ADL 2024 HW3 Report

陳彥廷，資管四，b10705013

Q1: LLM Tuning

Q1-1 : Describe

- How much training data did you use?

I use 8000 examples from the training set.

- How did you tune your model?

1. download pre-trained model `zake7749/gemma-2-2b-it-chinese-kyara-dpo` and load it weights.
2. add QLoRA architecture to the model.
3. design the prompt and fine-tune the model based on the training set.

- What hyper-parameters did you use?

Hyper Parameter	Value
Optimization Algorithm	paged_adamw_32bit
lr_scheduler_type	constant
Learning Rate	2e-4

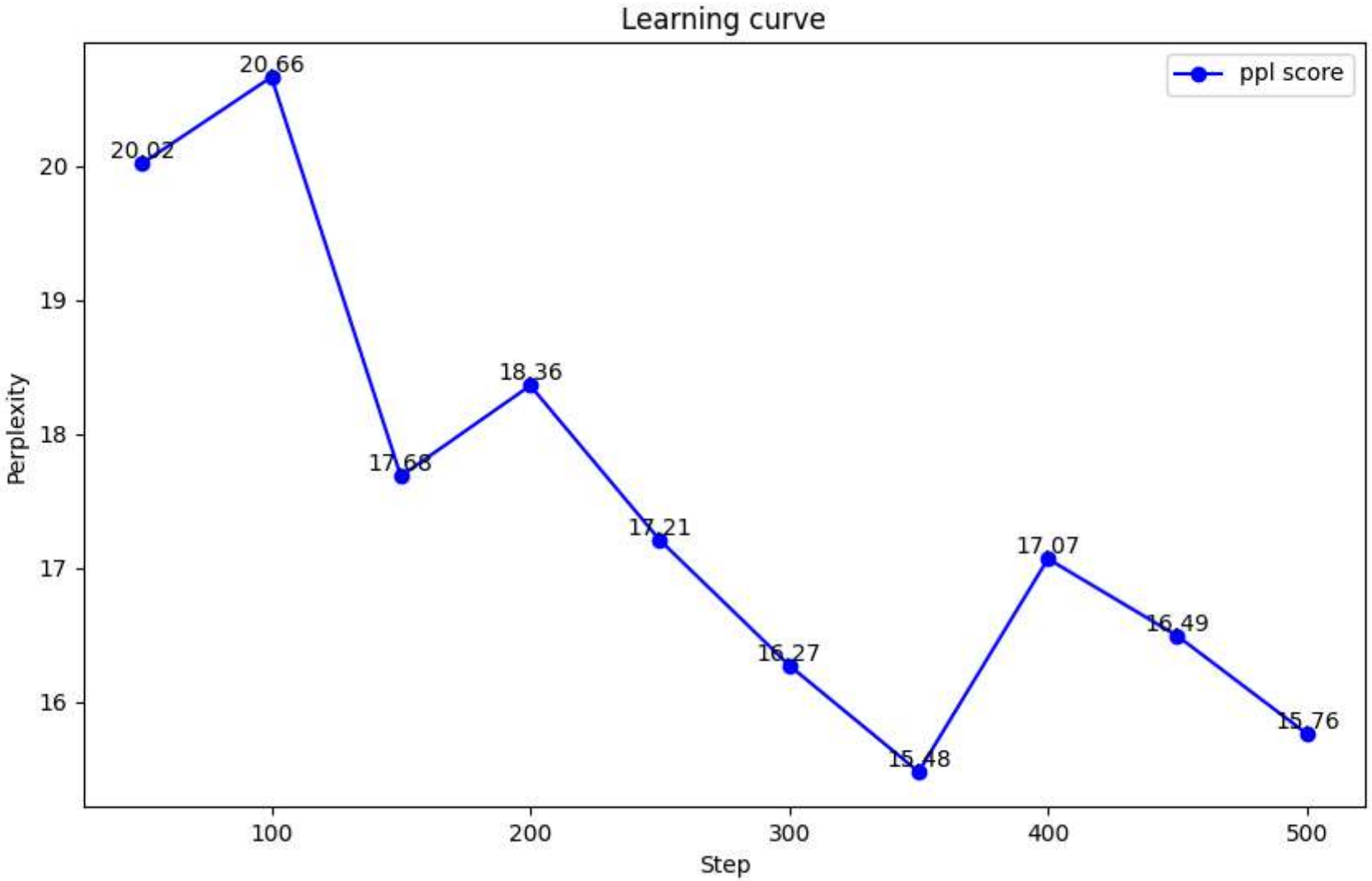
Hyper Parameter	Value
source_max_len	360
target_max_len	256
compute_dtype	torch.bfloat16
Batch Size per Device	3
Gradient Accumulation Steps	4
Batch Size	12
Step	350

Q1-2 : Show your performance

- What is the final performance of your model on the public testing set?

`15.479127301216126` on the public testing set.

- Plot the learning curve on the public testing set


Learning curve
ppl score
20.02
20.66
17.68
18.36
17.21
16.27
15.48
17.07
16.49
15.76
Perplexity
Step
20
19
18
17
16
100
200
300
400
500

Q2: LLM Inference Strategies

Q2-1 : Zero-Shot

- What is your setting?

In the Zero-Shot setting, we let the model perform inference directly on the task without providing any examples. The model relies solely on the knowledge it learned during pre-training to make predictions.

- How did you design your prompt?

```
def get_prompt(instruction: str) -> str:
    return f"你是一個對文言文和白話文非常了解的文學專家，以下是你跟用戶之間的對話，你要依據用戶的提示將內容翻譯成文言文或白話文。 用戶: {instruction}
```

Q2-2 : Few-Shot (In-context Learning)

- What is your setting? How did you design your prompt?

In the Few-Shot setting, I provide a small number of examples to help the model understand the format and requirements of the task. These examples follow the prompt closely, allowing the model to learn the instruction pattern.

```
def get_prompt(instruction: str) -> str:
    return (
        f"你是一個對文言文和白話文非常了解的文學專家，以下是你跟用戶之間的對話，你要依據用戶的提示將內容翻譯成文言文或白話文。 用戶: {instruction} 文
        f"用戶: 翻譯成文言文：\n雅裏惱怒地說： 從前在福山田獵時，你誣陷獵官，現在又說這種話。\n答案："
        f"文學專家: 雅裏怒曰： 昔畋於福山，卿誣獵官，今復有此言。"
        f"用戶: 沒過十天，鮑泉果然被拘捕。\n幫我把這句話翻譯成文言文"
        f"文學專家: 後未旬，果見囚執。"
        f"用戶: 辛未，命吳堅為左丞相兼樞密使，常楙參知政事。\n把這句話翻譯成現代文。"
        f"文學專家: 初五，命令吳堅為左承相兼樞密使，常增為參知政事。"
        f"用戶: 十八年，奚、契丹侵犯邊界，以皇上為河北道元帥，信安王為副，率禦史大夫李朝隱、京兆尹裴亻由先等八總管討伐他們。\n翻譯成文言文："
        f"文學專家: 十八年，奚、契丹犯塞，以上為河北道元帥，信安王禕為副，帥禦史大夫李朝隱、京兆尹裴伷先等八總管兵以討之。"
        f"用戶: 正月，甲子朔，鼕至，太後享通天宮；赦天下，改元。\n把這句話翻譯成現代文。"
        f"文學專家: 聖曆元年正月，甲子朔，鼕至，太後在通天宮祭祀；大赦天下，更改年號。"
        f"用戶: {instruction} 文學專家: "
    )
```

- How many in-context examples are utilized? How you select them?

I use 5 examples to help the model understand the format and requirements of the task. These examples are selected from the top 5 examples in the training set.

Q2-3 : Comparison

- What's the difference between the results of zero-shot, few-shot, and LoRA?

Three prompts are as follows:

```python
def get_prompt(instruction: str) -> str:
    """Format the instruction as a prompt for LLM."""
    # LoRA
    return f"你是人工智慧助理，以下是用戶和人工智能助理之間的對話。你要對用戶的問題提供有用、安全、詳細和禮貌的回答。USER: {instruction} ASSISTANT:"

    # Zero-shot
    return f"你是一個對文言文和白話文非常了解的文學專家，以下是你跟用戶之間的對話，你要依據用戶的提示將內容翻譯成文言文或白話文。 用戶: {instruction}

    # Few-shot
    return (
        f"你是一個對文言文和白話文非常了解的文學專家，以下是你跟用戶之間的對話，你要依據用戶的提示將內容翻譯成文言文或白話文。 用戶: {instruction} 文
        f"用戶: 翻譯成文言文：\n雅裏惱怒地說： 從前在福山田獵時，你誣陷獵官，現在又說這種話。\n答案："
        f"文學專家: 雅裏怒曰： 昔畋於福山，卿誣獵官，今復有此言。"
        f"用戶: 沒過十天，鮑泉果然被拘捕。\n幫我把這句話翻譯成文言文"
        f"文學專家: 後未旬，果見囚執。"
        f"用戶: 辛未，命吳堅為左丞相兼樞密使，常楙參知政事。\n把這句話翻譯成現代文。"
        f"文學專家: 初五，命令吳堅為左承相兼樞密使，常增為參知政事。"
        f"用戶: 十八年，奚、契丹侵犯邊界，以皇上為河北道元帥，信安王為副，率禦史大夫李朝隱、京兆尹裴亻由先等八總管討伐他們。\n翻譯成文言文："
        f"文學專家: 十八年，奚、契丹犯塞，以上為河北道元帥，信安王禕為副，帥禦史大夫李朝隱、京兆尹裴伷先等八總管兵以討之。"
        f"用戶: 正月，甲子朔，鼕至，太後享通天宮；赦天下，改元。\n把這句話翻譯成現代文。"
        f"文學專家: 聖曆元年正月，甲子朔，鼕至，太後在通天宮祭祀；大赦天下，更改年號。"
        f"用戶: {instruction} 文學專家: "
    )
```

The performance of the three prompts is as follows (Best score in 500 steps):

Model	Performance
LoRA	25.42

Model	Performance
Zero-shot	15.47
Few-shot	19.05

According to the results, the performance of the Zero-shot setting is the best, followed by the Few-shot setting, and the LoRA setting is the worst.

By the way, I found that sometimes the result of few-shot will just repeat the example, so I think that is why the performance of few-shot is not as good as zero-shot.

Reference

- ChatGPT
- Copilot