**Generative Chatbot**

Farnoosh Javadi

Sept 17, 2023

**Fine-tuning Strategy:**

The finetuning is done with LORA based on the next word prediction task. During finetuning, model is seeing the whole conversation in the format of:

“#Farnoosh: Hello! #Hassan: Hello! How can I help you? #Farnoosh: I want to update my profile #Hassan: Sure!” and predicts the and predicts the next word. But in inference it sees the conversation until the final dialog and generates the final answer.

**NOTE:** since we have access to emotions as an additional step I also added emotion to each dialogue so that emotions are also considered during generation.

**Next Steps:** For inference I assumed that the emotion of final answer is given in the input (as the prompt), But the better approach was to first classify the dialogue emotion and then append the emotion to the conversation history for emotion-aware generation. But due to limited time I assumed we have the emotion prediction results.

**Implementation Steps:**

**1) Data parsing and Dataset Creation**

For parsing daily dialog dataset I changed the [original parser](https://github.com/Sanghoon94/DailyDialogue-Parser/blob/master/parser.py) in a way that it appends the speaker and the emotion to the dialogue too.

\*Note that for the test split I considered the last dialogue as the label of the whole conversation as it is what we expect from the model to generate.

A screenshot of a computer program

Description automatically generated

**Output:**

Len(dataset\_train) = 11118

Sample\_data = “The kitchen stinks . \_\_eou\_\_ I'll throw out the garbage . \_\_eou\_\_”

Sample\_data after parser = “#Farnoosh $disgust: The kitchen stinks. #Hassan $neutral: I'll throw out the garbage”

**2) Model Loading**

I used pretrained gpt2 from huggingface as my model since it has a reasonable number of params considering the small size of our fine-tuning dataset.

Note that I also tried Bloom-560m, and llama 7B too. For Bloom-560 the generation results were not as good as gpt2 as there were so much repetition. And as for lama-7B its finetuning taking so long so I decided to move forward with gpt2. Although the code is generalizable to any other AutoRegressive based model like gpt3, etc. and you can input the model\_name in the command.

A computer code with black text

Description automatically generated

**3) Creating finetuning config and trainer**

**For setting up the trainer there were some necessary steps:**

**1) defining LORA config**

**A screenshot of a computer code

Description automatically generated**

**2) defining trainer arguments**

**A screenshot of a computer program

Description automatically generated**

**3) Setting up the trainer:** I used SFTTrainer that gives a wrapper around transformers Trainer to easily fine-tune models on instruction based datasets using PEFT adapters.

**A computer code with text

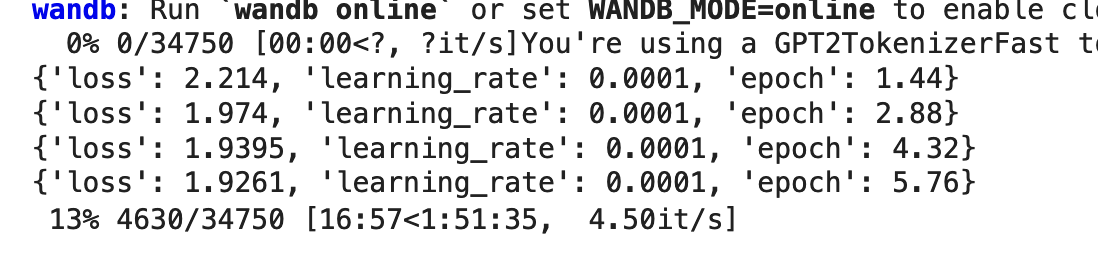
Description automatically generated**

**4) Train**

I could simple use trainer.train()

A black text on a white background

Description automatically generated

****

I wrote a script to plot loss\_curve:

A graph with a line

Description automatically generated

**5) Evaluation**

At first I needed to predict the next dialog per each data sample in training set:

A computer code with text

Description automatically generated

**Sample Prediction:** For more samples you can take a look at log.txt.

**Input:** “#Farnoosh $neutral: Hey man , you wanna buy some weed ? #Hasan $surprise: Some what ? #Farnoosh $neutral: Weed ! You know ? Pot , Ganja , Mary Jane some chronic ! #Hasan $neutral: Oh , umm , no thanks . #Farnoosh $neutral: I also have blow if you prefer to do a few lines . #Hasan $neutral: No , I am ok , really . #Farnoosh $neutral: Come on man ! I even got dope and acid ! Try some ! #Hasan $neutral: Do you really have all of these drugs ? Where do you get them from ? #Farnoosh $neutral: I got my connections ! Just tell me what you want and I ’ ll even give you one ounce for free . #Hasan $neutral: Sounds good ! Let ’ s see , I want . #Farnoosh $fear: Yeah ? #Hasan $neutral:“

**Answer:** I am not sure what you are talking about?

**Label:** I want you to put your hands behind your head ! You are under arrest !

Then I did the evaluation based on rouge and bleu score.

A screenshot of a computer code

Description automatically generated

**Results:**

**Rouge:** {'rouge1': 0.10297996132918898, 'rouge2': 0.023010147611108873, 'rougeL': 0.09625391605892789, 'rougeLsum': 0.09636512396510974}

**Bleu:** {'bleu': 0.012639909446031746, 'precisions': [0.2014664410603497, 0.02540766021994691, 0.009486847779215178, 0.0029930162953109413], 'brevity\_penalty': 0.647357765983805, 'length\_ratio': 0.6969339622641509, 'translation\_length': 8865, 'reference\_length': 12720}

