L4: Process Inputs: Chain of Thought Reasoning

Setup

Load the API key and relevant Python libaries.

In this course, we've provided some code that loads the OpenAI API key for you.

In [10]:
```
import os
import openai
import sys
from dotenv import load_dotenv, find_dotenv
_ = load_dotenv(find_dotenv())

openai.api_key  = os.environ['OPENAI_API_KEY']
```

In [11]:
```
def get_completion_from_messages(messages,
                                 model="gpt-3.5-turbo",
                                 temperature=0, max_tokens=500):
    response = openai.ChatCompletion.create(
        model=model,
        messages=messages,
        temperature=temperature,
        max_tokens=max_tokens,
    )
    return response.choices[0].message["content"]
```

Chain-of-Thought Prompting

In [12]:
```
delimiter = "####"
system_message = f"""
Follow these steps to answer the customer queries.
The customer query will be delimited with four hashtags,\
i.e. {delimiter}.

Step 1:{delimiter} First decide whether the user is \
asking a question about a specific product or products. \
Product cateogry doesn't count.

Step 2:{delimiter} If the user is asking about \
specific products, identify whether \
the products are in the following list.
All available products:
1. Product: TechPro Ultrabook
   Category: Computers and Laptops
   Brand: TechPro
   Model Number: TP-UB100
   Warranty: 1 year
   Rating: 4.5
   Features: 13.3-inch display, 8GB RAM, 256GB SSD, Intel Core i5 processor
   Description: A sleek and lightweight ultrabook for everyday use.
   Price: $799.99

2. Product: BlueWave Gaming Laptop
   Category: Computers and Laptops
   Brand: BlueWave
   Model Number: BW-GL200
   Warranty: 2 years
   Rating: 4.7
   Features: 15.6-inch display, 16GB RAM, 512GB SSD, NVIDIA GeForce RTX 306
   Description: A high-performance gaming laptop for an immersive experienc
   Price: $1199.99

3. Product: PowerLite Convertible
   Category: Computers and Laptops
   Brand: PowerLite
   Model Number: PL-CV300
   Warranty: 1 year
   Rating: 4.3
   Features: 14-inch touchscreen, 8GB RAM, 256GB SSD, 360-degree hinge
   Description: A versatile convertible laptop with a responsive touchscree
   Price: $699.99

4. Product: TechPro Desktop
   Category: Computers and Laptops
   Brand: TechPro
   Model Number: TP-DT500
   Warranty: 1 year
   Rating: 4.4
   Features: Intel Core i7 processor, 16GB RAM, 1TB HDD, NVIDIA GeForce GTX
   Description: A powerful desktop computer for work and play.
   Price: $999.99

5. Product: BlueWave Chromebook
   Category: Computers and Laptops
   Brand: BlueWave
```

```
   Model Number: BW-CB100
   Warranty: 1 year
   Rating: 4.1
   Features: 11.6-inch display, 4GB RAM, 32GB eMMC, Chrome OS
   Description: A compact and affordable Chromebook for everyday tasks.
   Price: $249.99

Step 3:{delimiter} If the message contains products \
in the list above, list any assumptions that the \
user is making in their \
message e.g. that Laptop X is bigger than \
Laptop Y, or that Laptop Z has a 2 year warranty.

Step 4:{delimiter}: If the user made any assumptions, \
figure out whether the assumption is true based on your \
product information.

Step 5:{delimiter}: First, politely correct the \
customer's incorrect assumptions if applicable. \
Only mention or reference products in the list of \
5 available products, as these are the only 5 \
products that the store sells. \
Answer the customer in a friendly tone.

Use the following format:
Step 1:{delimiter} <step 1 reasoning>
Step 2:{delimiter} <step 2 reasoning>
Step 3:{delimiter} <step 3 reasoning>
Step 4:{delimiter} <step 4 reasoning>
Response to user:{delimiter} <response to customer>

Make sure to include {delimiter} to separate every step.
"""
```

In [13]:
```
user_message = f"""
by how much is the BlueWave Chromebook more expensive \
than the TechPro Desktop"""

messages =  [
{'role':'system',
 'content': system_message},
{'role':'user',
 'content': f"{delimiter}{user_message}{delimiter}"},
]

response = get_completion_from_messages(messages)
print(response)
```

```
Step 1:#### The user is asking a question about two specific products, the Bl
ueWave Chromebook and the TechPro Desktop.
Step 2:#### The prices of the two products are as follows:
- BlueWave Chromebook: $249.99
- TechPro Desktop: $999.99
Step 3:#### The user is assuming that the BlueWave Chromebook is more expensi
ve than the TechPro Desktop.
Step 4:#### The assumption is incorrect. The TechPro Desktop is actually more
expensive than the BlueWave Chromebook.
Response to user:#### The BlueWave Chromebook is actually less expensive than
the TechPro Desktop. The BlueWave Chromebook costs $249.99 while the TechPro
Desktop costs $999.99.
```

In [14]:
```
user_message = f"""
do you sell tvs"""
messages =  [
{'role':'system',
 'content': system_message},
{'role':'user',
 'content': f"{delimiter}{user_message}{delimiter}"},
]
response = get_completion_from_messages(messages)
print(response)
```

```
Step 1:#### The user is asking if the store sells TVs.
Step 2:#### The list of available products does not include any TVs.
Response to user:#### I'm sorry, but we do not sell TVs at this store. Our av
ailable products include computers and laptops.
```

Inner Monologue

- Since we asked the LLM to separate its reasoning steps by a delimiter, we can hide the chain-of-thought reasoning from the final output that the user sees.

In [15]:
```
try:
    final_response = response.split(delimiter)[-1].strip()
except Exception as e:
    final_response = "Sorry, I'm having trouble right now, please try asking

print(final_response)
```

```
I'm sorry, but we do not sell TVs at this store. Our available products inclu
de computers and laptops.
```

In [ ]:
```

```