ADL-HW3

Q1

Model Description:

- Training data:
  - 資料量：考量到我使用 Colab Pro 提供的 A100 進行訓練，運算效率應該尚可接受，因此我使用了全部的 Training Data 做訓練，以求能達到最大的訓練成效。
  - 資料切割：我將資料以 9:1 的方式切分成 Training Data 跟 Validation Data.
- Model tuning:
  - 訓練方式與條整：我是根據 https://github.com/artidoro/qlora/blob/main/README.md 這個 Repository 中 Qlora 的實作，進行修改並訓練，而在參考過程中也有發現許多要調整的地方，包含：
    - Dataset Format: 本次使用的訓練資料會有我們自行設計過的Prompt，加上需要翻譯的文句，才是模型最終會得到的 input，因此會需要針對 Dataset 的 Format 進行修改。
    - Tokenizer : 本次作業使用的模型，其 Tokenizer 是使用 Fast Tokenizers，但範例中預設是將 use_fast 設定為 False, 因此需要修改。
    - 套件版本：範例與 Colab 預設的版本不同，在訓練時會遇到衝突，需要安裝指定的套件版本。
    - Prompt : 使用的 Prompt 如下

```
你是精通古今中文的翻譯助理，以下是用戶和翻譯助理之間的對話。
你要對用戶的問題提供詳細、精準的回答。將文言文翻譯成白話文，或白話文翻譯成文言文
這邊提供你兩個範例。
USER:翻譯成文言文：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
ASSISTANT:雅裏怒曰： 昔畋於福山，卿誣獵官，今復有此言。
USER: 能服信政，此謂正紀。翻譯成現代文：
ASSISTANT: 能守信於民，這叫作端正綱紀。
USER: {instruction} ASSISTANT:
```

  - 訓練迭代與參數：
    - 我透過嘗試數種參數去訓練，但本次作業並沒有嘗試太多種參數組合，大部分時間都在微調 prompt，而在訓練過程中可以發現幾點關於參數的觀察：
      - learning rate : 稍微的提高會有助於表現提升，但如果提升太高表現會迅速的下降。

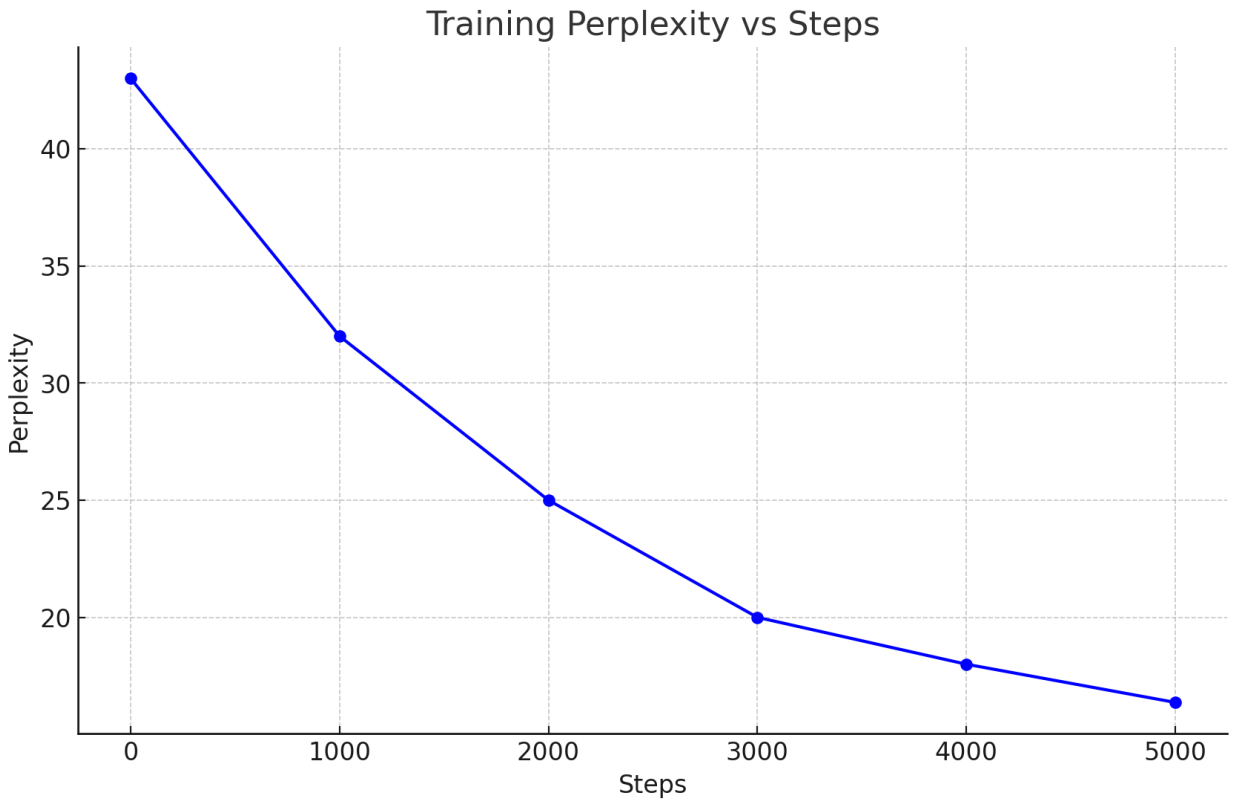
        - target max length : 設定為 256 以上會比較好，而增加多少不會影響到模型，推測是因為本次 output 都沒有超過這個長度。
        - max steps : 相較於 Llama 這種大模型，本次作業的模型需要多一點的迭代次數，我最終使用了 5000 Steps 來訓練。
        - prompt :
            - 若提及「翻譯任務」、「中文能力極強的助理/翻譯人員」，會有助於提升表現，但效果不大，整體訓練的 Perplexity 會從 36 進步到 34.21 左右。
            - 若提及「回答只能有中文」「不可以有 \n」，Perplexity 可以進一步降低到約 30.31，但仍然可能會出現英文或符號的回答。
            - 若使用 Few shot 的概念在 prompt 中提供幾個範例，fine-tune 效果會好上很多，Perplexity 可以降低到 16.37。
- Hyper-parameters:
    - 我使用的超參數如下：

max_steps	5000	per_device_eval_batch_size	8
learning_rate	0.0008	lora_r	64
per_device_train_batch_size	8	lora_alpha	16
lora_dropout	0.05	bits	4
quant_type	nf4	use_double_quant	True
target_max_len	512	source_max_len	1024
eval_dataset_size	1024 (~10%)	lr_scheduler_type	constant
gradient_accumulation_steps	16	weight_decay	0.0

Performance:

- Final performance:
    - 模型最好的表現是 Perplexity = 16.37。
- Learning curve:

附圖是有含範例的 prompt 的訓練結果，每一千個 step 做一個 checkpoint，並且用該 checkpoint 對 public_test 的產出做 evaluation：



Q2

Inference Strategies

Zero-shot 與 Few-shot 的主要差異在於是否有沒有在 prompt 中提供幾個提示的範例，無論是從近期研究的文獻或是直覺上來想，都會覺得 Few-shot 應該能讓模型對任務有更好的了解。

但應該仍然比不過有針對任務資料進行 LORA 過的版本。下面進行了 Zero-shot 與 Few-shot 的比較，並驗證了我的猜想。

以下模型的訓練皆同樣使用 Q1 提到的超參數。

- Zero-Shot
  - What is your setting? How did you design your prompt? (1%)

    Zero-Shot 的部分，我使用了作業預設的 prompt，並且再使用了自己修正過的 prompt：

    ```
    你是中文能力極高的助理，以下是用戶和助理之間的對話。
    你要對用戶的問題提供有用、詳細並且精準翻譯的回答。
    ```

```
以下的問題為文言文翻譯成白話文或白話文翻譯成文言文，請回答：
USER: {instruction} ASSISTANT:
```

與預設的差別在於，我更換了助理的角色，並給他一個更明確的任務要求，同時也請他用精準的、中文很強的標準去看待翻譯問題，來提升他的精準度，然而實驗數次後發現並沒有明顯的差別，並且因為沒有經過 fintune，模型產出的回答 Perplexity 很高，約為 3512.43。

- Few-Shot (In-context Learning)
  - What is your setting? How did you design your prompt? (1%)

    修改了一些用語，並添加了兩個例子，Prompt 的內容如下：

    ```
    你是精通古今中文的翻譯助理，以下是用戶和翻譯助理之間的對話。
    你要對用戶的問題提供詳細、精準的回答。將文言文翻譯成白話文，或白話文翻譯成文言文
    這邊提供你兩個範例。
    USER:翻譯成文言文：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
    ASSISTANT:雅裏怒曰： 昔畋於福山，卿誣獵官，今復有此言。
    USER: 能服信政，此謂正紀。翻譯成現代文：
    ASSISTANT: 能守信於民，這叫作端正綱紀。
    USER: {instruction} ASSISTANT:
    ```

  - How many in-context examples are utilized? How you select them? (1%)
    - 數量：我總共選了兩個，一個是文言文翻譯成白話文，另一個是白話文翻譯成文言文。
    - 挑選方式：例子是隨機挑選的。
- Comparison:
  - What's the difference between the results of zero-shot, few-shot, and LoRA? (2%)

LoRA (No Examples)	30.31
LoRA (With Examples)	16.37
Few-shot	547.86
Zero-shot	3512.43

    - 整體的表現而言， LoRA > few-shot > zero-shot.
      而這也驗證了一開始提到的假設，few-shot 提供了幾個模型幾個示範，確實有助於模型理解任務。
    - LoRA 的表現最好，若沒有搭配例子去做 Fine-tune，Perplexity 會是約 30.31
    - 若搭配有例子的 prompt, Perplexity 可以降到 16.37.

Q3

Llama3-Taiwan (8B)

訓練設定：

- 資料量：考量到他的模型大小相對於本次作業大很多，訓練資訊只選用了前五千筆
- 訓練方式：與作業一樣進行 QLoRA.
- 超參數：與上述相同，除了 max_steps 為 1000。

成果表現：

- Llama3 的表現比起 gemma 要好滿多的，Public test 的資料 Perplexity 僅有大約 7.3172，且他的訓練 Steps 比起 gemma 還少，由此可知模型本身的大小與 pre-train 的程度還是有差的。