning by reducing reliance on computationally intensive gradient-based updates [52]. The OpenDelta framework provides a versatile platform for integrating PET techniques with pre-trained models (PTMs), allowing for dynamic rerouting of tensors through delta modules [13].

The plugin-tuning framework further illustrates the potential for integrating PET techniques with existing architectures by consolidating insights from various classification tasks into a unified approach [8]. Additionally, integrating adaptive pruning and tuning (APT) with other parameter-efficient fine-tuning architectures offers a unified framework for various learning approaches, further enhancing the efficiency and scalability of PET methods [1]. The integration of PET techniques with existing architectures represents a significant advancement in the field, offering scalable and adaptable solutions for optimizing LLMs. By integrating various tuning methodologies, including full-parameter fine-tuning and parameter-efficient approaches, these enhancements significantly improve the flexibility and robustness of LLMs. This multifaceted strategy optimizes LLM performance in domain-specific tasks, as evidenced by specialized models like Med42 for medical applications, while facilitating knowledge transfer between server-side LLMs and client-side Small Language Models (SLMs) through frameworks like FedCoLLM [29, 60, 70, 31].

6.6 Methodological Innovations

Recent methodological innovations in parameter-efficient tuning (PET) have significantly advanced the field, offering novel solutions to enhance the adaptability and efficiency of large language models (LLMs). One notable innovation is the PET-GC approach, which adapts modules from smaller language models to larger ones without relying on gradient-based optimization, thereby reducing computational demands and enhancing model adaptability [7]. The integration of sparse architectures, such as SiRA, has also contributed to methodological advancements in PET by leveraging a sparse mixture of experts to enhance both resource utilization and model performance [38].

The development of the PVP framework, which pre-trains parameter-efficient tuning modules on extensive datasets before applying them to downstream tasks, represents a significant innovation in improving performance in low-data scenarios [10]. The introduction of Adaptive Pruning and Tuning (APT) exemplifies methodological innovations in PET by employing a gate mechanism that allows for fine-grained token-level and coarse-grained layer-level adjustments, contrasting with previous methods that relied on fixed-length prefixes [1]. Additionally, the development of the OpenDelta framework, which enables flexible adaptation of pre-trained models through plug-and-play delta tuning methods, illustrates the potential for dynamic rerouting of tensors to enhance model efficiency and adaptability [13].

Collectively, these methodological innovations in parameter-efficient tuning represent significant advancements in the field, offering scalable and adaptable solutions for optimizing LLMs across a wide range of applications. As the field of large language models (LLMs) advances, recent innovations such as integrating optimization algorithms, selective prediction frameworks, evolving knowledge distillation, and targeted continual training strategies promise to enhance performance and reliability significantly. These advancements improve LLM decision-making capabilities in dynamic environments by refining architecture and output quality and facilitate application in high-stakes scenarios through improved self-evaluation and selective prediction [96, 60, 97, 95].

6.7 Addressing Bias and Ethical Considerations

Bias and ethical considerations are paramount in developing and deploying parameter-efficient tuning (PET) techniques, as these methods significantly influence the behavior and output of large language models (LLMs). The potential for generating harmful or biased content necessitates careful modulation and evaluation of these models to mitigate such risks. Ethical considerations are particularly crucial when employing techniques like cross-lingual prompt tuning, which can enhance energy efficiency and reduce the carbon footprint associated with model training, thereby aligning with broader environmental and ethical goals [56]. The integration of ethical guidelines into the PET process is exemplified by frameworks such as OpenDelta, which emphasizes transparency and thorough review during development [13].

Effective dataset management is crucial for mitigating misinformation, as reliance on inaccurate factual information can raise serious ethical issues. This is particularly important in deploying LLMs, where careful consideration of data quality directly impacts model generalization and performance. Research indicates that selectively retaining informative samples during training enhances model refinement, while using incorrect or noisy data can lead to significant errors and ethical dilemmas [54, 95, 69]. Ensuring that data augmentation techniques do not introduce biases, especially in sensitive domains, is critical to maintaining the integrity and reliability of LLMs. Ethical considerations must be integrated into the design and evaluation of PET methods to ensure that these models are not only efficient but also aligned with societal values and ethical norms.

The integration of ethical guidelines into the development and deployment of Parameter-Efficient Tuning (PET) techniques is crucial, as it ensures that these advanced methodologies—such as the VL-PET framework and HiDe-PET—are not only efficient and effective but also responsible and aligned with broader societal values. This integration addresses potential issues such as performance degradation from excessive modular modifications and the need for generalization, evaluation, and cost-optimal deployment in machine learning applications, ultimately fostering trust and accountability in using these technologies [39, 4, 69]. By addressing potential biases and ensuring transparency, researchers can develop LLMs that are reliable, transparent, and free from bias, ultimately contributing to the responsible use of artificial intelligence in various applications.

7 Conclusion

This survey delves into parameter-efficient tuning (PET) methods, underscoring their pivotal role in enhancing the efficiency and performance of large language models (LLMs). Techniques like HyperPELT exemplify the ability to achieve superior performance with a reduced number of trainable parameters, particularly excelling in multi-task and few-shot learning environments. Similarly, the IAA method demonstrates significant progress by tuning a minimal fraction of model parameters while attaining results comparable to comprehensive fine-tuning.

Federated co-tuning approaches, such as FedCoLLM, highlight the dual benefits of improving both LLMs and smaller language models (SLMs), all while safeguarding data privacy and reducing resource consumption. ENGINE further exemplifies the importance of efficient tuning strategies by achieving optimal model performance with minimal training expenditure.

In the realm of clinical NLP tasks, a two-step PEFT framework showcases notable improvements in AUROC scores, emphasizing PET's impact on specialized domain performance. The utilization of

AI-generated, for reference only.

non-instructional data enhances instruction-following capabilities in LLMs, outperforming traditional instruction-tuned models.

Plugin-tuning emerges as a promising alternative to standard fine-tuning, achieving comparable performance with significantly fewer parameters, thus offering an efficient approach for classification tasks. Collectively, these insights underscore the transformative potential of PET techniques in optimizing LLMs, broadening their applicability across various fields while ensuring high performance and resource efficiency.

References

[1] Bowen Zhao, Hannaneh Hajishirzi, and Qingqing Cao. Apt: Adaptive pruning and tuning pre-trained language models for efficient training and inference. *arXiv preprint arXiv:2401.12200*, 2024.

[2] Zhuo Zhang, Yuanhang Yang, Yong Dai, Lizhen Qu, and Zenglin Xu. When federated learning meets pre-trained language models' parameter-efficient tuning methods. *arXiv preprint arXiv:2212.10025*, 2022.

[3] Yusheng Su, Chi-Min Chan, Jiali Cheng, Yujia Qin, Yankai Lin, Shengding Hu, Zonghan Yang, Ning Ding, Xingzhi Sun, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Exploring the impact of model scaling on parameter-efficient tuning, 2023.

[4] Zi-Yuan Hu, Yanyang Li, Michael R. Lyu, and Liwei Wang. Vl-pet: Vision-and-language parameter-efficient tuning via granularity control, 2023.

[5] Ali Edalati, Marzieh Tahaei, Ivan Kobyzev, Vahid Partovi Nia, James J. Clark, and Mehdi Rezagholizadeh. Krona: Parameter efficient tuning with kronecker adapter, 2022.

[6] Zhen-Ru Zhang, Chuanqi Tan, Haiyang Xu, Chengyu Wang, Jun Huang, and Songfang Huang. Towards adaptive prefix tuning for parameter-efficient language model fine-tuning. *arXiv preprint arXiv:2305.15212*, 2023.

[7] Feihu Jin, Jiajun Zhang, and Chengqing Zong. Parameter-efficient tuning for large language model without calculating its gradients. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 321–330, 2023.

[8] king parameter-efficient tuning.

[9] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

[10] Zhao Song, Ke Yang, Naiyang Guan, Junjie Zhu, Peng Qiao, and Qingyong Hu. Pvp: Pre-trained visual parameter-efficient tuning, 2023.

[11] Tianyu Ding, Tianyi Chen, Haidong Zhu, Jiachen Jiang, Yiqi Zhong, Jinxin Zhou, Guangzhi Wang, Zhihui Zhu, Ilya Zharkov, and Luming Liang. The efficiency spectrum of large language models: An algorithmic survey. *arXiv preprint arXiv:2312.00678*, 2023.

[12] Maximilian Mozes, Tolga Bolukbasi, Ann Yuan, Frederick Liu, Nithum Thain, and Lucas Dixon. Gradient-based automated iterative recovery for parameter-efficient tuning, 2023.

[13] Shengding Hu, Ning Ding, Weilin Zhao, Xingtai Lv, Zhen Zhang, Zhiyuan Liu, and Maosong Sun. Opendelta: A plug-and-play library for parameter-efficient adaptation of pre-trained models, 2023.

[14] Juncheng Xie, Shensian Syu, and Hung yi Lee. Non-instructional fine-tuning: Enabling instruction-following capabilities in pre-trained language models without instruction-following data, 2024.

[15] Jonathan Li, Will Aitken, Rohan Bhambhoria, and Xiaodan Zhu. Prefix propagation: Parameter-efficient tuning for long sequences, 2023.

[16] Tianci Xue, Ziqi Wang, and Heng Ji. Parameter-efficient tuning helps language model alignment. *arXiv preprint arXiv:2310.00819*, 2023.

[17] Xuyang Shen, Dong Li, Ruitao Leng, Zhen Qin, Weigao Sun, and Yiran Zhong. Scaling laws for linear complexity language models, 2024.

[18] Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Ruoyu Sun, and Zhi-Quan Luo. Entropic distribution matching in supervised fine-tuning of llms: Less overfitting and better diversity, 2024.

[19] Guanzheng Chen, Fangyu Liu, Zaiqiao Meng, and Shangsong Liang. Revisiting parameter-efficient tuning: Are we really there yet? *arXiv preprint arXiv:2202.07962*, 2022.

[20] Guanzheng Chen, Fangyu Liu, Zaiqiao Meng, and Shangsong Liang. Revisiting parameter-efficient tuning: Are we really there yet?, 2022.

[21] Yitao Liu, Chenxin An, and Xipeng Qiu. $\mathcal{Y}$-tuning: An efficient tuning paradigm for large-scale pre-trained models via label representation learning, 2023.

[22] Qi Zhu, Da Zheng, Xiang Song, Shichang Zhang, Bowen Jin, Yizhou Sun, and George Karypis. Parameter-efficient tuning large language models for graph representation learning, 2024.

[23] Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D. Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes, 2024.

[24] Yihang Zhai, Haixin Wang, Jianlong Chang, Xinlong Yang, Jinan Sun, Shikun Zhang, and Qi Tian. When parameter-efficient tuning meets general-purpose vision-language models, 2023.

[25] Venkat Srinivasan, Darshan Gandhi, Urmish Thakker, and Raghu Prabhakar. Training large language models efficiently with sparsity and dataflow, 2023.

[26] Yun Zhu, Yaoke Wang, Haizhou Shi, and Siliang Tang. Efficient tuning and inference for large language models on textual graphs. *arXiv preprint arXiv:2401.15569*, 2024.

[27] Pei-Fu Guo, Ying-Hsuan Chen, Yun-Da Tsai, and Shou-De Lin. Towards optimizing with large language models, 2024.

[28] Kecheng Zheng, Wei Wu, Ruili Feng, Kai Zhu, Jiawei Liu, Deli Zhao, Zheng-Jun Zha, Wei Chen, and Yujun Shen. Regularized mask tuning: Uncovering hidden knowledge in pre-trained vision-language models, 2023.

[29] Katrin Tomanek, Shanqing Cai, and Subhashini Venugopalan. Parameter efficient tuning allows scalable personalization of llms for text entry: A case study on abbreviation expansion, 2023.

[30] Huanqian Wang, Yang Yue, Rui Lu, Jingxin Shi, Andrew Zhao, Shenzhi Wang, Shiji Song, and Gao Huang. Model surgery: Modulating llm's behavior via simple parameter editing, 2025.

[31] Clément Christophe, Praveen K Kanithi, Prateek Munjal, Tathagata Raha, Nasir Hayat, Ronnie Rajan, Ahmed Al-Mahrooqi, Avani Gupta, Muhammad Umar Salman, Gurpreet Gosal, Bhargav Kanakiya, Charles Chen, Natalia Vassilieva, Boulbaba Ben Amor, Marco AF Pimentel, and Shadab Khan. Med42 – evaluating fine-tuning strategies for medical llms: Full-parameter vs. parameter-efficient approaches, 2024.

[32] Guochang Li, Chen Zhi, Jialiang Chen, Junxiao Han, and Shuiguang Deng. Exploring parameter-efficient fine-tuning of large language model on automated program repair. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering*, pages 719–731, 2024.

[33] Tianci Xue, Ziqi Wang, and Heng Ji. Parameter-efficient tuning helps language model alignment, 2023.

[34] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[35] Xiangyang Liu, Tianxiang Sun, Xuanjing Huang, and Xipeng Qiu. Late prompt tuning: A late prompt could be better than many prompts, 2022.

[36] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning, 2022.

[37] Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jia-Wei Low, Lidong Bing, and Luo Si. On the effectiveness of adapter-based tuning for pretrained language model adaptation. *arXiv preprint arXiv:2106.03164*, 2021.

[38] Yun Zhu, Nevan Wichers, Chu-Cheng Lin, Xinyi Wang, Tianlong Chen, Lei Shu, Han Lu, Canoee Liu, Liangchen Luo, Jindong Chen, and Lei Meng. Sira: Sparse mixture of low rank adaptation, 2023.

[39] Liyuan Wang, Jingyi Xie, Xingxing Zhang, Hang Su, and Jun Zhu. Hide-pet: Continual learning via hierarchical decomposition of parameter-efficient tuning, 2024.

[40] Wenhan Xia, Chengwei Qin, and Elad Hazan. Chain of lora: Efficient fine-tuning of language models via residual learning. *arXiv preprint arXiv:2401.04151*, 2024.

[41] Kai Yao, Penglei Gao, Lichun Li, Yuan Zhao, Xiaofeng Wang, Wei Wang, and Jianke Zhu. Layer-wise importance matters: Less memory for better performance in parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2410.11772*, 2024.

[42] Mengzhao Chen, Wenqi Shao, Peng Xu, Jiahao Wang, Peng Gao, Kaipeng Zhang, and Ping Luo. Efficientqat: Efficient quantization-aware training for large language models, 2024.

[43] Oliver Sieberling, Denis Kuznedelev, Eldar Kurtic, and Dan Alistarh. Evopress: Towards optimal dynamic model compression via evolutionary search, 2024.

[44] Zihan Fang, Zheng Lin, Zhe Chen, Xianhao Chen, Yue Gao, and Yuguang Fang. Automated federated pipeline for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2404.06448*, 2024.

[45] Chun-Wei Ho, Chao-Han Huck Yang, and Sabato Marco Siniscalchi. Differentially private adapters for parameter efficient acoustic modeling, 2023.

[46] Se Jung Kwon, Jeonghoon Kim, Jeongin Bae, Kang Min Yoo, Jin-Hwa Kim, Baeseong Park, Byeongwook Kim, Jung-Woo Ha, Nako Sung, and Dongsoo Lee. Alphatuning: Quantization-aware parameter-efficient adaptation of large-scale pre-trained language models. *arXiv preprint arXiv:2210.03858*, 2022.

[47] Memory-efficient fine-tuning of.

[48] Mingyang Zhang, Hao Chen, Chunhua Shen, Zhen Yang, Linlin Ou, Xinyi Yu, and Bohan Zhuang. Loraprune: Structured pruning meets low-rank parameter-efficient fine-tuning, 2024.

[49] Rana Muhammad Shahroz Khan, Pingzhi Li, Sukwon Yun, Zhenyu Wang, Shahriar Nirjon, Chau-Wai Wong, and Tianlong Chen. Portllm: Personalizing evolving large language models with training-free and portable model patches, 2024.

[50] Nicolas Wagner, Dongyang Fan, and Martin Jaggi. Personalized collaborative fine-tuning for on-device large language models, 2024.

[51] Hongsun Jang, Jaeyong Song, Jaewon Jung, Jaeyoung Park, Youngsok Kim, and Jinho Lee. Smart-infinity: Fast large language model training using near-storage processing on a real system, 2024.

[52] Feihu Jin, Yin Liu, and Ying Tan. Derivative-free optimization for low-rank adaptation in large language models, 2024.

[53] Ning Ding, Xingtai Lv, Qiaosen Wang, Yulin Chen, Bowen Zhou, Zhiyuan Liu, and Maosong Sun. Sparse low-rank adaptation of pre-trained language models, 2023.

[54] Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, and Xueqi Cheng. Scattered or connected? an optimized parameter-efficient tuning approach for information retrieval, 2022.

[55] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. Dylora: Parameter efficient tuning of pre-trained models using dynamic search-free low-rank adaptation, 2023.

[56] Nohil Park, Joonsuk Park, Kang Min Yoo, and Sungroh Yoon. On the analysis of cross-lingual prompt tuning for decoder-based multilingual model, 2023.

[57] Ju ho Kim, Jungwoo Heo, Hyun seo Shin, Chan yeong Lim, and Ha-Jin Yu. Integrated parameter-efficient tuning for general-purpose audio models, 2023.

[58] Zihao Fu, Haoran Yang, Anthony Man-Cho So, Wai Lam, Lidong Bing, and Nigel Collier. On the effectiveness of parameter-efficient fine-tuning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 12799–12807, 2023.

[59] Chendong Xiang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. A closer look at parameter-efficient tuning in diffusion models, 2023.

[60] Sen Huang, Kaixiang Yang, Sheng Qi, and Rui Wang. When large language model meets optimization, 2024.

[61] Jiacheng Ruan, Xian Gao, Suncheng Xiang, Mingye Xie, Ting Liu, and Yuzhuo Fu. Understanding robustness of parameter-efficient tuning for image classification, 2024.

[62] Arjun Singh, Nikhil Pandey, Anup Shirgaonkar, Pavan Manoj, and Vijay Aski. A study of optimizations for fine-tuning large language models. *arXiv preprint arXiv:2406.02290*, 2024.

[63] Jason Phang, Yi Mao, Pengcheng He, and Weizhu Chen. Hypertuning: Toward adapting large language models without back-propagation. In *International Conference on Machine Learning*, pages 27854–27875. PMLR, 2023.

[64] Niki van Stein, Diederick Vermetten, and Thomas Bäck. In-the-loop hyper-parameter optimization for llm-based automated design of heuristics, 2024.

[65] Youngsuk Park, Kailash Budhathoki, Liangfu Chen, Jonas Kübler, Jiaji Huang, Matthäus Kleindessner, Jun Huan, Volkan Cevher, Yida Wang, and George Karypis. Inference optimization of foundation models on ai accelerators, 2024.

[66] Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts.

[67] Wangchunshu Zhou, Canwen Xu, and Julian McAuley. Efficiently tuned parameters are task embeddings, 2022.

[68] Fernandes Junior and Ricardo Corso. Improving model performance: comparing complete fine-tuning with parameter efficient language model tuning on a small, portuguese, domain-specific, dataset. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2022.

[69] Abi Aryan, Aakash Kumar Nain, Andrew McMahon, Lucas Augusto Meyer, and Harpreet Singh Sahota. The costly dilemma: Generalization, evaluation and cost-optimal deployment of large language models, 2023.

[70] Tao Fan, Yan Kang, Guoqiang Ma, Lixin Fan, Kai Chen, and Qiang Yang. Fedcollm: A parameter-efficient federated co-tuning framework for large and small language models. *arXiv preprint arXiv:2411.11707*, 2024.

[71] David Grangier, Angelos Katharopoulos, Pierre Ablin, and Awni Hannun. Need a small specialized language model? plan early!, 2024.

[72] Aleksandra Edwards and Jose Camacho-Collados. Language models for text classification: Is in-context learning enough?, 2024.

[73] Stephen Obadinma, Hongyu Guo, and Xiaodan Zhu. Effectiveness of data augmentation for parameter efficient tuning with limited data, 2023.

[74] Zhuo Zhang, Yuanhang Yang, Yong Dai, Lizhen Qu, and Zenglin Xu. When federated learning meets pre-trained language models' parameter-efficient tuning methods, 2023.

[75] Alireza Salemi and Hamed Zamani. Comparing retrieval-augmentation and parameter-efficient fine-tuning for privacy-preserving personalization of large language models. *arXiv preprint arXiv:2409.09510*, 2024.

[76] Ruoyu Wang, Xiaoxuan Li, and Lina Yao. Deconfounded causality-aware parameter-efficient fine-tuning for problem-solving improvement of llms, 2024.

[77] Zhiyuan Peng, Xuyang Wu, Qifan Wang, Sravanthi Rajanala, and Yi Fang. Q-peft: Query-dependent parameter efficient fine-tuning for text reranking with large language models. *arXiv preprint arXiv:2404.04522*, 2024.

[78] Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, Jimmy Ba, and Amjad Almahairi. Residual prompt tuning: Improving prompt tuning with residual reparameterization, 2023.

[79] Zhaojiang Lin, Andrea Madotto, and Pascale Fung. Exploring versatile generative language model via parameter-efficient transfer learning. *arXiv preprint arXiv:2004.03829*, 2020.

[80] Chengyu Wang, Junbing Yan, Wei Zhang, and Jun Huang. Towards better parameter-efficient fine-tuning for large language models: A position paper. *arXiv preprint arXiv:2311.13126*, 2023.

[81] Evani Radiya-Dixit and Xin Wang. How fine can fine-tuning be? learning efficient language models. In *International Conference on Artificial Intelligence and Statistics*, pages 2435–2443. PMLR, 2020.

[82] Vinay Joshi, Prashant Laddha, Shambhavi Sinha, Om Ji Omer, and Sreenivas Subramoney. Qcqa: Quality and capacity-aware grouped query attention, 2024.

[83] Zeyinzi Jiang, Chaojie Mao, Ziyuan Huang, Ao Ma, Yiliang Lv, Yujun Shen, Deli Zhao, and Jingren Zhou. Res-tuning: A flexible and efficient tuning paradigm via unbinding tuner from backbone, 2023.

[84] Zhuoyi Yang, Ming Ding, Yanhui Guo, Qingsong Lv, and Jie Tang. Parameter-efficient tuning makes a good classification head, 2023.

[85] Neal Lawton, Anoop Kumar, Govind Thattai, Aram Galstyan, and Greg Ver Steeg. Neural architecture search for parameter-efficient fine-tuning of large pre-trained language models, 2023.

[86] Neal Lawton, Anoop Kumar, Govind Thattai, Aram Galstyan, and Greg Ver Steeg. Neural architecture search for parameter-efficient fine-tuning of large pre-trained language models. *arXiv preprint arXiv:2305.16597*, 2023.

[87] Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. Gps: Genetic prompt search for efficient few-shot learning, 2022.

[88] Chenyang Song, Xu Han, Zheni Zeng, Kuai Li, Chen Chen, Zhiyuan Liu, Maosong Sun, and Tao Yang. Conpet: Continual parameter-efficient tuning for large language models. *arXiv preprint arXiv:2309.14763*, 2023.

[89] Haoran Yang, Piji Li, and Wai Lam. Parameter-efficient tuning by manipulating hidden states of pretrained language models for classification tasks, 2022.

[90] MohammadAli SadraeiJavaeri, Ehsaneddin Asgari, Alice Carolyn McHardy, and Hamid Reza Rabiee. Superpos-prompt: Enhancing soft prompt tuning of language models with superposition of multi token embeddings, 2024.

[91] Sparse structure search for parameter-efficient tuning.

[92] Qiankun Gao, Chen Zhao, Yifan Sun, Teng Xi, Gang Zhang, Bernard Ghanem, and Jian Zhang. A unified continual learning framework with general parameter-efficient tuning, 2023.

[93] Chongyang Gao, Kezhen Chen, Jinmeng Rao, Baochen Sun, Ruibo Liu, Daiyi Peng, Yawen Zhang, Xiaoyuan Guo, Jie Yang, and VS Subrahmanian. Higher layers need more lora experts, 2024.

[94] Boyi Wei, Kaixuan Huang, Yangsibo Huang, Tinghao Xie, Xiangyu Qi, Mengzhou Xia, Prateek Mittal, Mengdi Wang, and Peter Henderson. Assessing the brittleness of safety alignment via pruning and low-rank modifications, 2024.

AI-generated, for reference only.

[95] Xuxi Chen, Zhendong Wang, Daouda Sow, Junjie Yang, Tianlong Chen, Yingbin Liang, Mingyuan Zhou, and Zhangyang Wang. Take the bull by the horns: Hard sample-reweighted continual training improves llm generalization, 2024.

[96] Chengyuan Liu, Yangyang Kang, Fubang Zhao, Kun Kuang, Zhuoren Jiang, Changlong Sun, and Fei Wu. Evolving knowledge distillation with large language models and active learning, 2024.

[97] Jiefeng Chen, Jinsung Yoon, Sayna Ebrahimi, Sercan O Arik, Tomas Pfister, and Somesh Jha. Adaptation with self-evaluation to improve selective prediction in llms, 2023.

Disclaimer:

SurveyX is an AI-powered system designed to automate the generation of surveys. While it aims to produce high-quality, coherent, and comprehensive surveys with accurate citations, the final output is derived from the AI's synthesis of pre-processed materials, which may contain limitations or inaccuracies. As such, the generated content should not be used for academic publication or formal submissions and must be independently reviewed and verified. The developers of SurveyX do not assume responsibility for any errors or consequences arising from the use of the generated surveys.