Report Project INF582

HUANG Siyuan, LE QUANG Dung

***Abstract*—In this project, we addressed the News Articles Title Generation challenge by employing the LlaMA model. Leveraging QLoRA for fine-tuning, our approach demonstrated notable success in effectively addressing the task of title generation for news articles.**

I. Introduction

In today's digital age, with a large amount of information, the need to capture information effectively is becoming increasingly important. A good header will help readers grasp the main content of the news, saving them time. Writers also need an effective tool that generates headlines, thereby saving time and attracting readers. Therefore, tools capable of automatically generating titles are becoming increasingly important.

Along with the development of Natural Language Processing, many methods have been proposed to solve this challenge. Starting with the use of rule-based and templates, people have gradually used basic statistical models such as n-gram language models and Hidden Markov Models (HMMs). Although these methods are generally easy to implement and understand, they have limited flexibility and scalability, leading to failure to capture complex linguistic structures and dependencies.

With the development of deep learning, title generation methods based on deep learning models such as Recurrent Neural Network, Long short-term memory (LSTMs) are introduced, with the ability to capture long-range dependencies and complex linguistic structures, as well as like learn representations directly from data, avoiding the need for handcrafted features. However, these methods are susceptible to classic deep learning problems such as vanishing gradients, particularly with long sequences, as well as the use of large amounts of data. Another mechanism introduced to address issues like long-range dependencies and semantic coherence is the attention mechanism. Although the attention mechanism can capture long-range dependencies, it adds complexity to the model architecture and training process, and may increase computational overhead and training time.

Recently, methods using pretrained models to fine-tune specific problems are becoming increasingly effective. The idea of these methods is that we will pretrain on previous models, then use the parameters after training to fine-tune the specific problem. This method proves to be very effective when it can take advantage of information from known models, can use pretraining parameters for different problems, thereby significantly reducing training time. Typical models using this mechanism are BERT, GPT, for example.

In this project, we use the Llama model proposed in [2]. This model exploits the effectiveness of using few parameters of the LoRA fine-tuning model, while also minimizing memory usage thanks to Block-wise k-bit Quantization. The results achieved for the Title generation problem were quite good, with a pretty good score on the leaderboard in the challenge.

II. Background

A. News Articles Title Generation

Let us revisit the challenge at hand. Given the input x representing a textual document, specifically a new article, our objective is to produce a compelling and informative headline from x. This entails crafting a title that succinctly encapsulates the main theme while also fostering reader engagement.

From a broader perspective, the task of title generation can be seen as to text summarization, wherein the goal is to distill the essential content of the article. However, it is imperative that the title remains concise (typically confined to a single line) yet possesses sufficient allure to captivate the reader's interest.

B. Low Rank Adaptation (LoRA)

The LoRA technique presents a fine-tuning methodology designed to address the computational

challenges associated with the adjustment of numerous parameters, a common occurrence in conventional fine-tuning procedures. During the fine-tuning stage, the objective is to refine the pretrained parameter W by incorporating an incremental adjustment denoted as ΔW. However, this adaptation often encounters issues related to computational complexity. In response, the LoRA approach endeavors to decompose ΔW into more manageable components. Grounded in the Intrinsic Rank Hypothesis, which states that substantial alterations to the neural network can be effectively captured using a lower-dimensional representation, LoRA proposes representing ΔW as the product of two matrices of reduced dimensions, A and B, both possessing a lower rank. Consequently, the updated weight matrix W' is expressed as: $W' = W + BA$.

Numerous experiments have demonstrated several advantages of LoRA. Specifically, LoRA mitigates memory requirements by diminishing the quantity of parameters necessitating updates, facilitating the handling of extensive-scale models. Furthermore, by streamlining computational demands, LoRA expedites the training and fine-tuning processes of large-scale models for novel tasks. Additionally, the reduced parameter count inherent to LoRA facilitates the fine-tuning of substantial models even on less powerful hardware configurations, such as modest GPUs or CPUs.

III. Methodology/Approach

A. QLoRA fine tuning

QLoRA, as described in [5], is an efficient fine-tuning technique for Large Language Models (LLMs), adept at diminishing memory consumption while upholding the efficacy of complete 16-bit fine-tuning. The method involves propagating gradients through a static, 4-bit quantized pre-trained model to Low Rank Adapters (LoRA). These LoRA adapters, integrated into every layer of the Transformer architecture, consist of trainable rank decomposition matrices designed to curtail the number of trainable parameters pertinent to downstream tasks.

The essential components of QLoRA includes:

- 4-bit NormalFloat (NF4) Quantization: This method utilizes the NormalFloat data type, lead to zero-centered normally distributed data, to achieve optimal quantization.

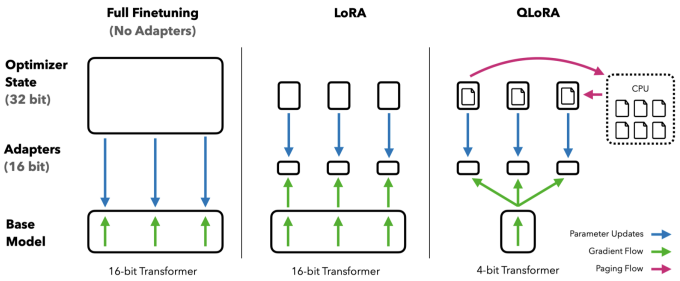

Fig. 1. The difference in memory requirement among fine-tuning methods. Figure from [5].

- Double Quantization: This technique facilitates further memory conservation by employing a double quantization strategy. Through this approach, additional memory savings are achieved, contributing to enhanced efficiency in memory utilization.
- Paged Optimizers: Paged Optimizers involve the swapping of optimizer states between the Central Processing Unit (CPU) and Graphics Processing Unit (GPU) using NVIDIA unified memory. This mechanism serves to mitigate GPU out-of-memory errors by intelligently managing memory resources and ensuring seamless transitions between CPU and GPU, thereby optimizing memory usage and preventing potential memory-related issues.

As described in [5], fine-tuning with QLoRA on a compact yet high-quality dataset yields state-of-the-art outcomes, surpassing prior benchmarks even with the utilization of smaller model architectures compared to the previous state-of-the-art models.

B. Algorithm Setup

In this project, we follow the steps described in [4]. The steps in the algorithm are summarized as follows:

1) Set up environment, load the dataset.
2) Loading and fine-tuning the model with QLoRA: the model is loaded in 4-bit NormalFloat (NF4) quantization with double quantization, and then trained using the SFTTrainer.
3) Running inference using the fine-tuned model. We use the fine-tuned model to generate titles for dialogues from the dataset.
4) Merging and saving the fine-tuned model: After training, the adapter model is saved. The

LoRA adapter and the base model are merged and saved together in a specified output folder.

5) Merging and saving the fine-tuned model: After training, the adapter model is saved. The LoRA adapter and the base model are merged and saved together in a specified output folder.
6) Running inference using the merged model: We use the merged model to generate summaries for dialogues from the given dataset. A pipeline is set up for title generation using the merged model, and then inference is run on a dialogue sample from the dataset. The generated title is compared to the ground truth title.

The prompt serves as a crucial guiding framework for generation tasks, directing the AI model's focus towards generating content that aligns with the specific instructions or context provided. Our prompt used in the training and inference process is shown below:

```
def prompt_formatter(sample):
    return f"""<s>### Instruction:
You are a helpful, respectful and honest assistant. \
Your task is to give the passage a title that fits the main idea. \
Your answer should be based on the provided passage only. \

### passage:
{sample['text']}

### title:
{sample['titles']} </s>"""
```

Listing 1. Our prompt formatting for llama2

The model underwent training for 2 epochs with a batch size of 4, incorporating gradient accumulation every 2 steps to effectively manage computational resources and memory. Model checkpoints are saved after each epoch. A learning rate of 2×10^{-4} was implemented, in conjunction with the "paged_adamw_32bit" optimizer, which is tailored for the efficient processing of large-scale models. Mixed precision training was enabled, with BF16 activated and FP16 disabled, while TF32 was kept active to strike a balance between computational speed and precision. To prevent exploding gradients, the maximum gradient norm was restricted to 0.3, and a warmup ratio of 0.03 was applied to gradually increase the learning rate at the start of training. The learning rate scheduler was set to a constant type, ensuring a stable learning rate throughout the training process.

IV. Results and Discussion

We train our model in a NVIDIA GF RTX 4090 GPU. The total training time is about 1 day. The inference time is about 2 hours. We discuss our result in several aspects.

A. ROUGE-L F-Score metrics

Briefly, this metric measures the precision and recall of the longest common subsequence (LCS) between the generated summary and the reference summaries. ROUGE-L F-Score is particularly useful for evaluating abstractive summarization systems, where the generated summaries may not match the reference summaries verbatim but should capture the main content and meaning effectively.

TABLE I
ROUGE-L F-Scores for LLaMA 2-7B Models

Model	ROUGE-L F-Score
LLaMA 2-7B (Fine-tuned)	0.21684
LLaMA 2-7B (Without Fine-tuning)	0.04783

In our project, we submit our results three times, and received the lowest score (without fine-tuning) of 0.04783, while these two other scores are nearly the same, about 0.21684 (with fine-tuning), as shown in Table I. In the training part in which we receive the lowest score, we have not had yet the result of fine-tuning score. This explain the efficacy of the fine-tuning process.

An additional noteworthy observation pertains to the disparity in language between the generated titles across different models. Specifically, the model exhibiting the lowest score yields titles in English, while titles generated by other models are in French. This variance in linguistic output may significantly influence the disparity in scores observed during the training phase. Moreover, it suggests that while the training process may effectively grasp the semantic nuances of the texts, the subsequent fine-tuning process plays a crucial role in discerning and aligning with the appropriate language context within the texts.

B. Title length

To enhance comprehension of our generated titles, an analysis of their length was conducted. The average length of titles produced by the models with higher scores approximates 30, aligning closely with the average length observed in the training dataset. Conversely, the model with a lower score yields an average title length of approximately 18. Consequently, our fine-tuned model adeptly replicates the title lengths observed within the training set.

V. Conclusion

This project has demonstrated the significant impact of fine-tuning on the performance of the LLaMA 2-7B model in the context of generating titles for **French** News. Through our training and fine-tuning process, including the utilization of techniques such as QLoRA, we have achieved notable improvements in the ROUGE-L F-Score metrics. The fine-tuned model produced titles with greater semantic alignment to the provided passages, resulting in more accurate and contextually relevant title generation. The enhancements observed in model performance underscore the efficacy of fine-tuning, particularly in tasks requiring nuanced understanding and generation of language.

References

[1] Sutton and Barto. Reinforcement Learning, *MIT Press*, 2020.
[2] Llama model, Hugging Face. Website here
[3] Understanding LoRA — Low Rank Adaptation For Finetuning Large Models, Bhavin Jawade. Website here
[4] How to fine-tune, compile and serve Llama2 7B-chat for summarization on a single GPU (12GB), Limin Ma. Website here
[5] QLORA: Efficient Finetuning of Quantized LLMs, Tim Dettmers et al. Website here