This week I did an implementation of code in both Python and Java. The code is supposed to get authorised via Google to VertexAI APIs and by using generated prompts, provide proper answers to every question.

Code is pushed in Github:

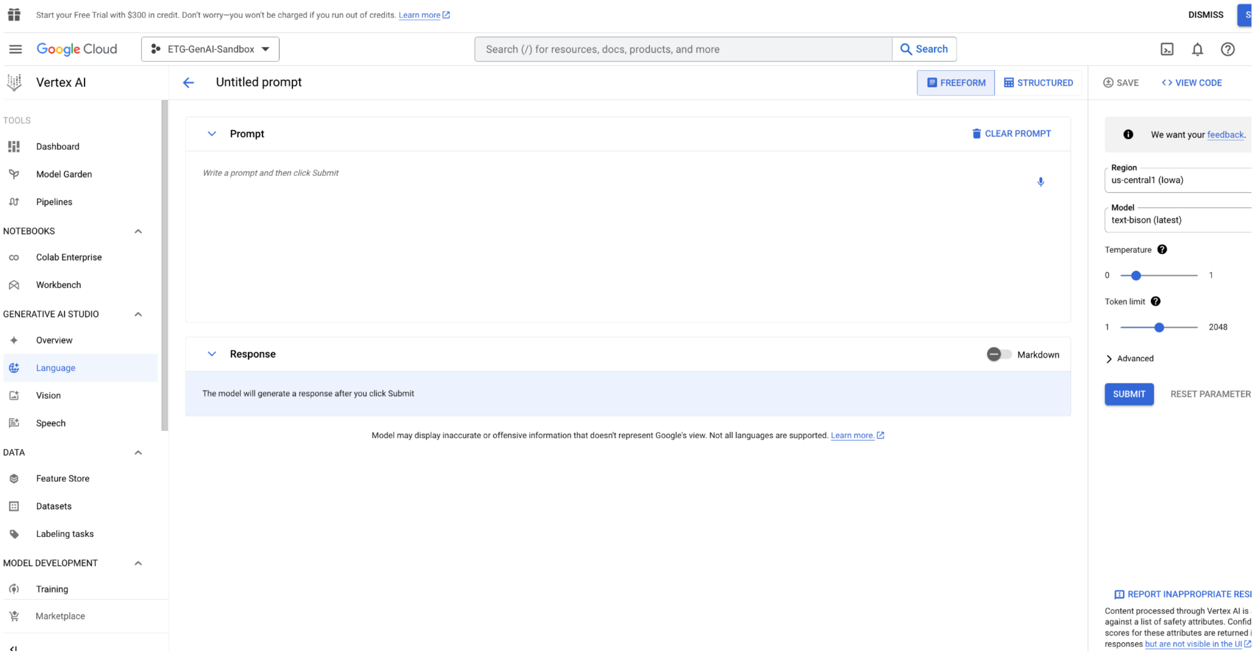
Progress of prompt modeling is as below:

1. During the past weeks, plain text has been created as a Promp:

[Travel Agency Prompt]

2. Training a Model via Google Cloud Vertex AI APIs:

- The first step is producing the prompt in Vertex APIS:



Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits. Learn more

Write a prompt and then click Submit

The model will generate a response after you click Submit

Model may display inaccurate or offensive information that doesn't represent Google's view. Not all languages are supported. Learn more.

We want your feedback.

Region: us-central1 (Iowa)

Model: text-bison (latest)

Temperature

0 1

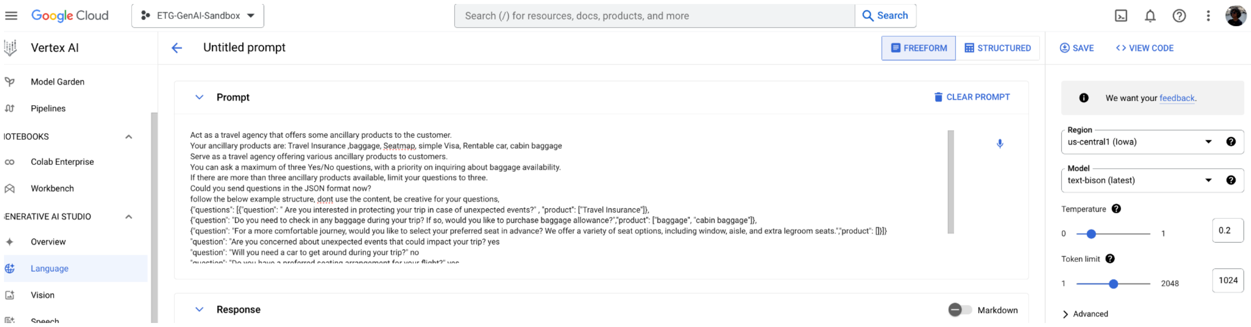
Token limit

1 2048

Content processed through Vertex AI is against a list of safety attributes. Confid scores for these attributes are returned i responses but are not visible in the UI

Figure 1: Google Cloud Vertex AI

- By definition of prompt and submission, we can expect a response from Vertex AI.

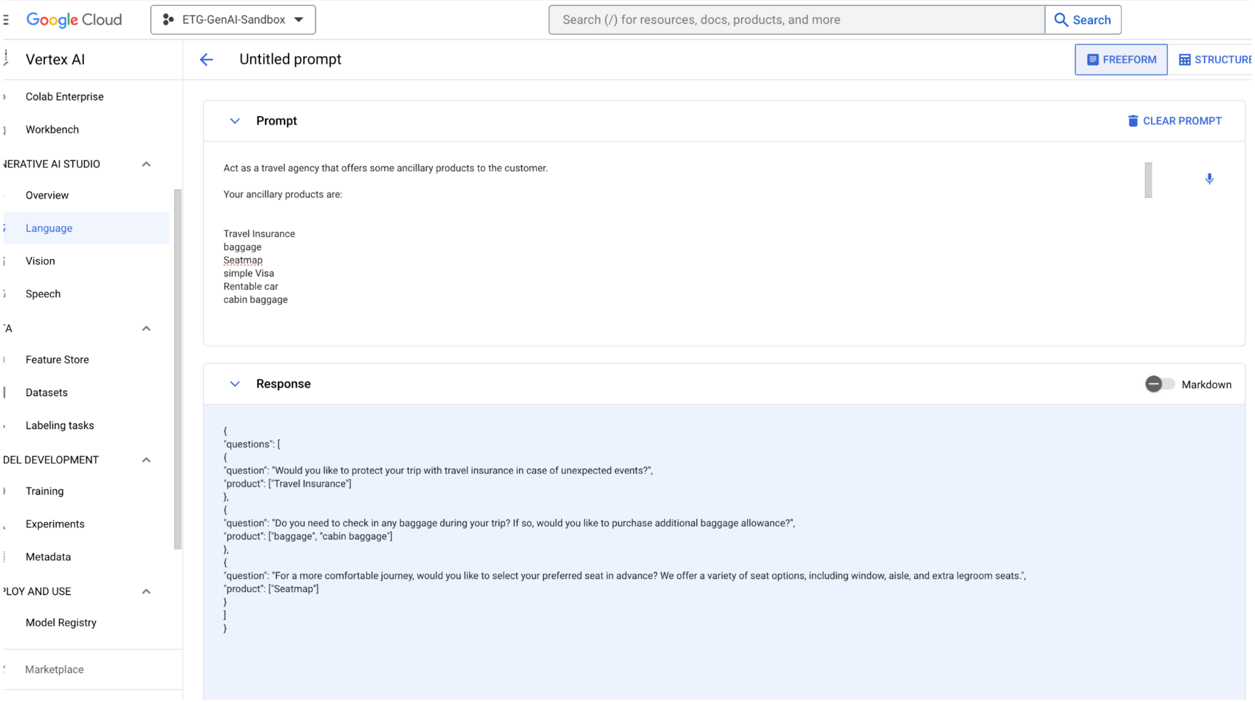


```
Act as a travel agency that offers some ancillary products to the customer.
Your ancillary products are: Travel Insurance ,baggage, Seatmap, simple Visa, Rentable car, cabin baggage
Serve as a travel agency offering various ancillary products to customers.
You can ask a maximum of three Yes/No questions, with a priority on inquiring about baggage availability.
If there are more than three ancillary products available, limit your questions to three.
Could you send questions in the JSON format now?
follow the below example structure, dont use the content, be creative for your questions,
{"questions": [{"question": " Are you interested in protecting your trip in case of unexpected events?" , "product": ["Travel Insurance"]},
{"question": "Do you need to check in any baggage during your trip? If so, would you like to purchase baggage allowance?","product": ["baggage", "cabin baggage"]},
{"question": "For a more comfortable journey, would you like to select your preferred seat in advance? We offer a variety of seat options, including window, aisle, and extra legroom seats.","product": []}]}
"question": "Are you concerned about unexpected events that could impact your trip? yes
"question": "Will you need a car to get around during your trip?" no
"question": "Do you have a preferred seating arrangement for your flight?" yes
```

Region
us-central1 (Iowa)
Model
text-bison (latest)
Temperature
0 1 0.2
Token limit
1 2048 1024

Figure 2: Define and submit a prompt for Google Cloud Vertex AI

- Getting a proper response from Vertex AI



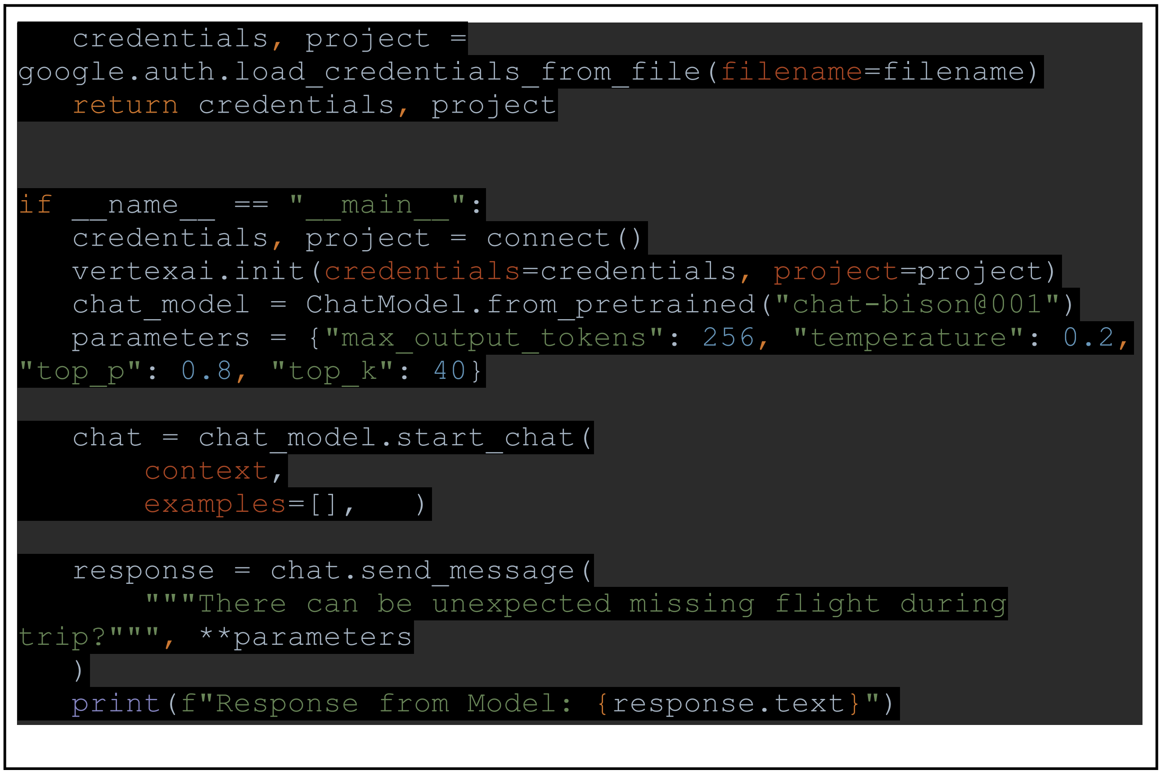
```
Act as a travel agency that offers some ancillary products to the customer.

Your ancillary products are:


Travel Insurance
baggage
Seatmap
simple Visa
Rentable car
cabin baggage
```

```
{
"questions": [
{
"question": "Would you like to protect your trip with travel insurance in case of unexpected events?",
"product": ["Travel Insurance"]
},
{
"question": "Do you need to check in any baggage during your trip? If so, would you like to purchase additional baggage allowance?",
"product": ["baggage", "cabin baggage"]
},
{
"question": "For a more comfortable journey, would you like to select your preferred seat in advance? We offer a variety of seat options, including window, aisle, and extra legroom seats.",
"product": ["Seatmap"]
}
]
}
```

Figure 3: Response from Google Cloud Vertex AI

Accuracy of response is based on prompt. As much as a prompt definition is more understandable and with more detail, the response will become more reliable.
The sensitivity of response can be adjusted by temperature.

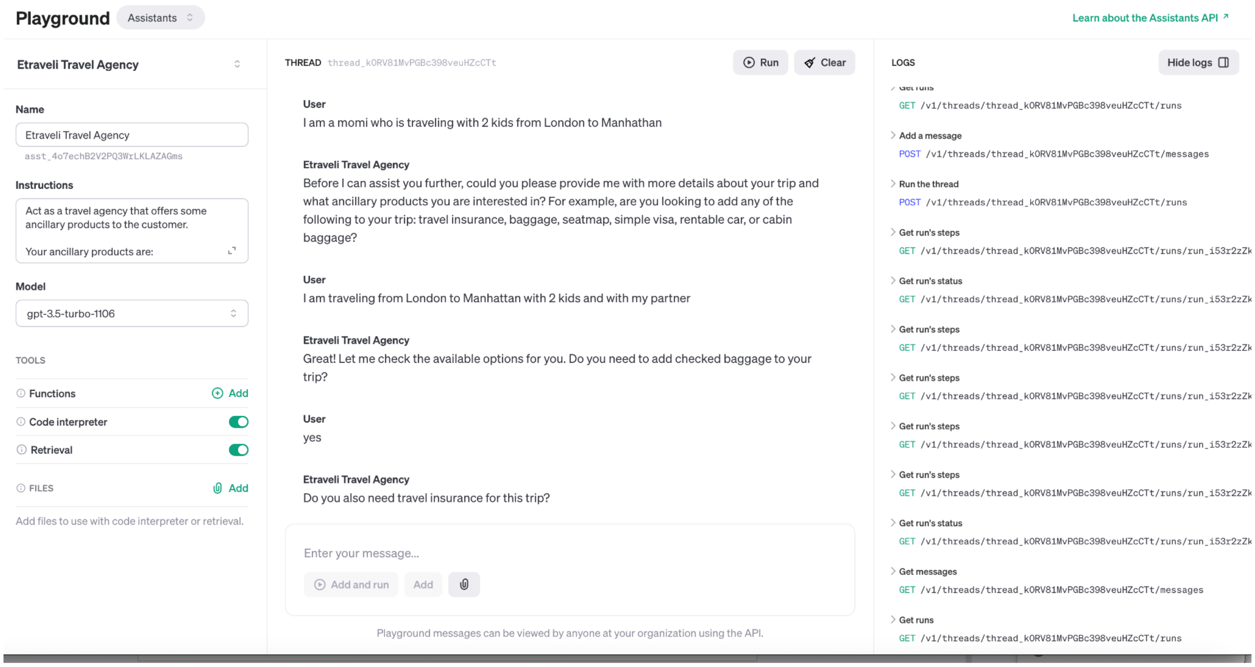
2. During this week whatever is done via Google Cloud Vertex AI is implemented to the Python code:
Code is reachable via git at below address:
customer-service-chatbot



```
   credentials, project =
google.auth.load_credentials_from_file(filename=filename)
   return credentials, project


if __name__ == "__main__":
   credentials, project = connect()
   vertexai.init(credentials=credentials, project=project)
   chat_model = ChatModel.from_pretrained("chat-bison@001")
   parameters = {"max_output_tokens": 256, "temperature": 0.2,
"top_p": 0.8, "top_k": 40}

   chat = chat_model.start_chat(
       context,
       examples=[],    )

   response = chat.send_message(
       """There can be unexpected missing flight during
trip?""", **parameters
   )
   print(f"Response from Model: {response.text}")
```

Figure 4: Implementation of Google Cloud Vertex AI via Python

3. This week's work is more about using the latest APIs in OpenAI to generate a fine-tuned model


Playground
Assistants
Learn about the Assistants API ↗
Run
Clear
Hide logs
TOOLS
Functions
Add
FILES
Add
Add files to use with code interpreter or retrieval.
Add and run
Add
Playground messages can be viewed by anyone at your organization using the API.

Etraveli Travel Agency

Name: Etraveli Travel Agency
asst_4o7echB2V2PQ3WrLKLAZAGms

Instructions: Act as a travel agency that offers some ancillary products to the customer.

Your ancillary products are:

Model: gpt-3.5-turbo-1106

Code interpreter
Retrieval

THREAD thread_kORV81MvPGBc398veuHZcCTt

User
I am a momi who is traveling with 2 kids from London to Manhathan

Etraveli Travel Agency
Before I can assist you further, could you please provide me with more details about your trip and what ancillary products you are interested in? For example, are you looking to add any of the following to your trip: travel insurance, baggage, seatmap, simple visa, rentable car, or cabin baggage?

User
I am traveling from London to Manhattan with 2 kids and with my partner

Etraveli Travel Agency
Great! Let me check the available options for you. Do you need to add checked baggage to your trip?

User
yes

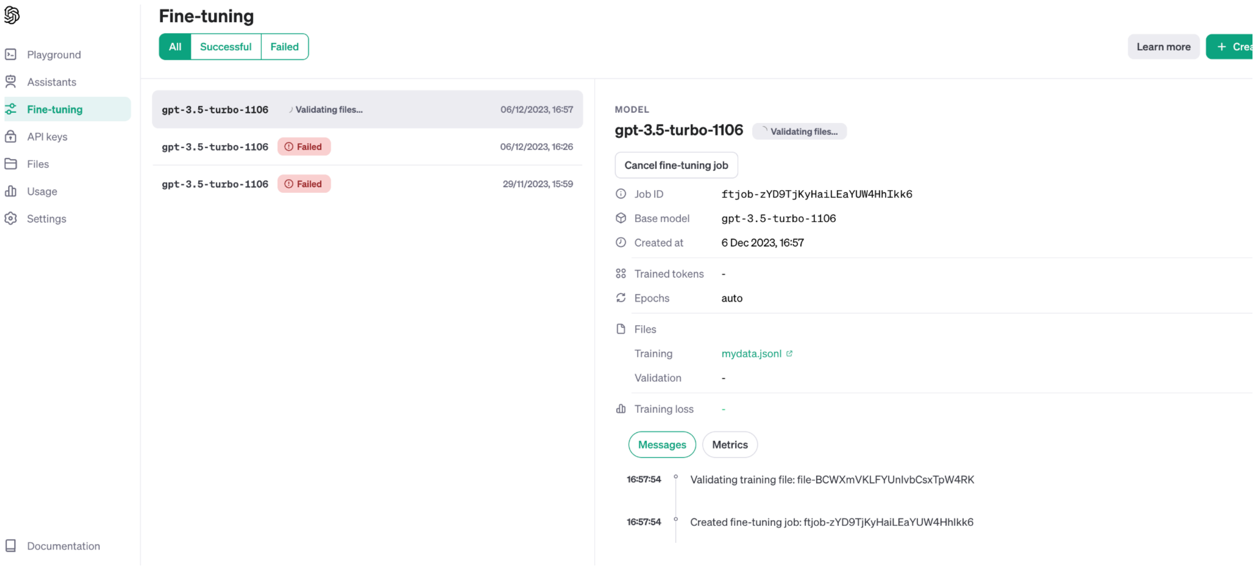
Etraveli Travel Agency
Do you also need travel insurance for this trip?

Enter your message...

LOGS

```
Get runs
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs
Add a message
POST /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/messages
Run the thread
POST /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs
Get run's steps
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's status
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's steps
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's steps
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's steps
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's steps
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get run's status
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs/run_i53r2zZk
Get messages
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/messages
Get runs
GET /v1/threads/thread_kORV81MvPGBc398veuHZcCTt/runs
```

Figure 5: AI Assistance via OpenAI APIs



Fine-tuning

gpt-3.5-turbo-1106	Validating files...	06/12/2023, 16:57
gpt-3.5-turbo-1106	Failed	06/12/2023, 16:26
gpt-3.5-turbo-1106	Failed	29/11/2023, 15:59

MODEL
gpt-3.5-turbo-1106 Validating files...

Job ID	ftjob-zYD9TjKyHaiLEaYUW4HhIkk6
Base model	gpt-3.5-turbo-1106
Created at	6 Dec 2023, 16:57
Trained tokens	-
Epochs	auto
Files	
Training	mydata.jsonl
Validation	-
Training loss	-

```
16:57:54  Validating training file: file-BCWXmVKLFYUnlvbCsxTpW4RK
16:57:54  Created fine-tuning job: ftjob-zYD9TjKyHaiLEaYUW4Hhlkk6
```

Figure 6: Fine-tune model via OpenAI APIs

4. Right now and until next week fine-tuning will be implemented in Python

   The code is reachable here:
   fine_tune.model