ADL HW3 Report

資工三 B10705005 陳思如

Q1 LLM Tuning

1. how much training data?

   I use 9500 data in training data set.

   It is because I set the val_set_size = 0.05. The axolotl will randomly split the training data set into the validation set based on the val_set_size we set.

2. How did I tune the model?

   I have done some preprocess on the training data before the finetune. I set the instruction and output into similar format.

   instruction = "請幫我把下列語句翻譯成OO文。{要翻譯的話}答案："

   (OO文 is based on the original instruction prompt.

   These are the OO文 options: 白話文/現代文/古文/文言文/中國古代的話)

   output = "{翻譯完成的答案}"

   I believe that the training data with similar format helps the model to learn more precisely and efficiently.

   Then, I use the `axolotl` package to finetune the Taiwan-Llama with qlora. I train with `axolotl` with the following command:

   ```
   accelerate launch --num_processes 1 -m axolotl.cli.train examples/llama-2/qlora.yml
   ```

3. What hyperparameters?

   ```
   val_set_size: 0.05
   load_in_4but=true
   sequence_len: 2048
   sample_packing: true
   pad_to_sequence_len: true
   lora_r: 8
   lora_alpha: 16
   lora_dropout: 0.05
   lora_target_linear: true
   gradient_accumulation_steps: 4
   micro_batch_size: 2
   num_epochs: 4
   optimizer: paged_adamw_32bit
   lr_scheduler: cosine
   learning_rate: 0.0002
   bf16: true
   warmup_steps: 10
   weight_decay: 0.0
   special_tokens:
     bos_token: "<s>"
     eos_token: "</s>"
     unk_token: "<unk>"
   ```

4. Performance on public testing set?

   I have runned the finetune for 4 epochs and I have saved the adapter model for each epoch. I have evaluated each adapter checkpoint perfomance with running public testing dataset with the ppl. Moreover, since I have preprocessed the data set before the finetune, I have two kinds of ppl score that can represent the performance.

   Here is the result:

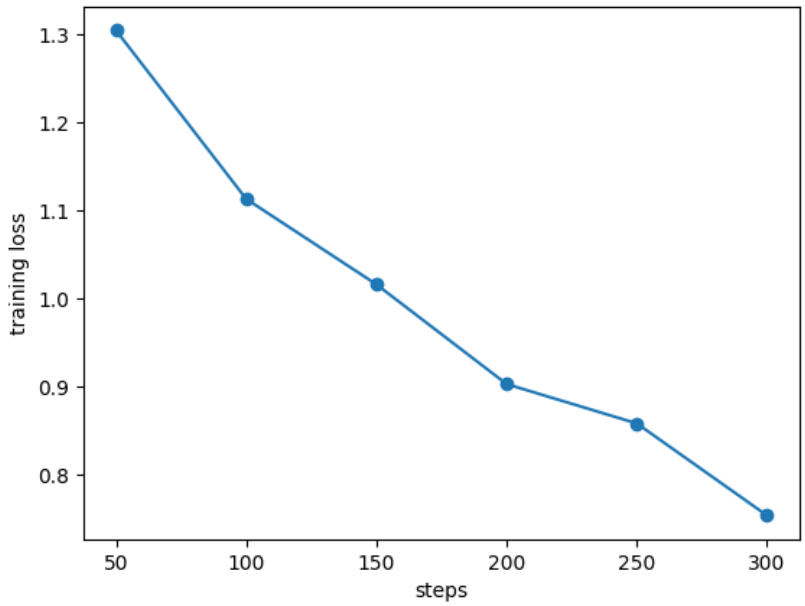
epoch	1	2	3	4
processed dataset ppl score	3.67823	**3.46591**	3.66788	3.90255
unprocessed dataset ppl score	3.70658	3.50497	3.71518	3.952907

   I have found out that epoch2 has the best result. This also indicates that the preprocess is quite important if I wanted to use the the trained model with preprocessed data.

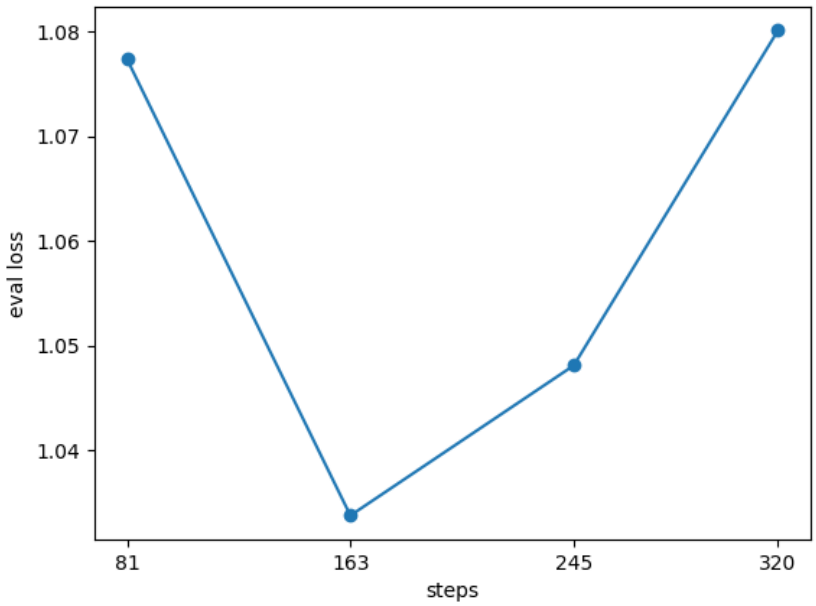
5. Learning curve on public testing set?

   I used the `axolotl` default train and validation split method. This splits the training dataset and doesn't pass the public testing set as validation dataset.

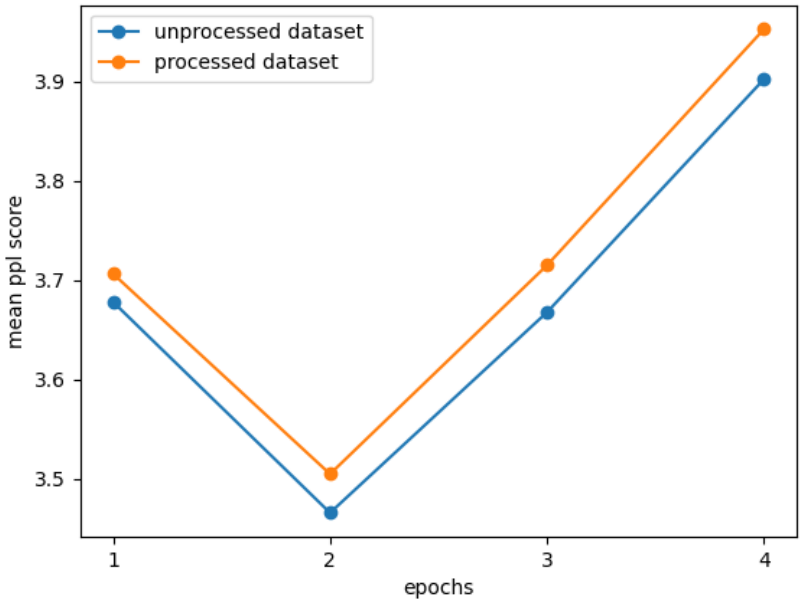
   Here is the learning curve of the train loss on training data itself:



   Here is the learning curve of loss on train split evaluation data:



Here is the learning curve of the ppl score on public testing data:



Q2 LLM Inference

1. Zero Shot?

   I just used the preprocessed public testing dataset. The ppl score = 7.070909

   ```
   instruction = "請幫我把下列語句翻譯成OO文。{要翻譯的話}答案："

   output = "{翻譯完成的答案}"
   ```

2. Few Shot?

   I have tried with 1-shot, 3-shot and 8-shot.

   First I preprocessed the instruction as the usual.

```
instruction = "請幫我把下列語句翻譯成OO文。{要翻譯的話}答案："

output = "{翻譯完成的答案}"
```

Then based on the shot number, I inserted numbers of data into the front of the instruction. And these datas are from the preprocessed train dataset and also based on the same translation language.

```
instruction =
    "請幫我把下列語句翻譯成OO文。{要翻譯的話}答案：{翻譯完成的答案}\n" * shot_num
//from train
    + "以上是之前的翻譯紀錄，現在" //concat to public test instruction
    + "請幫我把下列語句翻譯成OO文。{要翻譯的話}答案：" //public test instruction
to ask

output = "{翻譯完成的答案}"
```

Here is the ppl score for different shot inference:

	1-shot	3-shot	8-shot
ppl score	7.43625	7.240722	7.357

So, I decided to use the 3-shot. I think 3-shot is better is that it can learn from some of the examples but not limited on too less information or bounded from the too much restrictions.

3. Comparison

	zero shot	few shot	qlora
ppl score	7.070909	7.240722	3.46591

We can see that the qlora has the best performance.

The Qlora has finetuned the llama model which makes the Llama model has another peft model to make the prediction much more fit on this traditional translation task. However, zero shot or few shot inference only used the original Taiwan-Llama model which is designed for general purpose instead of this only task. That is why Qlora has the best performance comparing with the other two.