UDACITY

**Introduction to Generative AI with AWS**

**Project Documentation Report**

Visit [UDACITY Introduction to Generative AI with AWS Project Documentation Report] to make a copy of this document.

Complete the answers to the questions below to complete your project report. Create a PDF of the completed document and submit the PDF with your project.

Question Your answer:

Step 2: Domain Choice

What domain did you choose to fine-tune the Meta Llama 2 7B model on?

Choices:

1. Financial
2. Healthcare
3. IT Financial

Step 3: Model Evaluation Section

What was the response of the model to your domain-specific input in the model_evaluation.ipynb file?

1)

```
payload = {
    "inputs": "Traditional approaches to data management such as",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}
try:
    response = finetuned_predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The investment tests performed indicate a promising trend in the financial performance, showing a steady increase in returns and a reduction in risk expo
sure. These results suggest that the current investment strategy is effective and may yield significant benefits if maintained. Further analysis and moni
toring are recommended to ensure continued growth and to adapt to any potential market changes.
==================================
```

2)

```
# Define the payload with the combined prompt
payload = {
    "inputs": "the relative volume for the long out of the money options, indicates.",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}

try:
    response = predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
the relative volume for the long out of the money options, indicates
> 20% of the options available in this strike are held by speculators.
The long out of the money call options are trading at a 62.45% premium relative to the at the money call option. This indicates a high level of demand for the long out of the money c
all
```

3)

```python
# Define the payload with the combined prompt
payload = {
    "inputs": "The results for the short in the money options ",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}

try:
    response = predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The results for the short in the money options
> 15, 10, and 5 are as follows:
Short 15 1500 1200 0 0 0 1200 1500
Short 10 1200 1000 0 0
```

4)

```python
payload = {
    "inputs": "The results are encouraging for aggressive investors.",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}

try:
    response = predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The results are encouraging for aggressive investors.
>
The average stock in the index is down 37% year-to-date, but the best performing stocks are up 20% and the worst performers are down 47%.
It’s important to note that the index is heavily weighted to stocks that are down more than
```

Step 4: Fine-Tuning Section
After fine-tuning the model, what was the response of the model to your domain-specific input in the model_finetuning.ipynb file?

1)

```
payload = {
    "inputs": "The results are encouraging for aggressive investors",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}
try:
    response = finetuned_predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The results are encouraging for aggressive investors with the opportunity to learn of split announcements early and acquire option positions quickly. App
arently the downside risk is minimal and the short-run returns can be quite good. For investors who can buy during day -1, the mean of the range generall
y exceeded a 10 percent return in four days. However, this might be considered evidence against the semi-strong efficient market hypothesis due to the in
formation requirements and time constraints involved.

==================================
```

2)

```
payload = {
    "inputs": "the relative volume for the long out of the money options, indicates",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}
try:
    response = finetuned_predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The relative volume for the long out of the money options indicates a spike starting with day -1, a peak on day 0 and the highest relative volume on day
+3 followed by generally declining values. This is the only instance where volume on day +3 is the highest value. Strong volume on day +3 is consistent w
ith the price series peaks on day +3.

==================================
```

3)

```
payload = {
    "inputs": "The investment tests performed indicate",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}
try:
    response = finetuned_predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The investment tests indicate whether an investor could take advantage of the price movements in the options of stocks announcing splits. The analysis is
in two parts: 1) The cumulative percent change for the various cross sectional series, and 2) The specific returns to an investor who acquired the "typic
al" option during this period and paid commissions to buy and sell.

==================================
```

Do the outputs from the fine-tuned model provide domain-specific insightful and relevant content? You can continue experimenting with the inputs of the model to test it's domain knowledge.

4)

```
payload = {
    "inputs": "The results for the short in the money options",
    "parameters": {
        "max_new_tokens": 64,
        "top_p": 0.9,
        "temperature": 0.6,
        "return_full_text": False,
    },
}
try:
    response = finetuned_predictor.predict(payload, custom_attributes="accept_eula=true")
    print_response(payload, response)
except Exception as e:
    print(e)
```

```
The results for the short in the money options showed negative returns the first day, a peak cumulative value on day 3 of 6.305, one subsequent negative value and an ending value of 6.201. The relative series (split minus match) was never negative, it peaked at 11.259 percent on day 3 and finished at over 8 percent.

==================================
```

| Question | Your answer: |
|---|---|
| **Step 2: Domain Choice**<br>What domain did you choose to fine-tune the Meta Llama 2 7B model on?<br>Choices:<br>1. Financial<br>2. Healthcare<br>3. IT | Financial |
| **Step 3: Model Evaluation Section**<br>What was the response of the model to your domain-specific input in the **model_evaluation.ipynb file**? | PFA |
| **Step 4: Fine-Tuning Section**<br>After fine-tuning the model, what was the response of the model to your domain-specific input in the **model_finetuning.ipynb file**? | PFA |