Fine-tuning Pretrained Language Models to Build Customized Chatbots

Abstract: This project is about fine-tuning a pretrained LLM for certain purpose, going through the process of building a chatbot. This project uses 2 different datasets to finetune the same model. I set up code for fine-tuning open-sourced pretrained LMs and created an interface to test and interact with the fine-tuned chatbot. Finally, I compare the 2 finetuned models' reaction to different problems.

1. **Overview**
   Large Language model is gaining huge popularity nowadays and chatgpt has changed people's life greatly. This project is to explore how to build a chatbot based on existing language models by fine-tuning.
   This topic is interesting in that LLM is trending model and how it works and why it is so powerful and can talk to people like human interests me.
   The whole process includes working with the Transformers library from Hugging Face, which provides pre-trained models, datasets, and tools and fine-tuning the model on 2 different datasets of conversational examples such as chat logs, several scenes of a movie and so on.
   Based on existing language models which are pre-trained, I then use chosen dataset to fine-tune the model to learn how people chat more naturally, thus making the output of the model more human-like.

2. **Experiment setup**
   For this project, I used 2 datasets which are "**wiki_movies**" and "**timdettmers/openassistant-guanaco**". Guannaco dataset is a subset of the Open Assistant dataset and only contains the highest-rated paths in the conversation tree, with a total of 9,846 sample. Wiki_movies dataset consists of roughly 100k (templated) questions over 75k entities based on questions with answers in the open movie database (OMDb) and it is the QA part of the Movie Dialog dataset.
   Also, because Wiki_movies dataset organized as "Question" and "Answer" Columns rather than simple "text" column as Guannaco, we also need to preprocess the Wiki_movies dataset before we can feed it to the model.
   For the implementation, I used model EleutherAI/gpt-neo-125m, and executed on local laptop equipped with 3060 Ti GPU.
   Gpt-neo-125m is a transformer model designed using EleutherAI's replication of the GPT-3 architecture. GPT-Neo refers to the class of models, while 125M represents the number of parameters of this particular pre-trained model. This model was trained on the Pile for 300 billion tokens over 572,300 steps. It was trained as a masked autoregressive language model, using cross-entropy loss. This way, the model learns an inner representation of the English language that can then be used to extract features useful for downstream tasks. The model is best at what it was pretrained for however, which is generating texts from a prompt.

3. **Experiment results**
   This project goes through the process of building a chatbot by finetuning a pretrained language model and uses 2 different datasets to test how well the finetuning performs for different datasets and what dataset is better for finetuning. I found that firstly after finetuning, the chatbot can answer simple questions and below is the picture of finetuning using the dataset "**timdettmers/openassistant-guanaco**".
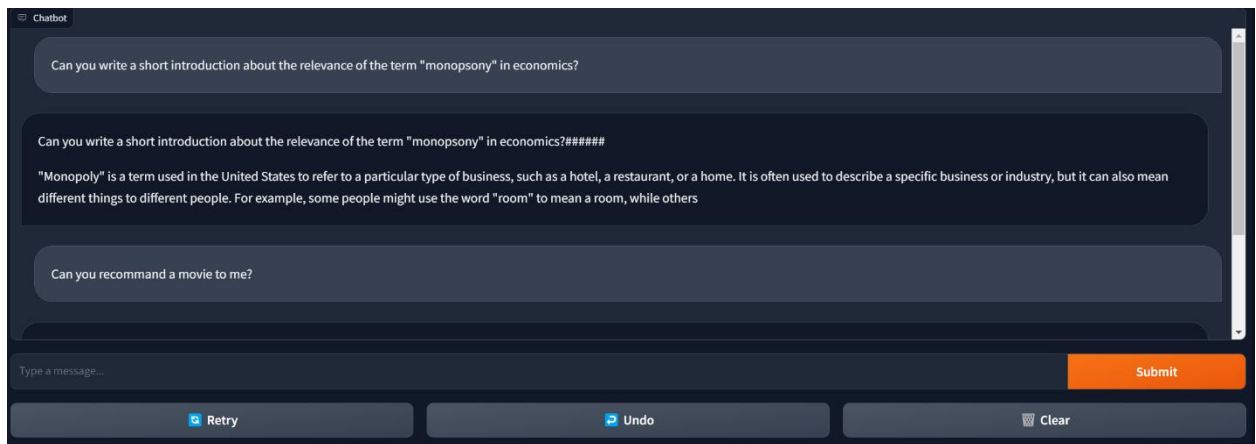
Secondly, for fine-tuning using different datasets, the model will be better at specific questions because of the fine-tuning dataset's property. Wiki_movies dataset is mainly about movies while Guanaco dataset is mainly about general dialog. Now let's see the response of 2 different models for the same problem: Can you recommend a movie to me?

Wiki_movies:
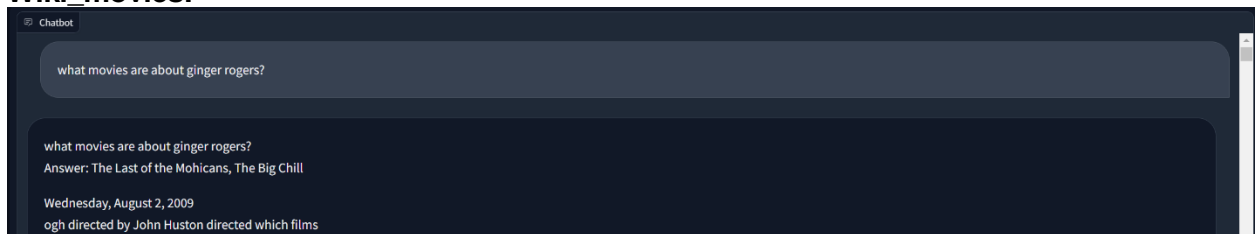
Openassistant-guanaco:
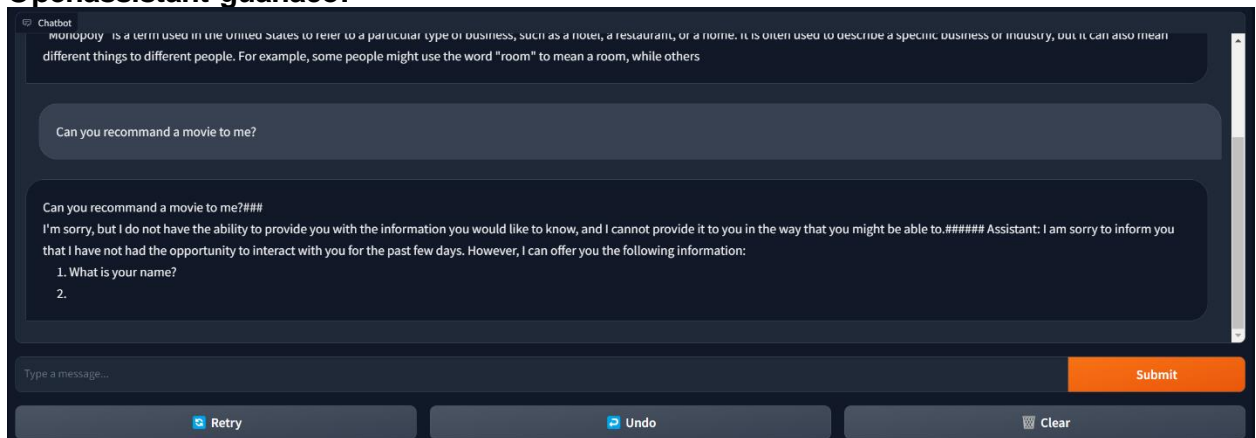
We can see from above 2 pictures and see that finetuning using Wiki_movies dataset makes the chatbot(model) better at answering problems related to movies.

Another interesting finding is that Because Wiki_movies dataset organized as Question and Answer Column and the answer part is not organized as well as the Guanaco dataset, which means it's more like answer rather than human-like conversation, the finetuning model of wiki_movies does not behave as human-like as Guanaco dataset's result.

Finally, for fine-tuning the models, I tried the learning rate from 0.005 to 1e-5 and finds out for this pretrained model, 0.005 is too big and the loss can not go below 5 while when we use learning rate 1e-5, it can go as low as 1.1. This makes me think that when

are training the model, maybe we can try adaptive learning rate to make the training process better.

4. **Discussion**

The result of training 2 different datasets gave me what I hoped for, proving the effects of finetuning the language model and makes me familiar with how to build a chatbot. It also proves that by finetuning we can make the model more and more knowledgeable and human-like. However, it also seems that what feed to the model is also very important. As shown above, when using wiki_movies dataset, because the answer is less organized as human-like conversation, the model will not answer questions as human-like as the model finetuned with more human-like dataset Guanaco.

5. **Conclusion**

This project goes through the process of finetuning a pretrained language model to build a chatbot that can contact with us. This project also shows the effect by finetuning using different datasets and tested the chatbot with same questions. What's more, this project also tested that the dataset itself is very important in that a more conversational and human-like dataset will make the chatbot more human-like. Finally, this project tuned parameters such as learning rate to optimize the training process and showed that adaptive learning rate will be helpful in this project.

References:

EleutherAI/gpt-neo-125m · Hugging Face

timdettmers/openassistant-guanaco · Datasets at Hugging Face

wiki_movies · Datasets at Hugging Face

Include any links, papers, blog posts, or GitHub repositories you have used here.