Fine-Tuning based on 2000 drug examples from an Excel file

DS565
Emily Weng, 20016

Step 1: Preparing the Data and Launching the Fine Tuning

```
import pandas as pd
```
✓ 0.4s

```
n = 2000
```
✓ 0.0s

```
df = pd.read_excel('Medicine_description.xlsx', sheet_name='Sheet1',
        header=0, nrows=n)
```
✓ 0.1s

```
reasons = df["Reason"].unique()
reasons_dict = {reason: i for i, reason in enumerate(reasons)}
df["Drug_Name"] = "Drug: " + df["Drug_Name"] + "\n" + "Malady:"
df["Reason"] = " " + df["Reason"].apply(lambda x: "" + str(reasons_dict[x]))
df.drop(["Description"], axis=1, inplace=True)
```
✓ 0.0s

```
df.rename(columns={"Drug_Name": "prompt", "Reason": "completion"}, inplace=True)
```
✓ 0.0s

```
jsonl = df.to_json(orient="records", indent=0, lines=True)
```
✓ 0.0s

```
with open("drug_malady_data.jsonl", "w") as f:
    f.write(jsonl)
```
✓ 0.0s

Step 2: Remember to set up OpenAI Key

Step 3: Get training data

OpenAI now uses chat format so I had to convert prompt format to chat in order to finetune

```python
import json

Tabnine | Edit | Test | Explain | Document | Ask
def convert_to_chat_format(input_file, output_file):
    """
    Convert a prompt-completion file to chat format.
    Args:
        input_file (str): Path to the input .jsonl file.
        output_file (str): Path to save the chat-formatted .jsonl file.
    """
    with open(input_file, "r") as infile, open(output_file, "w") as outfile:
        for line in infile:
            data = json.loads(line)
            chat_format = {
                "messages": [
                    {"role": "user", "content": data["prompt"].strip()},
                    {"role": "assistant", "content": data["completion"].strip()}
                ]
            }
            outfile.write(json.dumps(chat_format) + "\n")
    print(f"Converted {input_file} to chat format and saved as {output_file}")

# File paths for training and validation files
training_file = "drug_malady_data_prepared_train.jsonl"
validation_file = "drug_malady_data_prepared_valid.jsonl"

# Output paths for chat-formatted files
chat_training_file = "chat_formatted_train.jsonl"
chat_validation_file = "chat_formatted_valid.jsonl"

# Convert both files
convert_to_chat_format(training_file, chat_training_file)
convert_to_chat_format(validation_file, chat_validation_file)
```

✓ 0.0s

Step 4: Fine tune both of the files.

```
!openai api files.create -f "chat_formatted_train.jsonl" -p "fine-tune"
✓ 2.7s
Upload progress: 100%|| 216k/216k [00:00<00:00, 697kit/s]
{
  "id": "file-VTJQ8Dzu9HVienxu97uWQk",
  "bytes": 216118,
  "created_at": 1732336852,
  "filename": "chat_formatted_train.jsonl",
  "object": "file",
  "purpose": "fine-tune",
  "status": "processed",
  "status_details": null
}
```

```
!openai api files.create -f "chat_formatted_valid.jsonl" -p "fine-tune"
✓ 1.6s
Upload progress: 100%|| 54.0k/54.0k [00:00<00:00, 300kit/s]
{
  "id": "file-98YZjx4bFXuBjteouvPdBN",
  "bytes": 53966,
  "created_at": 1732336866,
  "filename": "chat_formatted_valid.jsonl",
  "object": "file",
  "purpose": "fine-tune",
  "status": "processed",
  "status_details": null
}
```

Step 5: Create Fine Tune Job

```python
response = openai.fine_tuning.jobs.create(
    training_file="file-VTJQ8Dzu9HVienxu97uWQk",
    validation_file="file-98YZjx4bFXuBjteouvPdBN",
    model="gpt-3.5-turbo",
    suffix="drug_malady_model"
)

print(f"Fine-tuning job created: {response}")
```

✓ 2.2s Python

```
Fine-tuning job created: FineTuningJob(id='ftjob-yY0zyaj5uNHJmdGRGRlGf8tV', created_at=1732336896, error=Error(code=None, mess
```

Step 6: Wait for the fine tuning to finish

```
import time
fine_tune_job_id='ftjob-yY0zyaj5uNHJmdGRGRlGf8tV'
while True:
    job_status = openai.fine_tuning.jobs.retrieve(fine_tune_job_id)
    print(f"Status: {job_status.status}")

    if job_status.status in ["succeeded", "failed"]:
        break
    time.sleep(30)  # Wait 30 seconds before checking again
```

✓ 42m 20.1s

```
Status: validating_files
Status: validating_files
Status: running
Status: running
Status: running
Status: running
Status: running
```

Step 7: Get job detail

```
job_id = "ftjob-yY0zyaj5uNHJmdGRGRlGf8tV"  # Replace with your fine-tuning job ID
job_details = openai.fine_tuning.jobs.retrieve(job_id)
print(f"Fine-Tuned Model ID: {job_details.fine_tuned_model}")
```

✓ 0.2s

```
Fine-Tuned Model ID: ft:gpt-3.5-turbo-0125:personal:drug-malady-model:AWcuggcr
```

Step 8: Test results:

```python
response = openai.chat.completions.create(
    model="ft:gpt-3.5-turbo-0125:personal:drug-malady-model:AWcuggcr",  # Replace with your fine-tuned model ID
    messages=[
        {"role": "user", "content": "What is the recommended drug for pain relief?"}
    ]
)
print(response.choices[0].message.content)
```

✓ 9.5s Python

```
The recommended drug for pain relief varies depending on the type and severity of the pain. Common over-the-counter options in
```

Github Link:

https://github.com/emilywengster/sfbu/tree/d8a5fa44caf7220fae637f62b5669eeea21cee7f/Generative%20AI/Fine-Tuning/2000%20Drug%20Examples