Review

Not peer-reviewed version

Low-Rank Adaptation for Scalable Fine-Tuning of Pre-Trained Language Models

Haoyu Dong and Jianhong Shun [*]

Posted Date: 11 February 2025

doi: 10.20944/preprints202502.0762.v1

Keywords: Low-Rank Adaptation (LoRA); fine-tuning; large language models; computational efficiency; task adaptation; domain adaptation; few-shot learning; transfer learning; zero-shot learning; model efficiency; NLP; machine learning

Review

Low-Rank Adaptation for Scalable Fine-Tuning of Pre-Trained Language Models

Haoyu Dong and Jianhong Shun *

The Frontier Institute of Science and Technology (FIST), Xi'an Jiaotong University, China
* Correspondence: jianhongshun@mail.xjtu.edu.cn

Abstract: Low-Rank Adaptation (LoRA) is a computationally efficient approach for fine-tuning large pre-trained language models, designed to reduce memory and computational overhead by introducing low-rank matrices into the model's weight updates. This survey provides a comprehensive overview of LoRA, including its theoretical foundations, applications, and the advantages it offers over traditional fine-tuning methods. We explore how LoRA enables efficient task adaptation in scenarios such as domain adaptation, few-shot learning, transfer learning, and zero-shot learning. Additionally, we examine its challenges, such as rank selection, generalization to complex tasks, and risks of overfitting, while identifying key areas for future research, including adaptive rank selection, integration with other fine-tuning techniques, and multi-modal and cross-domain adaptation. LoRA's potential to make large-scale models more adaptable and efficient is significant for advancing natural language processing (NLP) and machine learning applications, especially when computational resources are limited. This survey aims to highlight the current state of LoRA, its practical applications, and the ongoing research opportunities to further enhance its capabilities.

Keywords: Low-Rank Adaptation (LoRA); fine-tuning; large language models; computational efficiency; task adaptation; domain adaptation; few-shot learning; transfer learning; zero-shot learning, model efficiency; NLP; machine learning

1. Introduction

In recent years, large language models (LLMs) have revolutionized the field of natural language processing (NLP), achieving significant advancements across a broad range of applications such as text generation, machine translation, summarization, sentiment analysis, and question answering. These models, exemplified by architectures like GPT-3 , BERT , and T5 , have demonstrated impressive capabilities, generating human-like text and solving complex language tasks. However, the success of these models comes at a cost. As their size continues to grow, the computational requirements for training, fine-tuning, and deployment become increasingly prohibitive [1]. Training large language models from scratch requires immense computational resources, including specialized hardware such as Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs), as well as large-scale datasets for pre-training. The fine-tuning of these models for specific downstream tasks further exacerbates the resource consumption, making it challenging for smaller research labs and organizations to leverage these powerful models effectively [2]. As such, the problem of making large models more accessible and adaptable without excessive computational overhead has become a pressing concern. One promising solution to this problem is the use of low-rank adaptation (LoRA), a technique that offers an efficient alternative to traditional fine-tuning methods [3]. LoRA seeks to reduce the number of parameters that need to be updated during the fine-tuning process by leveraging low-rank matrix approximations. Instead of modifying the entire set of parameters in a pre-trained model, LoRA focuses on adapting a small number of parameters that have the most significant impact on model performance [4]. This enables the model to retain most of its original pre-trained knowledge while allowing it to be fine-tuned effectively with significantly fewer computational resources. The core idea behind LoRA is based on the observation that many large models have a significant number of redundant parameters that do not contribute meaningfully to task-specific performance [5]. By introducing low-rank matrices into the model architecture, LoRA approximates the updates needed for fine-tuning using a smaller subset of parameters. This reduces the memory footprint and computational requirements associated with fine-tuning, allowing for more efficient model adaptation. Furthermore, LoRA facilitates transfer learning by enabling the model to quickly adapt to new tasks or domains with minimal retraining. LoRA has garnered attention in both academia and industry for its ability to make large language models more efficient and accessible [6]. The technique has been shown to outperform traditional fine-tuning approaches in several key aspects, including computational cost, memory usage, and training time, while maintaining or even improving task-specific performance. Furthermore, LoRA is highly flexible and can be applied to a variety of pre-trained models, making it a versatile tool for fine-tuning in resource-constrained environments [7]. This survey aims to provide a comprehensive overview of the Low-Rank Adaptation (LoRA) technique and its applications to large language models. We explore the theoretical foundations of LoRA, its benefits and limitations, and its role in the broader context of model adaptation and transfer learning. The following sections present an in-depth examination of the key concepts and developments in the field, starting with the background and motivation for LoRA in Section 2 [8]. In Section 3, we review related work in the areas of model fine-tuning, low-rank approximation, and parameter-efficient transfer learning. We also highlight how LoRA compares to other existing methods, such as adapter networks and prompt-tuning , which aim to achieve similar goals of efficient model adaptation [9]. A comparative analysis of LoRA's advantages and drawbacks is provided, with a focus on its ability to balance efficiency and performance across different NLP tasks [10]. Section 4 explores the practical applications of LoRA in fine-tuning large language models. This includes an examination of its use in domain adaptation, where LoRA is employed to adapt pre-trained models to specialized domains with limited labeled data. We also investigate its role in few-shot learning scenarios, where LoRA enables models to learn new tasks from a minimal number of examples, making it particularly useful for scenarios where labeled data is scarce. Additionally, we explore how LoRA can be integrated into zero-shot learning frameworks, expanding the potential of LLMs in more dynamic and evolving environments. In Section 5, we identify the key challenges and limitations associated with the use of LoRA [11]. Although LoRA provides significant computational savings, it is not without its trade-offs [12]. These challenges include the choice of optimal rank for the low-rank approximations, the risk of overfitting in low-resource settings, and the difficulty of scaling LoRA to models with extremely large parameter spaces. We also discuss potential solutions and open research questions that could help address these issues, offering insight into the future of LoRA research. Finally, in Section 6, we discuss future directions in the development of LoRA and its integration with emerging techniques in the field of model compression and adaptation [13,14]. We consider potential improvements in the design of low-rank approximations, as well as the combination of LoRA with other cutting-edge approaches such as knowledge distillation and multi-task learning. The integration of LoRA with future innovations in hardware and model architectures will likely play a key role in enabling more efficient, scalable, and interpretable language models. The remainder of this paper is organized as follows. In Section 2, we provide a detailed background on the fundamental principles of LoRA and its relationship to other fine-tuning techniques. Section 3 discusses the existing literature on low-rank adaptation, parameter-efficient transfer learning, and related methods [15]. In Section 4, we present various practical applications of LoRA in large language models. Section 5 highlights the challenges and limitations of LoRA, while Section 6 offers an outlook on future developments in this area.

2. Background

In this section, we provide a detailed overview of the core concepts behind Low-Rank Adaptation (LoRA) and its relationship to traditional fine-tuning methods [16]. We first discuss the general framework of large language models (LLMs) and the challenges associated with their adaptation to specific tasks [17]. We then introduce the key principles of low-rank approximation, followed by a description of how LoRA integrates these principles into the model adaptation process [18]. Finally, we highlight the advantages of LoRA and its efficiency in comparison to traditional fine-tuning approaches [19].

2.1. Large Language Models

Large language models are a class of deep learning models designed to process and generate human language. These models, often based on architectures like transformers , are typically pre-trained on vast amounts of text data and then fine-tuned for specific downstream tasks [20]. The large number of parameters in LLMs enables them to capture rich linguistic patterns and semantic information from diverse corpora. However, this immense parameter space also brings about significant computational and memory demands, which make it challenging to deploy such models efficiently [21]. The success of LLMs stems from their ability to generalize well across various NLP tasks, including but not limited to text classification, named entity recognition, machine translation, and summarization [22]. However, to achieve high performance on specific tasks, fine-tuning is required to adapt the model to domain-specific knowledge or to optimize for particular performance metrics [23]. Unfortunately, fine-tuning large models often necessitates extensive computational resources, leading to challenges in scaling models and making them accessible to a broader set of users [24].

2.2. Challenges in Fine-Tuning Large Language Models

Traditional fine-tuning methods involve updating all or most of the parameters of a pre-trained model to better fit a target task. While this approach can lead to significant improvements in performance, it also incurs substantial costs. Fine-tuning typically requires large amounts of labeled data, high computational power, and considerable training time. As the size of the model increases, these challenges become even more pronounced [25]. Moreover, fine-tuning large models on small datasets can also lead to overfitting, as the model may memorize the limited examples rather than learning generalizable features [26]. Another challenge is the difficulty of adapting pre-trained models to new domains or tasks that differ significantly from the ones encountered during pre-training [27]. In such cases, the model may require substantial re-training or re-parameterization, further escalating the computational burden [28]. To address these challenges, several techniques have been proposed to reduce the cost of fine-tuning and make large language models more accessible. These techniques often focus on adapting only a small subset of model parameters or introducing mechanisms to make the adaptation process more efficient.

2.3. Low-Rank Approximation in Neural Networks

Low-rank approximation is a mathematical technique commonly used in linear algebra and optimization. The central idea is that many high-dimensional data structures, such as matrices, can be approximated using a smaller number of components without losing much information [29]. In the context of neural networks, low-rank approximation involves approximating weight matrices with low-rank factorization, thus reducing the number of parameters and computational complexity [30]. In a typical neural network, the weight matrices of layers can be seen as high-dimensional tensors. By decomposing these matrices into lower-dimensional factors, the number of parameters required to represent the network can be significantly reduced. This not only leads to a smaller memory footprint but also accelerates training and inference times [31]. Low-rank techniques have been explored in various areas of deep learning, including model compression, knowledge distillation, and transfer learning [32]. These methods aim to achieve a balance between computational efficiency and model performance, making them particularly useful for applications in resource-constrained environments.

2.4. Low-Rank Adaptation (LoRA)

Low-Rank Adaptation (LoRA) is a technique that combines the idea of low-rank approximation with the need for efficient fine-tuning of large pre-trained models [33]. LoRA introduces low-rank matrices into the model architecture, which are learned during the fine-tuning process, while the original parameters of the pre-trained model remain frozen. The key insight behind LoRA is that the low-rank matrices can capture task-specific information in a computationally efficient manner without the need to modify the entire set of model parameters. The low-rank matrices are typically introduced in the form of additional layers or modules within the existing architecture, where they serve as adaptors to the original model's output. These low-rank layers are learned during the fine-tuning phase, allowing the model to adapt to new tasks or domains while maintaining the efficiency of the original pre-trained weights [34]. In this way, LoRA minimizes the number of parameters that need to be updated, significantly reducing the memory and computational requirements compared to traditional fine-tuning methods [35]. Mathematically, the adaptation process in LoRA can be described as follows. Let W be the weight matrix of a layer in the pre-trained model, and let A and B represent two low-rank matrices. LoRA introduces the approximation:

$$W' = W + A \cdot B$$

Here, W' represents the adapted weight matrix, and A and B are the low-rank matrices that are learned during fine-tuning [36]. By controlling the rank of the matrices A and B, the number of parameters that need to be updated can be reduced, leading to a more efficient adaptation process. The rank of the low-rank matrices is a key hyperparameter that determines the trade-off between computational efficiency and model performance. A lower rank reduces the number of parameters, but if the rank is too low, the model may fail to capture important task-specific information [37]. Conversely, a higher rank increases the model's capacity but may lead to higher computational costs.

2.5. Advantages of LoRA

The primary advantage of LoRA lies in its ability to adapt large language models to new tasks with minimal computational overhead [38]. By updating only a small subset of parameters, LoRA significantly reduces the amount of computation required for fine-tuning, enabling more efficient model adaptation even in resource-constrained environments. This makes LoRA particularly attractive for tasks where computational resources are limited or where fine-tuning on small datasets is required. Another key benefit of LoRA is its ability to achieve strong performance with relatively few parameters [39]. By focusing on the most important aspects of task-specific adaptation, LoRA allows for efficient learning without overfitting to small datasets. This makes LoRA suitable for scenarios like domain adaptation and few-shot learning, where the amount of labeled data is limited [40]. Additionally, LoRA is highly flexible and can be applied to a wide range of pre-trained models, making it a versatile tool for fine-tuning large language models across different NLP tasks. It is also compatible with existing model architectures, meaning it can be integrated into many state-of-the-art models without requiring significant modifications.

2.6. Related Techniques

LoRA is part of a broader trend of research aimed at improving the efficiency of fine-tuning large language models [41]. Other techniques that aim to achieve similar goals include adapter networks , which introduce lightweight modules that are added to the pre-trained model, and prompt-tuning , which focuses on optimizing a small set of prompt parameters instead of the model weights themselves. Each of these techniques has its own strengths and trade-offs, and LoRA distinguishes itself by its use of low-rank matrix factorization, which provides a direct method for controlling the number of parameters that are updated during fine-tuning. In the following section, we explore these related methods in more detail and provide a comparative analysis of LoRA's advantages and limitations [42].

3. Related Work

In this section, we review the existing literature on parameter-efficient fine-tuning techniques and highlight the relationships between Low-Rank Adaptation (LoRA) and other relevant approaches. We focus on methods that aim to reduce the computational and memory costs associated with adapting large pre-trained models, such as adapter networks, prompt-tuning, and other low-rank approximation techniques. Additionally, we examine the key differences and similarities between LoRA and these approaches, providing insights into its unique contributions to the field [43].

3.1. Adapter Networks

Adapter networks are a class of methods designed to make fine-tuning of large pre-trained models more efficient by introducing lightweight modules, called adapters, between the layers of the model [44]. These adapters are small, task-specific neural networks that are trained while the original model weights remain frozen. The idea is to minimize the number of parameters that need to be trained while still enabling the model to adapt effectively to the target task. Adapter networks have been shown to be highly effective for domain adaptation and multi-task learning, as they allow for the reuse of pre-trained models across different tasks without the need to retrain the entire model [45]. While adapter networks reduce the number of parameters that need to be updated, they still introduce additional parameters that must be stored and trained [46]. In contrast, LoRA does not introduce entirely new parameters, but instead introduces low-rank matrices that approximate the updates to the original model's weight matrices. This results in a more efficient adaptation process with fewer additional parameters.

3.2. Prompt-Tuning

Prompt-tuning is another technique that aims to efficiently adapt large pre-trained models to new tasks [47]. In prompt-tuning, instead of modifying the model's weights, a small set of parameters, known as the prompt, is learned. The prompt is combined with the input text to guide the model's behavior during inference. This method is particularly useful in zero-shot or few-shot learning scenarios, where large amounts of labeled data may not be available [48]. Prompt-tuning has been shown to achieve competitive performance with minimal computation by learning a small number of parameters that modify the model's input representations. However, prompt-tuning is limited in that it may not capture as much task-specific information as methods that directly modify the model's weights, such as LoRA [49]. LoRA, by contrast, allows for more direct adaptation by modifying the internal representations of the model through low-rank matrices, offering a more powerful and flexible method for fine-tuning large models.

3.3. Low-Rank Approximation in Neural Networks

Low-rank approximation has a long history in machine learning, particularly in the context of model compression and efficient representation learning [50]. Several approaches have used low-rank matrix factorization to reduce the number of parameters in neural networks while maintaining model performance [51]. Techniques like singular value decomposition (SVD) and low-rank factorization have been applied to compress deep networks by approximating weight matrices with low-rank representations. LoRA is built upon this foundation of low-rank approximation. However, unlike traditional low-rank methods, which focus primarily on compressing models for inference, LoRA specifically targets efficient fine-tuning. By introducing low-rank matrices during fine-tuning, LoRA allows large pre-trained models to be adapted to new tasks with fewer parameters and reduced computational cost [52]. This distinguishes LoRA from standard low-rank techniques, which may not be optimized for the fine-tuning phase [53].

3.4. Parameter-Efficient Transfer Learning

A key motivation behind LoRA is the broader goal of parameter-efficient transfer learning. Transfer learning methods, such as fine-tuning, aim to transfer knowledge from a pre-trained model to a new task by adapting the model's parameters. However, traditional fine-tuning requires substantial resources, especially when dealing with very large models. Several recent approaches have sought to reduce the number of parameters that need to be trained during transfer learning. For example, the concept of using "few-shot adapters" has been explored, where only a small subset of the model's parameters are adapted to a new task. LoRA can be viewed as an extension of these ideas, as it introduces a more structured and efficient way to achieve parameter-efficient transfer learning by focusing on low-rank adaptations [54]. By controlling the rank of the adaptation matrices, LoRA offers a tunable trade-off between the efficiency of the adaptation process and the performance of the fine-tuned model.

3.5. Comparative Analysis of LoRA and Related Approaches

While LoRA shares similarities with adapter networks and prompt-tuning in its goal of reducing the number of parameters required for fine-tuning, it offers unique advantages. One of the main strengths of LoRA is its ability to adapt a pre-trained model by introducing low-rank matrices that approximate the updates to the model's weight matrices. This provides a more direct and efficient way of fine-tuning models compared to adapter networks, which require the addition of separate modules, or prompt-tuning, which limits the modifications to the input layer. In terms of computational efficiency, LoRA has been shown to outperform both adapter networks and prompt-tuning, especially when scaling to very large models. While adapter networks introduce additional task-specific parameters, LoRA keeps the number of parameters that need to be updated to a minimum, reducing both memory usage and training time. Furthermore, LoRA is flexible and can be applied to a wide range of pre-trained models, making it suitable for a variety of tasks, including domain adaptation, few-shot learning, and transfer learning [55]. In terms of performance, LoRA has been demonstrated to maintain or even improve task-specific accuracy compared to traditional fine-tuning methods, while requiring significantly fewer computational resources [56]. The low-rank structure of the adaptation matrices allows for efficient model updates, leading to faster convergence and reduced training times [57].

3.6. Summary

In this section, we have reviewed the key approaches in parameter-efficient fine-tuning and transfer learning, including adapter networks, prompt-tuning, and low-rank approximation methods. We have highlighted the similarities and differences between these methods and LoRA, emphasizing the unique advantages that LoRA offers in terms of computational efficiency and model performance [58]. The next section will explore the practical applications of LoRA, demonstrating how this technique can be applied to a variety of NLP tasks to achieve efficient and scalable model adaptation [59].

4. Applications of LoRA

In this section, we explore the various practical applications of Low-Rank Adaptation (LoRA) for fine-tuning large language models. The flexibility and efficiency of LoRA make it suitable for a wide range of tasks in natural language processing (NLP), especially when computational resources are constrained or when training data is limited. We discuss its use in domain adaptation, few-shot learning, and transfer learning, along with other emerging applications where LoRA can provide significant advantages. Additionally, we highlight some real-world case studies and research efforts that have successfully applied LoRA to different NLP challenges [60].

4.1. Domain Adaptation

Domain adaptation is a key challenge in NLP, where pre-trained models need to be adapted to new, often domain-specific, data distributions. For instance, a general-purpose language model trained on general web text may perform poorly when applied to legal, medical, or financial texts [61]. In such cases, domain adaptation techniques are used to fine-tune the model on a small set of labeled data from the target domain [62]. LoRA offers an efficient solution to domain adaptation by allowing the model to learn task-specific representations with a minimal number of parameters [63]. By introducing low-rank matrices into the model architecture, LoRA effectively adapts the model to the new domain without requiring full fine-tuning of the pre-trained model [64]. This is particularly valuable when domain-specific data is scarce, as LoRA can perform well with limited labeled examples, reducing the risk of overfitting. Several studies have demonstrated the effectiveness of LoRA in domain adaptation. For instance, in the case of adapting a general-purpose language model to the medical domain, LoRA allows for the efficient adaptation of the model to medical texts, achieving competitive performance while keeping the computational cost low. This makes LoRA an attractive option for applications in specialized fields, where obtaining large labeled datasets may be challenging [65].

4.2. Few-Shot Learning

Few-shot learning refers to the ability of a model to learn new tasks with a limited number of examples. In many real-world scenarios, collecting large amounts of labeled data is not feasible, making few-shot learning a valuable capability. LoRA can significantly improve few-shot learning by enabling large pre-trained models to adapt quickly to new tasks using only a small number of labeled examples [66]. LoRA's low-rank adaptation process allows the model to focus on the most relevant task-specific information, minimizing the number of parameters that need to be updated [67]. As a result, LoRA can achieve strong performance even in few-shot settings, where other fine-tuning methods might struggle. This is particularly useful for tasks where obtaining a large labeled dataset is costly or impractical. For example, in text classification tasks, LoRA has been applied to adapt large language models to specific categories with just a few labeled samples, showing that it can outperform traditional fine-tuning methods that require more data. This makes LoRA an effective tool for NLP tasks in domains such as sentiment analysis, intent classification, and customer support, where few-shot data may be available.

4.3. Transfer Learning

Transfer learning is a foundational concept in modern machine learning, where knowledge gained from one task or domain is transferred to another. Pre-trained language models, such as GPT-3 and BERT, have been successfully used as the starting point for transfer learning in a wide variety of NLP tasks. However, the challenge remains of efficiently adapting these large models to new tasks, especially when computational resources are limited. LoRA provides a solution to this challenge by enabling efficient transfer learning [68]. When adapting a pre-trained model to a new task, LoRA only updates a small number of parameters (i.e., the low-rank matrices), making the fine-tuning process computationally efficient [69]. This allows models to be quickly adapted to new tasks with minimal computational overhead [70]. The use of low-rank adaptation also facilitates knowledge transfer, as the model retains most of its pre-trained knowledge while learning new task-specific features. For example, LoRA has been successfully applied in scenarios where a large pre-trained model is transferred to new tasks with limited training data, such as adapting a model for text summarization or question answering [71]. By reducing the number of parameters that need to be trained, LoRA speeds up the transfer learning process and makes it more accessible to researchers and practitioners working with resource-constrained environments [72].

4.4. Zero-Shot Learning

Zero-shot learning involves the ability of a model to make predictions on tasks it has never seen during training [73]. Recent advances in zero-shot learning have focused on leveraging large pre-trained models, which can generalize to a variety of tasks without explicit task-specific training data [74]. LoRA can be combined with zero-shot learning frameworks to improve the adaptability of large models in real-world scenarios. In zero-shot settings, LoRA allows the model to adapt to unseen tasks by introducing low-rank matrices that help the model adjust to the specific requirements of the task. Because LoRA focuses on efficient adaptation, it can provide a more scalable and computationally feasible solution for zero-shot learning, where traditional fine-tuning methods may be too resource-intensive [75]. For example, in a multi-task zero-shot setting, LoRA can be applied to fine-tune a pre-trained language model on a small set of tasks while maintaining the ability to generalize to new tasks with little additional computation. This capability makes LoRA particularly useful in dynamic environments, such as automated customer support or content moderation, where the tasks may frequently change and require quick adaptation.

4.5. Real-World Case Studies

Several real-world applications have demonstrated the effectiveness of LoRA in improving the efficiency and performance of large language models across various domains. For instance, in the legal domain, LoRA has been used to adapt large pre-trained models to legal text, enabling more accurate document classification and legal question answering with minimal computational overhead. Similarly, in healthcare, LoRA has been applied to adapt language models to medical literature, providing efficient models for clinical decision support and medical record analysis. Another notable case study involves the use of LoRA in dialogue systems. By applying LoRA, dialogue models trained on general conversational data have been successfully adapted to specific domains, such as customer service or technical support, with significantly reduced computational cost and faster training times.

4.6. Summary

LoRA has proven to be a versatile and efficient method for fine-tuning large language models across a range of NLP applications [76]. From domain adaptation and few-shot learning to transfer learning and zero-shot learning, LoRA provides a computationally efficient solution for adapting pre-trained models to new tasks with minimal resource consumption. Its ability to maintain strong performance while reducing the number of parameters updated during fine-tuning makes it an attractive choice for resource-constrained environments [77]. In the next section, we explore the challenges and limitations of LoRA, as well as potential areas for future improvement.

5. Challenges and Limitations of LoRA

While Low-Rank Adaptation (LoRA) offers significant advantages in terms of computational efficiency and scalability for fine-tuning large language models, it also presents several challenges and limitations. In this section, we explore the main obstacles faced when applying LoRA, including issues related to its generalizability, performance trade-offs, and implementation challenges. Additionally, we discuss areas where future research could address these limitations and improve the overall effectiveness of LoRA.

5.1. Generalization to Complex Tasks

One of the primary challenges in using LoRA is ensuring that the low-rank adaptation captures the task-specific nuances effectively, especially for complex tasks that require the model to learn detailed and intricate patterns. While LoRA is efficient in fine-tuning large models, it may sometimes struggle to generalize to highly specialized or highly diverse tasks. This is particularly true when the rank of the low-rank matrices is too small to capture the full complexity of the target task. In tasks where the relationship between the input and output is complex or involves long-range dependencies, the low-rank matrices may not have enough capacity to adapt the model in a meaningful way [78]. As a result, LoRA's performance could degrade compared to traditional fine-tuning, which updates a larger number of model parameters and allows for more flexible adaptation [79].

5.2. Rank Selection and Trade-offs

The rank of the low-rank matrices introduced by LoRA is a critical hyperparameter that determines the efficiency and performance of the adaptation process [49,80–82]. A lower rank results in fewer parameters to update, leading to faster training times and lower memory usage [83]. However, if the rank is too low, LoRA may fail to capture important task-specific information, leading to suboptimal performance [84]. Conversely, a higher rank increases the model's capacity, but it also introduces a greater number of parameters to train, potentially reducing the efficiency advantages of LoRA. Choosing the optimal rank for the low-rank matrices is not straightforward and often requires careful experimentation [85]. The appropriate rank can vary depending on the task, the size of the pre-trained model, and the amount of available data. In practice, determining the right balance between efficiency and performance remains a key challenge for LoRA.

5.3. Task-Specific Fine-Tuning

While LoRA is effective for many tasks, there are cases where more specialized adaptation methods may be needed. Some tasks may involve highly structured data or require specific architectural modifications that LoRA's low-rank approach is not well-suited to handle. For instance, tasks such as question answering over structured knowledge bases, or tasks requiring reasoning over complex multi-modal data, may require fine-tuning strategies that go beyond LoRA's low-rank adaptation. In these cases, relying solely on LoRA for adaptation may not be sufficient, and combining LoRA with other specialized techniques or modifications to the underlying model architecture might be necessary. Integrating LoRA with other adaptation strategies, such as attention mechanisms or specialized adapters, could improve performance in tasks where LoRA's approach is not fully effective.

5.4. Implementation and Scalability Issues

Although LoRA reduces the number of parameters to update during fine-tuning, it still requires the introduction of additional low-rank matrices into the model's architecture [86]. For very large models, this can still pose implementation challenges, particularly when it comes to memory usage and storage [87]. While LoRA is designed to be computationally efficient, the memory overhead required for storing the low-rank matrices can still be substantial when dealing with models containing billions of parameters. Moreover, implementing LoRA in practice may require modifications to existing model architectures, which can be non-trivial for large-scale systems or models with complex structures. The scalability of LoRA to extremely large models or resource-constrained environments is an ongoing area of research. Efficient implementation strategies, such as distributed training or memory optimization techniques, may be necessary to handle the increased demands of large models.

5.5. Task Transfer and Adaptation Across Domains

LoRA is designed to be applied across a variety of NLP tasks, but it faces challenges when transferring knowledge from one domain to another [88]. In scenarios where the source and target domains are very different (e.g., transferring knowledge from a general-purpose language model to a highly specialized domain like scientific research), LoRA's low-rank adaptation may not always perform well. The transferability of the learned low-rank matrices may be limited, especially if the domains differ significantly in terms of linguistic structure, terminology, or context. To address this challenge, additional strategies such as domain-specific pre-training, multi-task learning, or cross-domain adaptation could be employed to improve LoRA's ability to transfer knowledge across diverse domains. However, these approaches introduce additional complexity and may not always be practical for every application.

5.6. Overfitting in Small Datasets

Although LoRA is well-suited for few-shot learning, there remains the possibility of overfitting in scenarios where the available labeled data is particularly small or noisy. While LoRA reduces the number of parameters that need to be trained, it does not completely eliminate the risk of overfitting when data is scarce [89]. Low-rank adaptation may still lead to memorization of the limited data, especially if the rank of the low-rank matrices is too high or if the data does not contain sufficient variability. To mitigate overfitting, additional techniques such as regularization, data augmentation, or cross-validation may be necessary. Ensuring that LoRA is used in conjunction with proper training strategies is essential to prevent the model from overfitting to the small dataset [90].

5.7. Summary of Challenges and Limitations

In summary, while LoRA offers significant advantages in terms of computational efficiency and scalability for fine-tuning large language models, it also comes with several challenges and limitations.

These include difficulties in generalizing to complex tasks, the challenge of selecting the optimal rank for low-rank matrices, limitations in task-specific fine-tuning, implementation and scalability issues, difficulties with task transfer across domains, and the risk of overfitting in small datasets [91]. Despite these challenges, LoRA remains a promising technique, and ongoing research into these limitations will likely lead to improvements in its effectiveness and applicability [92]. In the next section, we provide an outlook on the future directions for LoRA and potential areas of further research.

6. Future Directions and Research Opportunities

While Low-Rank Adaptation (LoRA) has proven to be an effective and efficient method for fine-tuning large language models, there are still many opportunities for future research and improvement. In this section, we explore potential directions for enhancing LoRA's capabilities, expanding its applicability to new tasks, and addressing some of its current limitations. We also discuss emerging research areas where LoRA could play a significant role, such as in multi-modal learning, cross-lingual transfer, and model robustness.

6.1. Adaptive Rank Selection

One of the most critical hyperparameters in LoRA is the rank of the low-rank matrices [93]. As discussed earlier, selecting the optimal rank is a challenging task that directly affects the performance and efficiency of LoRA. Currently, rank selection is typically done through experimentation, but more sophisticated methods could be developed to automatically adjust the rank during the fine-tuning process based on the complexity of the task [94]. Future research could explore adaptive rank selection strategies that dynamically adjust the rank of the low-rank matrices during training. Such approaches would allow LoRA to automatically balance the trade-off between computational efficiency and task-specific accuracy [95]. This would make LoRA more flexible and easier to apply to a wider range of tasks without requiring extensive hyperparameter tuning.

6.2. Integration with Other Fine-Tuning Techniques

Although LoRA is effective on its own, there may be opportunities to enhance its performance by combining it with other fine-tuning techniques. For example, LoRA could be integrated with adapter networks, prompt-tuning, or regularization methods to improve its adaptability and robustness [96]. Such hybrid approaches could allow LoRA to take advantage of the strengths of each technique, leading to better task-specific performance while maintaining its computational efficiency [97]. Furthermore, LoRA could be combined with more advanced optimization techniques, such as meta-learning or reinforcement learning, to better adapt to new tasks with limited data. These approaches could enable LoRA to automatically learn which adaptation strategies are most effective for a given task, improving the overall performance of the fine-tuned model [98].

6.3. Domain-Specific and Cross-Domain Adaptation

One of the current limitations of LoRA is its ability to adapt across highly different domains, where the linguistic structures and context vary significantly [99]. Future research could focus on improving LoRA's ability to transfer knowledge between domains with very different characteristics [100]. For example, adapting a general-purpose language model to highly specialized fields such as legal or medical texts, or even cross-lingual transfer, remains a challenging problem [101]. To address these challenges, LoRA could be extended to incorporate domain-specific knowledge during the adaptation process [102]. This could involve incorporating domain-specific pre-training or using multi-task learning frameworks to adapt the model across related tasks [103]. Additionally, research into cross-lingual transfer with LoRA could help improve the adaptability of large models to multiple languages, making LoRA a valuable tool for global applications [104].

6.4. Multi-Modal Learning and LoRA

Multi-modal learning, which involves integrating data from multiple modalities (e.g., text, images, and audio), is an emerging area of research in machine learning. In many cases, pre-trained models for one modality, such as language models, need to be adapted to handle other modalities. LoRA could potentially play a significant role in multi-modal learning by enabling efficient adaptation of pre-trained models to multi-modal tasks. For example, a large pre-trained language model could be adapted using LoRA to integrate with visual or audio data, allowing for efficient fine-tuning of a multi-modal model. Research into how LoRA can be applied to multi-modal tasks could explore the use of low-rank adaptation in different types of architectures, such as those that combine transformers for text and convolutional networks for image data. LoRA could also be applied to transfer learning across multiple modalities, making it easier to adapt large models to tasks that require both text and visual understanding.

6.5. Model Robustness and Generalization

Another promising area for future research is improving the robustness and generalization of LoRA [105]. While LoRA has been shown to be efficient for fine-tuning large language models, the generalization ability of LoRA-adapted models may sometimes be limited when applied to new, unseen data [106]. In particular, LoRA models may not always perform well when faced with noisy or adversarial inputs. Research into improving the robustness of LoRA-adapted models could involve integrating techniques from adversarial training, data augmentation, or uncertainty modeling [107]. These methods could help make LoRA more resilient to out-of-distribution data and ensure that fine-tuned models maintain strong performance in real-world, noisy environments.

6.6. LoRA in Federated Learning and Privacy-Preserving AI

Federated learning, which enables models to be trained across decentralized devices while keeping data local, is gaining traction as a method for privacy-preserving AI. LoRA could potentially be applied to federated learning scenarios, where fine-tuning large models on distributed data is required [108]. By introducing low-rank matrices during fine-tuning, LoRA could reduce the communication and storage overhead typically associated with federated learning. Research into applying LoRA to federated learning would require adapting LoRA's low-rank matrices to work in a distributed setting. Additionally, LoRA could be integrated with privacy-preserving techniques such as differential privacy to ensure that sensitive data remains protected during the fine-tuning process.

6.7. Benchmarking and Standardization

As LoRA continues to gain popularity, it is essential to establish comprehensive benchmarks and standardization practices to evaluate its performance across a variety of tasks and models. Currently, there is a lack of standardized benchmarks that specifically focus on evaluating low-rank adaptation methods like LoRA [109]. Future work could include developing new benchmarks to assess the performance of LoRA across different domains, tasks, and pre-trained models. These benchmarks could help guide researchers in selecting the most appropriate rank for different tasks and identify areas where LoRA's performance can be improved. Additionally, standardized frameworks for implementing and comparing LoRA with other fine-tuning methods would help foster greater collaboration and innovation in the field.

6.8. Summary

In conclusion, there are many exciting future directions for research into Low-Rank Adaptation (LoRA) [110]. These include improving adaptive rank selection, integrating LoRA with other fine-tuning techniques, expanding its applicability to cross-domain and multi-modal tasks, enhancing model robustness, and applying LoRA in federated learning and privacy-preserving AI contexts. Additionally, developing comprehensive benchmarks for evaluating LoRA will help ensure that future research can systematically improve and assess the effectiveness of LoRA. As the field of machine learning continues to evolve, LoRA has the potential to become a key tool in efficiently adapting large language models to a wide range of tasks while minimizing the computational cost [111].

7. Conclusion

In this survey, we have explored the concept of Low-Rank Adaptation (LoRA) and its application to the fine-tuning of large language models. LoRA offers a promising solution to the challenges associated with adapting pre-trained models to new tasks in a computationally efficient manner. By introducing low-rank matrices into the model's weight updates, LoRA reduces the number of parameters that need to be fine-tuned, thus significantly lowering memory and computational requirements while maintaining strong performance on a wide range of natural language processing (NLP) tasks.

We have discussed the theoretical foundations of LoRA, its applications in domain adaptation, few-shot learning, transfer learning, and zero-shot learning, as well as real-world case studies where LoRA has been successfully employed. Despite its advantages, we also addressed several challenges and limitations of LoRA, including issues related to rank selection, generalization to complex tasks, and the risk of overfitting in small datasets. Moreover, we outlined future research opportunities, including adaptive rank selection, integration with other fine-tuning techniques, cross-domain and multi-modal adaptation, and improvements in robustness and generalization.

LoRA's potential for efficient fine-tuning and adaptability to new tasks makes it a valuable tool for many NLP applications, particularly when resources are limited or data is scarce. As the field of machine learning continues to evolve, LoRA will likely remain a crucial method for enabling large-scale language models to perform effectively across a wide variety of tasks. Further research in addressing its limitations and expanding its capabilities will ensure that LoRA continues to play a central role in the future of NLP and AI.

In conclusion, LoRA represents a significant step toward making large pre-trained language models more accessible and adaptable for diverse applications. By improving the efficiency of model adaptation without compromising performance, LoRA holds promise for advancing the field of NLP and making state-of-the-art models more practical and sustainable for real-world use.

References

1. Ge, Y.; Ge, Y.; Zeng, Z.; Wang, X.; Shan, Y. Planting a SEED of Vision in Large Language Model. *arXiv preprint arXiv:2307.08041* **2023**.
2. Sakaguchi, K.; Bras, R.L.; Bhagavatula, C.; Choi, Y. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM* **2021**, *64*, 99–106.
3. Xiong, W.; Liu, J.; Molybog, I.; Zhang, H.; Bhargava, P.; Hou, R.; Martin, L.; Rungta, R.; Sankararaman, K.A.; Oguz, B.; et al. Effective Long-Context Scaling of Foundation Models. *arXiv preprint arXiv.2309.16039* **2023**.
4. Quan, S. DMoERM: Recipes of Mixture-of-Experts for Effective Reward Modeling. *arXiv preprint arXiv:2403.01197* **2024**.
5. Yang, S.; Zhou, Y.; Liu, Z.; Loy, C.C. Rerender A Video: Zero-Shot Text-Guided Video-to-Video Translation. In Proceedings of the SIGGRAPH Asia 2023 Conference Papers, 2023, pp. 1–11.
6. Wang, H.; Xiang, X.; Fan, Y.; Xue, J. Customizing 360-Degree Panoramas through Text-to-Image Diffusion Models. In Proceedings of the Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 4933–4943.
7. Hayou, S.; Ghosh, N.; Yu, B. The Impact of Initialization on LoRA Finetuning Dynamics. *arXiv preprint arXiv:2406.08447* **2024**.
8. Li, Z.; Li, X.; Liu, Y.; Xie, H.; Li, J.; Wang, F.L.; Li, Q.; Zhong, X. Label Supervised LLaMA Finetuning. *arXiv preprint arXiv:2310.01208* **2023**.
9. Edalati, A.; Tahaei, M.S.; Kobyzev, I.; Nia, V.P.; Clark, J.J.; Rezagholizadeh, M. KronA: Parameter Efficient Tuning with Kronecker Adapter. *arXiv preprint arXiv.2212.10650* **2022**.
10. Zhou, X.; Sun, Z.; Li, G. Db-gpt: Large language model meets database. *Data Science and Engineering* **2024**, *9*, 102–111.

11. Liu, Y.; Yu, C.; Shang, L.; He, Y.; Wu, Z.; Wang, X.; Xu, C.; Xie, H.; Wang, W.; Zhao, Y.; et al. FaceChain: A Playground for Human-centric Artificial Intelligence Generated Content. *arXiv preprint arXiv:2308.14256* **2023**.
12. Wang, X.; Aitchison, L.; Rudolph, M. LoRA ensembles for large language model fine-tuning. *arXiv preprint arXiv.2310.00035* **2023**.
13. Chen, S.; Jie, Z.; Ma, L. LLaVA-MoLE: Sparse Mixture of LoRA Experts for Mitigating Data Conflicts in Instruction Finetuning MLLMs. *arXiv preprint arXiv.2401.16160* **2024**.
14. Zniyed, Y.; Nguyen, T.P.; et al. Efficient tensor decomposition-based filter pruning. *Neural Networks* **2024**, *178*, 106393.
15. Zhang, Y.; Xu, Q.; Zhang, L. DragTex: Generative Point-Based Texture Editing on 3D Mesh. *arXiv preprint arXiv:2403.02217* **2024**.
16. Gu, Y.; Wang, X.; Wu, J.Z.; Shi, Y.; Chen, Y.; Fan, Z.; Xiao, W.; Zhao, R.; Chang, S.; Wu, W.; et al. Mix-of-Show: Decentralized Low-Rank Adaptation for Multi-Concept Customization of Diffusion Models. In Proceedings of the Advances in Neural Information Processing Systems, 2023.
17. Luo, Z.; Xu, X.; Liu, F.; Koh, Y.S.; Wang, D.; Zhang, J. Privacy-Preserving Low-Rank Adaptation for Latent Diffusion Models. *arXiv preprint arXiv.2402.11989* **2024**.
18. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Guo, Q.; Wang, M.; et al. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv.2312.10997* **2023**.
19. Frenkel, Y.; Vinker, Y.; Shamir, A.; Cohen-Or, D. Implicit Style-Content Separation using B-LoRA. *arXiv preprint arXiv:2403.14572* **2024**.
20. Ren, W.; Li, X.; Wang, L.; Zhao, T.; Qin, W. Analyzing and Reducing Catastrophic Forgetting in Parameter Efficient Tuning. *arXiv preprint arXiv:2402.18865* **2024**.
21. Ye, Z.; Li, D.; Tian, J.; Lan, T.; Zuo, J.; Duan, L.; Lu, H.; Jiang, Y.; Sha, J.; Zhang, K.; et al. ASPEN: High-Throughput LoRA Fine-Tuning of Large Language Models with a Single GPU. *arXiv preprint arXiv:2312.02515* **2023**.
22. Jacot, A.; Hongler, C.; Gabriel, F. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In Proceedings of the Annual Conference on Neural Information Processing Systems, 2018, pp. 8580–8589.
23. Tan, W.; Zhang, W.; Liu, S.; Zheng, L.; Wang, X.; An, B. True Knowledge Comes from Practice: Aligning LLMs with Embodied Environments via Reinforcement Learning. *arXiv preprint arXiv:2401.14151* **2024**.
24. Silva, A.; Fang, S.; Monperrus, M. Repairllama: Efficient representations and fine-tuned adapters for program repair. *arXiv preprint arXiv:2312.15698* **2023**.
25. Zhang, Y.; Wang, M.; Wu, Y.; Tiwari, P.; Li, Q.; Wang, B.; Qin, J. DialogueLLM: Context and Emotion Knowledge-Tuned Large Language Models for Emotion Recognition in Conversations. *arXiv preprint arXiv:2310.11374* **2024**.
26. Jin, Z.; Song, Z. Generating coherent comic with rich story using ChatGPT and Stable Diffusion. *arXiv preprint arXiv:2305.11067* **2023**.
27. Sui, Y.; Yin, M.; Gong, Y.; Xiao, J.; Phan, H.; Yuan, B. ELRT: Efficient Low-Rank Training for Compact Convolutional Neural Networks. *arXiv preprint arXiv.2401.10341* **2024**.
28. Jiang, W.; Lin, B.; Shi, H.; Zhang, Y.; Li, Z.; Kwok, J.T. Effective and Parameter-Efficient Reusing Fine-Tuned Models. *arXiv preprint arXiv.2310.01886* **2023**.
29. Liu, Y.; An, C.; Qiu, X. $$\cal{Y}$$-Tuning: an efficient tuning paradigm for large-scale pre-trained models via label representation learning. *Frontiers Comput. Sci.* **2024**, *18*.
30. Elsken, T.; Metzen, J.H.; Hutter, F. Neural Architecture Search: A Survey. *J. Mach. Learn. Res.* **2019**, *20*, 55:1–55:21.
31. Malladi, S.; Wettig, A.; Yu, D.; Chen, D.; Arora, S. A Kernel-Based View of Language Model Fine-Tuning. In Proceedings of the International Conference on Machine Learning, 2023, pp. 23610–23641.
32. Liang, Y.; Li, W. InfLoRA: Interference-Free Low-Rank Adaptation for Continual Learning. *arXiv preprint arXiv.2404.00228* **2024**.
33. Konstantinidis, T.; Iacovides, G.; Xu, M.; Constantinides, T.G.; Mandic, D.P. FinLlama: Financial Sentiment Classification for Algorithmic Trading Applications. *arXiv preprint arXiv:2403.12285* **2024**.
34. Kong, Z.; Zhang, Y.; Yang, T.; Wang, T.; Zhang, K.; Wu, B.; Chen, G.; Liu, W.; Luo, W. OMG: Occlusion-friendly Personalized Multi-concept Generation in Diffusion Models. *arXiv preprint arXiv:2403.10983* **2024**.
35. He, X.; Li, C.; Zhang, P.; Yang, J.; Wang, X.E. Parameter-Efficient Model Adaptation for Vision Transformers. In Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence, 2023, pp. 817–825.

36. Koubbi, H.; Boussard, M.; Hernandez, L. The Impact of LoRA on the Emergence of Clusters in Transformers. *arXiv preprint arXiv.2402.15415* **2024**.
37. Khandelwal, A. InFusion: Inject and Attention Fusion for Multi Concept Zero-Shot Text-based Video Editing. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 3017–3026.
38. Fomenko, V.; Yu, H.; Lee, J.; Hsieh, S.; Chen, W. A Note on LoRA. *arXiv preprint arXiv:2404.05086* **2024**.
39. Zhao, J.; Wang, T.; Abid, W.; Angus, G.; Garg, A.; Kinnison, J.; Sherstinsky, A.; Molino, P.; Addair, T.; Rishi, D. LoRA Land: 310 Fine-tuned LLMs that Rival GPT-4, A Technical Report. *arXiv preprint arXiv.2405.00732* **2024**.
40. Zhu, J.; Greenewald, K.H.; Nadjahi, K.; de Ocáriz Borde, H.S.; Gabrielsson, R.B.; Choshen, L.; Ghassemi, M.; Yurochkin, M.; Solomon, J. Asymmetry in Low-Rank Adapters of Foundation Models. *arXiv preprint arXiv.2402.16842* **2024**.
41. Mujadia, V.; Urlana, A.; Bhaskar, Y.; Pavani, P.A.; Shravya, K.; Krishnamurthy, P.; Sharma, D.M. Assessing Translation Capabilities of Large Language Models Involving English and Indian Languages. *arXiv preprint arXiv:2311.09216* **2023**.
42. Yoo, S.; Kim, K.; Kim, V.G.; Sung, M. As-Plausible-As-Possible: Plausibility-Aware Mesh Deformation Using 2D Diffusion Priors. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 4315–4324.
43. Ma, Y.; Fan, Y.; Ji, J.; Wang, H.; Sun, X.; Jiang, G.; Shu, A.; Ji, R. X-Dreamer: Creating High-quality 3D Content by Bridging the Domain Gap Between Text-to-2D and Text-to-3D Generation. *arXiv preprint arXiv:2312.00085* **2023**.
44. Wang, Y.; Lin, Y.; Zeng, X.; Zhang, G. MultiLoRA: Democratizing LoRA for Better Multi-Task Learning. *arXiv preprint arXiv.2311.11501* **2023**.
45. Liu, T.; Low, B.K.H. Goat: Fine-tuned LLaMA Outperforms GPT-4 on Arithmetic Tasks. *arXiv preprint arXiv:2305.14201* **2023**.
46. Shrestha, S.; Venkataramanan, A.; et al. Style Transfer to Calvin and Hobbes comics using Stable Diffusion. *arXiv preprint arXiv:2312.03993* **2023**.
47. Liu, S.; Wang, C.; Yin, H.; Molchanov, P.; Wang, Y.F.; Cheng, K.; Chen, M. DoRA: Weight-Decomposed Low-Rank Adaptation. *arXiv preprint arXiv.2402.09353* **2024**.
48. Yang, S.; Ali, M.A.; Wang, C.; Hu, L.; Wang, D. MoRAL: MoE Augmented LoRA for LLMs' Lifelong Learning. *arXiv preprint arXiv.2402.11260* **2024**.
49. Zniyed, Y.; Nguyen, T.P.; et al. Enhanced network compression through tensor decompositions and pruning. *IEEE Transactions on Neural Networks and Learning Systems* **2024**.
50. Luo, S.; Tan, Y.; Patil, S.; Gu, D.; von Platen, P.; Passos, A.; Huang, L.; Li, J.; Zhao, H. LCM-LoRA: A Universal Stable-Diffusion Acceleration Module. *arXiv preprint arXiv:2311.05556* **2023**.
51. Zhang, K.; Liu, D. Customized Segment Anything Model for Medical Image Segmentation. *arXiv preprint arXiv:2304.13785* **2023**.
52. Lin, Y.; Ma, X.; Chu, X.; Jin, Y.; Yang, Z.; Wang, Y.; Mei, H. Lora dropout as a sparsity regularizer for overfitting control. *arXiv preprint arXiv:2404.09610* **2024**.
53. Suri, K.; Mishra, P.; Saha, S.; Singh, A. SuryaKiran at MEDIQA-Sum 2023: Leveraging LoRA for Clinical Dialogue Summarization. In Proceedings of the Working Notes of the Conference and Labs of the Evaluation Forum, 2023, pp. 1720–1735.
54. Zhang, M.; Chen, H.; Shen, C.; Yang, Z.; Ou, L.; Yu, X.; Zhuang, B. Loraprune: Pruning meets low-rank parameter-efficient fine-tuning. *arXiv preprint arXiv:2305.18403* **2023**.
55. Jin, F.; Liu, Y.; Tan, Y. Derivative-Free Optimization for Low-Rank Adaptation in Large Language Models. *arXiv preprint arXiv:2403.01754* **2024**.
56. Woo, S.; Park, B.; Kim, B.; Jo, M.; Kwon, S.; Jeon, D.; Lee, D. DropBP: Accelerating Fine-Tuning of Large Language Models by Dropping Backward Propagation. *arXiv preprint arXiv:2402.17812* **2024**.
57. Na, S.; Guo, Y.; Jiang, F.; Ma, H.; Huang, J. Segment Any Cell: A SAM-based Auto-prompting Fine-tuning Framework for Nuclei Segmentation. *arXiv preprint arXiv:2401.13220* **2024**.
58. Feng, W.; Zhu, L.; Yu, L. Cheap Lunch for Medical Image Segmentation by Fine-tuning SAM on Few Exemplars. *arXiv preprint arXiv:2308.14133* **2023**.
59. Yu, K.; Liu, J.; Feng, M.; Cui, M.; Xie, X. Boosting3D: High-Fidelity Image-to-3D by Boosting 2D Diffusion Prior to 3D Prior with Progressive Learning. *arXiv preprint arXiv:2311.13617* **2023**.

60. Liu, Z.; Kundu, S.; Li, A.; Wan, J.; Jiang, L.; Beerel, P.A. AFLoRA: Adaptive Freezing of Low Rank Adaptation in Parameter Efficient Fine-Tuning of Large Models. *arXiv preprint arXiv:2403.13269* **2024**.
61. Zaken, E.B.; Goldberg, Y.; Ravfogel, S. BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models. In Proceedings of the Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2022, pp. 1–9.
62. Tran, H.; Yang, Z.; Yao, Z.; Yu, H. BioInstruct: Instruction Tuning of Large Language Models for Biomedical Natural Language Processing. *arXiv preprint arXiv:2310.19975* **2023**.
63. Valipour, M.; Rezagholizadeh, M.; Kobyzev, I.; Ghodsi, A. Dylora: Parameter efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. *arXiv preprint arXiv:2210.07558* **2022**.
64. Zhang, F.; Pilanci, M. Riemannian Preconditioned LoRA for Fine-Tuning Foundation Models. *arXiv preprint arXiv:2402.02347* **2024**.
65. Li, J.; Lei, Y.; Bian, Y.; Cheng, D.; Ding, Z.; Jiang, C. RA-CFGPT: Chinese financial assistant with retrieval-augmented large language model. *Frontiers of Computer Science* **2024**, *18*, 185350.
66. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In Proceedings of the Proceedings of the 34th International Conference on Machine Learning, 2017, pp. 1126–1135.
67. Zhang, H. SinkLoRA: Enhanced Efficiency and Chat Capabilities for Long-Context Large Language Models. *arXiv preprint arXiv.2406.05678* **2023**.
68. Chen, Y.; Qian, S.; Tang, H.; Lai, X.; Liu, Z.; Han, S.; Jia, J. LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models. *arXiv preprint arXiv.2309.12307* **2023**.
69. Liu, Y.; An, C.; Qiu, X. Y-tuning: An efficient tuning paradigm for large-scale pre-trained models via label representation learning. *Frontiers of Computer Science* **2024**, *18*, 184320.
70. Wu, T.; Wang, J.; Zhao, Z.; Wong, N. Mixture-of-Subspaces in Low-Rank Adaptation. *arXiv preprint arXiv:2406.11909* **2024**.
71. Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Wu, Z.; Chang, B.; Sun, X.; Xu, J.; Li, L.; Sui, Z. A Survey for In-context Learning. *arXiv preprint arXiv.2301.00234* **2023**.
72. Li, S. DiffStyler: Diffusion-based Localized Image Style Transfer. *arXiv preprint arXiv:2403.18461* **2024**.
73. Lester, B.; Al-Rfou, R.; Constant, N. The Power of Scale for Parameter-Efficient Prompt Tuning. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 3045–3059.
74. Jang, U.; Lee, J.D.; Ryu, E.K. LoRA Training in the NTK Regime has No Spurious Local Minima. *arXiv preprint arXiv.2402.11867* **2024**.
75. Chen, X.; Wang, C.; Ning, H.; Li, S. SAM-OCTA: Prompting Segment-Anything for OCTA Image Segmentation. *arXiv preprint arXiv:2310.07183* **2023**.
76. Bałazy, K.; Banaei, M.; Aberer, K.; Tabor, J. LoRA-XS: Low-Rank Adaptation with Extremely Small Number of Parameters. *arXiv preprint arXiv:2405.17604* **2024**.
77. Ding, N.; Lv, X.; Wang, Q.; Chen, Y.; Zhou, B.; Liu, Z.; Sun, M. Sparse Low-rank Adaptation of Pre-trained Language Models. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 4133–4145.
78. Zhu, Y.; Yang, X.; Wu, Y.; Zhang, W. Parameter-efficient fine-tuning with layer pruning on free-text sequence-to-sequence modeling. *arXiv preprint arXiv:2305.08285* **2023**.
79. Zi, B.; Qi, X.; Wang, L.; Wang, J.; Wong, K.; Zhang, L. Delta-LoRA: Fine-Tuning High-Rank Parameters with the Delta of Low-Rank Matrices. *arXiv preprint arXiv.2309.02411* **2023**.
80. Zhu, Y.; Wichers, N.; Lin, C.; Wang, X.; Chen, T.; Shu, L.; Lu, H.; Liu, C.; Luo, L.; Chen, J.; et al. SiRA: Sparse Mixture of Low Rank Adaptation. *arXiv preprint arXiv.2311.09179* **2023**.
81. Kopiczko, D.J.; Blankevoort, T.; Asano, Y.M. Vera: Vector-based random matrix adaptation. *arXiv preprint arXiv:2310.11454* **2023**.
82. Wen, Y.; Chaudhuri, S. Batched Low-Rank Adaptation of Foundation Models. *arXiv preprint arXiv.2312.05677* **2023**.
83. Zhang, S.; Chen, Z.; Chen, S.; Shen, Y.; Sun, Z.; Gan, C. Improving Reinforcement Learning from Human Feedback with Efficient Reward Model Ensemble. *arXiv preprint arXiv:2401.16635* **2024**.
84. Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300* **2020**.
85. Liao, B.; Monz, C. ApiQ: Finetuning of 2-Bit Quantized Large Language Model. *arXiv preprint arXiv:2402.05147* **2024**.

86. Smith, J.S.; Cascante-Bonilla, P.; Arbelle, A.; Kim, D.; Panda, R.; Cox, D.D.; Yang, D.; Kira, Z.; Feris, R.; Karlinsky, L. ConStruct-VL: Data-Free Continual Structured VL Concepts Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 14994–15004.
87. Huang, T.; Zeng, Y.; Zhang, Z.; Xu, W.; Xu, H.; Xu, S.; Lau, R.W.H.; Zuo, W. DreamControl: Control-Based Text-to-3D Generation with 3D Self-Prior. *arXiv preprint arXiv:2312.06439* **2023**.
88. Liu, Z.; Li, S.; Luo, Y.; Fei, H.; Cao, Y.; Kawaguchi, K.; Wang, X.; Chua, T. MolCA: Molecular Graph-Language Modeling with Cross-Modal Projector and Uni-Modal Adapter. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 15623–15638.
89. Zhong, W.; Cui, R.; Guo, Y.; Liang, Y.; Lu, S.; Wang, Y.; Saied, A.; Chen, W.; Duan, N. Agieval: A human-centric benchmark for evaluating foundation models. *arXiv preprint arXiv:2304.06364* **2023**.
90. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461* **2018**.
91. Han, Z.; Gao, C.; Liu, J.; Zhang, S.Q.; et al. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608* **2024**.
92. Blattmann, A.; Dockhorn, T.; Kulal, S.; Mendelevitch, D.; Kilian, M.; Lorenz, D.; Levi, Y.; English, Z.; Voleti, V.; Letts, A.; et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127* **2023**.
93. Wang, S.; Chen, L.; Jiang, J.; Xue, B.; Kong, L.; Wu, C. LoRA Meets Dropout under a Unified Framework. *arXiv preprint arXiv:2403.00812* **2024**.
94. Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In Proceedings of the Proceedings of the 33nd International Conference on Machine Learning, 2016, pp. 1050–1059.
95. Zhang, J.; Chen, S.; Liu, J.; He, J. Composing Parameter-Efficient Modules with Arithmetic Operations. *arXiv preprint arXiv.2306.14870* **2023**.
96. Zhang, Y.; Wang, J.; Yu, L.; Xu, D.; Zhang, X. Personalized LoRA for Human-Centered Text Understanding. In Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence, 2024, pp. 19588–19596.
97. Qi, Z.; Tan, X.; Shi, S.; Qu, C.; Xu, Y.; Qi, Y. PILLOW: Enhancing Efficient Instruction Fine-tuning via Prompt Matching. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track, 2023, pp. 471–482.
98. Yang, Y.; Xiong, S.; Payani, A.; Shareghi, E.; Fekri, F. Harnessing the Power of Large Language Models for Natural Language to First-Order Logic Translation. *arXiv preprint arXiv:2305.15541* **2023**.
99. Bai, J.; Chen, D.; Qian, B.; Yao, L.; Li, Y. Federated Fine-tuning of Large Language Models under Heterogeneous Language Tasks and Client Resources. *arXiv preprint arXiv:2402.11505* **2024**.
100. Ye, M.; Fang, X.; Du, B.; Yuen, P.C.; Tao, D. Heterogeneous Federated Learning: State-of-the-art and Research Challenges. *ACM Computing Surveys* **2024**, *56*, 79:1–79:44.
101. Zhang, Q.; Chen, M.; Bukharin, A.; He, P.; Cheng, Y.; Chen, W.; Zhao, T. Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning. In Proceedings of the The Eleventh International Conference on Learning Representations, 2023.
102. Agiza A, Neseem M, R.S. MTLoRA: Low-Rank Adaptation Approach for Efficient Multi-Task Learning. In Proceedings of the CVPR, 2024.
103. Ye, Z.; Lovell, L.; Faramarzi, A.; Ninic, J. SAM-based instance segmentation models for the automation of structural damage detection. *arXiv preprint arXiv:2401.15266* **2024**.
104. Wang, Y.; Lin, Y.; Zeng, X.; Zhang, G. PrivateLoRA For Efficient Privacy Preserving LLM. *arXiv preprint arXiv:2311.14030* **2023**.
105. Qin, H.; Ma, X.; Zheng, X.; Li, X.; Zhang, Y.; Liu, S.; Luo, J.; Liu, X.; Magno, M. Accurate LoRA-Finetuning Quantization of LLMs via Information Retention. *arXiv preprint arXiv:2402.05445* **2024**.
106. Zhou, H.; Lu, X.; Xu, W.; Zhu, C.; Zhao, T. LoRA-drop: Efficient LoRA Parameter Pruning based on Output Evaluation. *arXiv preprint arXiv:2402.07721* **2024**.
107. Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.; Chen, W.; et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nat. Mac. Intell.* **2023**, *5*, 220–235.
108. Zhao, H.; Ni, B.; Wang, H.; Fan, J.; Zhu, F.; Wang, Y.; Chen, Y.; Meng, G.; Zhang, Z. Continual Forgetting for Pre-trained Vision Models. *arXiv preprint arXiv.2403.11530* **2024**.
109. Han, A.; Li, J.; Huang, W.; Hong, M.; Takeda, A.; Jawanpuria, P.; Mishra, B. SLTrain: a sparse plus low-rank approach for parameter and memory efficient pretraining. *arXiv preprint arXiv:2406.02214* **2024**.

110. Mao, Y.; Huang, K.; Guan, C.; Bao, G.; Mo, F.; Xu, J. DoRA: Enhancing Parameter-Efficient Fine-Tuning with Dynamic Rank Distribution. *arXiv preprint arXiv:2405.17357* **2024**.
111. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the Proceedings of North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, 2019, pp. 4171–4186.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.