СЕКЦИЯ – ТЕХНИЧЕСКИЕ НАУКИ

SECTION –TECHNICAL SCIENCES

UDK 004.04

Mozharovskii Evgenii
bachelor's degree, Lomonosov Moscow State University
Russian Federation, Moscow

LORA AND QLORA IN LLM FINE-TUNING FOR CUSTOM CODING APPLICATIONS: EFFICIENCY AND SCALABILITY

Abstract: The article discusses the use of Low-Rank Adaptation (LoRA) and Quantized Low-Rank Adaptation (QLoRA) methods in the retraining of large language models (LLM) for specialized programming tasks. The similarities and differences between the LoRA and QLoRA methods are discussed. The advantages of these methods in the context of improving the efficiency and scalability of retraining, as well as their impact on the quality and speed of models are investigated. A comparison of traditional approaches and proposed methods is carried out, emphasizing their potential for reducing computational costs and optimizing resources while maintaining high accuracy and performance in applied tasks. Examples of practical use of LoRA and QLoRA methods in various fields are given.

Keywords: Low-rank decomposition (LoRA), quantized low-rank decomposition (QLoRA), large language model, fine-tuning, custom coding.

INTRODUCTION

Artificial intelligence and machine learning technologies continue to transform various fields of human activity, including software development. One of the significant achievements in this area is large language models (LLM), such as GPT-3 and GPT-4, which demonstrate capabilities in natural language generation and understanding. These models not only enhance the process of automatic text generation but are also widely applied in programming tasks, enabling developers to automate routine tasks and increase productivity.

Because LLM are trained on broad and generalized datasets, fine-tuning is essential for creating applications specific to particular domains. Traditionally, this fine-tuning process required significant computational resources and time. However,

the emergence of innovative methods offering efficient alternatives has drastically reduced these requirements. Among these methods, low-rank adaptation (LoRA) and quantized low-rank adaptation (QLoRA) stand out. This paper aims to explore the potential of applying LoRA and QLoRA methods for fine-tuning LLM in the context of custom coding applications, and to analyze their efficiency and scalability compared to traditional methods.

MAIN PART. LARGE LANGUAGE MODELS

The term LLM refers to machine learning models that utilize deep learning techniques and vast amounts of training data to understand and generate natural language. Their ability to grasp the meaning and context of words and sentences enables LLM to excel at tasks such as text generation, language translation, and content summarization. They are trained on massive datasets, typically containing billions of words from various sources, such as websites, books, and articles. This extensive training allows LLM to comprehend language nuances, grammar, context, and even some aspects of general knowledge.

These models operate by receiving input data, such as a command or query, applying knowledge derived from extensive training data, and then using neural networks to accurately predict and generate contextually relevant output. They are built on a transformer architecture based on neural networks to understand the relationships between words in sentences. Transformers utilize encoders to process input sequences and decoders to handle output sequences, both of which are layers in their neural network. Existing fine-tuning methods vary in complexity, required resources, and efficiency. Table 1 presents the main approaches to LLM fine-tuning, along with their advantages and disadvantages.

Table 1. Key approaches to LLM fine-tuning, their advantages, and disadvantages [1]

Method	Description	Advantages	Disadvantages
Traditional fine-tuning	Involves further training of the model on a specific dataset.	High accuracy on specific tasks.	Requires substantial computational resources and data.

Transfer learning	The model is initially trained on a large, general dataset and then fine-tuned on a smaller, specialized dataset.	Reduces time and resources required for training.	Potential challenges with knowledge transfer.
Token hedging	Special tokens added to the beginning of input sequences to transfer specific knowledge into the model.	Enables rapid adaptation without full retraining.	Limited accuracy on complex tasks.
Regularization method	Use of regularization techniques, such as dropout and L2 regularization.	Improves generalization capability.	Requires fine-tuning of hyperparameters.
Use of external modules	Integration of external modules or libraries, such as caching or external knowledge bases.	Enhances performance without requiring retraining.	Complexity in integration and additional resource requirements.

From the author's perspective, the choice of LLM fine-tuning method depends on the specific task requirements, available resources, and expected outcomes. It is important to consider both the advantages and disadvantages of each method to effectively apply models to specific programming tasks. The LoRA method may be preferred when it is necessary to retain knowledge gained from training on large datasets and to adapt the model to new data without significantly increasing complexity. On the other hand, if reducing the model size and enhancing its efficiency are priorities, the QLoRA method might be more suitable. It is also crucial to consider available resources, such as memory and computational power, as each method has its own requirements for these resources [2].

LORA AND QLORA AS LLM FINE-TUNING METHODS

Advanced fine-tuning methods for large LLM, such as LoRA and QLoRA, significantly enhance the efficiency and scalability of the fine-tuning process for specialized programming applications. The LoRA method was first proposed in 2021 by researchers from Microsoft Research. In their paper «LoRA: Low-Rank Adaptation of Large Language Models» the authors introduced an approach that allows large language models to be adapted to new tasks using low-rank matrices [3]. The QLoRA method was introduced in 2023 as a further development of the LoRA idea, combining low-rank adaptation with quantization. This method is aimed at achieving maximum

performance with minimal cost, making it particularly attractive for large-scale and distributed systems (fig. 1).

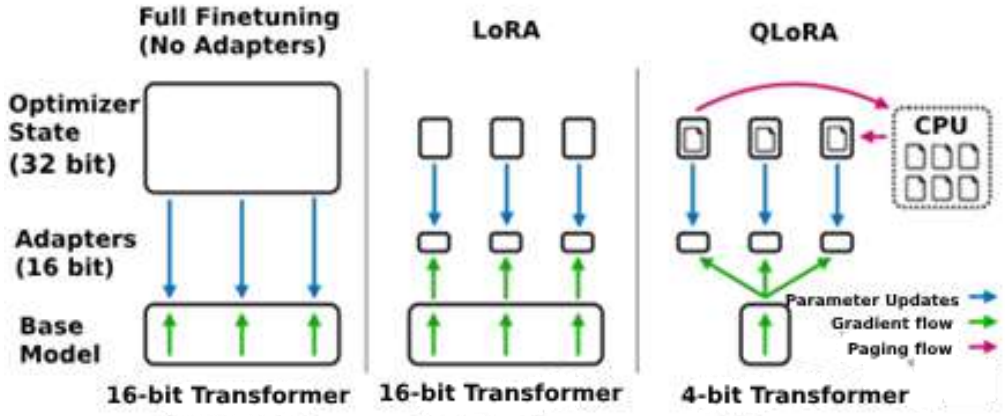

Figure 1. Mechanism of LoRA and QLoRA functioning

Both methods have quickly gained popularity due to their efficiency and ability to significantly improve the quality of fine-tuning large language models for various applied tasks [4]. These methods are based on the idea of reducing the dimensionality of model weights while retaining their informational value, which allows for a reduction in the dimensionality of the training dataset without losing meaningful information.

The **LoRA** method represents an innovative approach to fine-tuning. The main idea of LoRA is to use low-rank matrices to adapt the model's weights to new tasks or training conditions. Fine-tuning an LLM using the LoRA method involves updating only a small number of parameters. This process consists of several steps. The first step involves replacing the model's weight matrix WW with the product of two low-rank matrices AA and BB ($W \approx A \times B W \approx A \times B$). Typically, the rank of these matrices is significantly smaller than the original size of the weight matrix, reducing computational costs. After decomposing the weights, the model is fine-tuned on a specific dataset. Since only the low-rank matrices AA and BB are updated, this significantly reduces the memory required to store gradients and intermediate values. The updated matrices AA and BB are multiplied to obtain the updated weights WW, which are then integrated back into the model.

The advantages of the LoRA method include improved scalability of model fine-tuning, reduced risk of over fitting, decreased training time, and more efficient use of

computational resources. Thanks to its adaptability and optimization capabilities, LoRA has become an important tool for developers and researchers working with LLM in various applied fields.

The **QLoRA** method enhances LoRA's efficiency by incorporating quantization techniques, further reducing memory and computational requirements. The main goal of QLoRA is to improve the efficiency and performance of model fine-tuning by using low-rank matrices to approximate the model's weights. This reduces the number of parameters that need to be updated during fine-tuning, thereby decreasing computational costs and the risk of over fitting. QLoRA applies weight quantization, which is the process of converting the model's weights to a less precise format (e.g., lower precision floating-point), reducing memory usage and speeding up computations. This is particularly useful for working with large models in distributed systems and on resource-constrained devices.

By utilizing NVIDIA's unified memory feature, QLoRA can more efficiently handle GPU memory spikes, ensuring a smoother training process. Key advantages of QLoRA include:

- **Reduced computational costs:** the combination of low-rank adaptation and quantization significantly decreases the computational and memory requirements for fine-tuning.
- **Accelerated training process:** smaller data volumes and improved computational efficiency shorten the model fine-tuning time.
- **High performance:** despite the reduced precision of weights due to quantization, the model maintains high quality and adaptability thanks to the use of low-rank approximations [5].

The use of LoRA and QLoRA not only enhances model performance during fine-tuning but also promotes more efficient deployment in real-world applications, where quick training and high adaptability to changing conditions are required. These methods open new opportunities for developers by reducing model fine-tuning time and improving overall performance with minimal resource expenditure.

COMPARATIVE ANALYSIS OF LORA AND QLORA

Both LoRA and QLoRA are based on the idea of decomposing a matrix into low-rank components to reduce the dimensionality of the training dataset without losing meaningful information. This allows the model to adapt to new data without significantly increasing complexity. LoRA and QLoRA are used to fine-tune LLM on small datasets and improve their performance. They allow the knowledge gained from training on large datasets to be preserved and the model to adapt to new data without significantly increasing complexity. The differences are presented in Table 2.

Table 2. Differences between LoRA and QLoRA

Characteristics	LoRA	QLoRA
Number of low-rank components	Fixed Number of Low-Rank Components	Adaptive number of components that can be optimized during training
Scalability	Medium, suitable for medium-scale models	High scalability, can be easily adapted to work with various dataset sizes and LLM models
Computational complexity	High, as it requires matrix decomposition into low-rank components, which may demand additional computational resources.	Low, because it optimizes the number of components dynamically, reducing unnecessary computational overhead
Accuracy	High accuracy due to effective fine-tuning of specific model components	Medium accuracy, balancing between computational efficiency and performance, but can be fine-tuned further for better results

The described similarities and differences make each method suitable for different situations and tasks, and also allow us to evaluate their advantages. The LoRA and QLoRA methods can be easily scaled to work with different sizes of data sets and LLM models. This makes them universal tools for fine-tuning LLM and ensures their applicability to various types of data, including text, numeric and mixed [6].

PRACTICAL APPLICATIONS OF LORA AND QLORA

The LoRa and QLOR methods provide important advantages in MO, especially when fine-tuning LLM Examples of their practical use cover various fields, enhancing the accessibility and efficiency of model adaptation.

Natural language processing: Using LoRA and QLoRA methods to adapt LLMs, such as GPT, to work in chat bots and virtual assistants, allow you to increase

the accuracy and relevance of responses, while minimizing computing resources. For example, virtual assistants on smartphones can be adapted to understand specific terminology or user accents. The American company **Amazon** uses LoRA to adapt the natural language processing models of the Alexa voice assistant to different regional accents and dialects, improving speech recognition and understanding without significant computational costs [7].

Computer vision: In medical diagnostics, LoRA and QLoRA are used to adapt models that classify medical images such as X-rays or MRIs. This enables quick and accurate disease identification, assisting doctors in diagnosis. In security systems, QLoRA helps adapt models to detect suspicious objects or faces in real time using surveillance cameras [8].

Internet of things (IoT): QLoRA is used to adapt models running on smart devices like home assistants or wearables. This allows for ML tasks such as voice recognition or health data analysis with minimal energy consumption.

Industrial applications: LoRA and QLoRA help adapt models for monitoring and controlling processes in factories. These models can predict equipment failures and optimize production processes in real time.

Mobile applications: LoRA and QLoRA help adapt models to provide recommendations based on user behavior, even with the limited computational resources of the device. This improves user experience by offering more accurate and personalized recommendations.

An example of the use of LoRA and QLoRA methods is the company **OpenAI**, known for its advanced AI research and GPT model development [9]. The implementation of the LoRA method has significantly reduced the memory and computational resources required to retrain the model for specific tasks. In the financial sector, the model is trained on specific data such as financial reports and customer queries using LoRA. This allows ChatGPT to be quickly and efficiently adapted to work with financial terms and specific queries.

The application of LoRA and QLoRA methods in real-world applications demonstrates their ability to improve the performance and efficiency of ML models, making them accessible for use in various fields and on resource-constrained devices.

CONCLUSION

LoRA and QLoRA methods offer powerful tools for the efficient adaptation and deployment of large ML models. Their use can significantly reduce computational costs while maintaining high quality and accuracy of results. This opens new possibilities for applying ML in various fields, making it accessible even for resource-constrained devices.

In the future, LoRA and QLoRA methods are expected to continue to evolve, finding more applications in real-world scenarios. Their versatility and efficiency make them an essential component of the modern ML landscape, contributing to technological advancements and improving the quality of life.

REFERENCES

1. Mao Y., Ge Y., Fan Y., Xu W., Mi Y., Hu Z., Gao Y. A review of LoRa for large language models // arXiv: 2407.11046. 2024. P. 112-117.

2. Kong R., Li Q., Fang X., Feng Q., He Q., Dong Y., Wang W., Li Y., Kong L., Liu Y. LoRa-Switch: Improving the efficiency of dynamic LLM adapters through joint system and algorithm design // arXiv: 2405.17741. 2024. P. 151-157

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

4. Tyumentsev D. Application of cryptographic technologies for information security in cloud services // Stolypin Bulletin. 2024. Vol. 6. No. 3.

5. Inouye D., Lindo L., Lee R., Allen E. Applied automatic tuning to LoRa hyperparameters. 2024. P. 221-230.

6. Zhang F., Pilanchi M. Rimanov Pre-trained LoRa for fine-tuning baseline models // arXiv: 2402.02347. 2024. P. 478-457.

7. Amazon launches new Echo, Echo Dot with clock and Echo Studio in India, rolls out new Alexa features // URL: https://www.digit.in/news/general/amazon-

launches-new-echo-echo-dot-with-clock-and-echo-studio-in-india-rolls-out-new-alexa-features-50368.html/amp/ (date of application 14.06.2024).

8. Bobunov A. Development of test automation methodologies in the financial sector: a comparative analysis of approaches in the USA, Europe and Asia // Cold Science. 2024. No. 2/2024. P. 61-70.

9. Israfilov A. Cybersecurity in the automotive industry: vulnerabilities and protection // Sciences of Europe. 2024. No. 145. P. 60-63.