# A shopping assistant chatbot

A shopping assistant chatbot is an AI-powered conversational agent that can interact with customers to help them search, browse, and purchase products online. The main goals of such a chatbot include:

* Providing a natural and intuitive self-service experience for customers to find and buy products through conversational interactions.
* Answering common customer queries related to products, pricing, availability, shipping, returns, etc.
* Making personalized product recommendations based on customer preferences and purchase history.
* Assisting customers in completing purchases as quickly and seamlessly as possible.

Overall, a shopping bot makes the online shopping experience more intuitive and efficient for customers by providing a helpful conversational agent that can fulfill most shopping-related information needs.

Setup

import os

import openai

from dotenv import load\_dotenv, find\_dotenv\_ = load\_dotenv(find\_dotenv())

openai.api\_key = os.getenv('OPENAI\_API\_KEY')

def get\_completion(prompt, model="gpt-3.5-turbo"):

messages = [{"role": "user", "content": prompt}]

response = openai.ChatCompletion.create(

model=model,

messages=messages,

temperature=0, # this is the degree of randomness of the model's output

)

return response.choices[0].message["content"]

# print(str(response.choices[0].message))

return response.choices[0].message["content"]

messages = [

{'role':'system', 'content':'You are an assistant that speaks like Shakespeare.'},

{'role':'user', 'content':'tell me a joke'},

{'role':'assistant', 'content':'Why did the chicken cross the road'},

{'role':'user', 'content':'I don\'t know'} ]

response = get\_completion\_from\_messages(messages, temperature=1)

print(response)

Output:



messages = [

{'role':'system', 'content':'You are friendly chatbot.'},

{'role':'user', 'content':'Hi, my name is Nandini'} ]

response = get\_completion\_from\_messages(messages, temperature=1)

print(response)

Output:



messages = [

{'role':'system', 'content':'You are friendly chatbot.'},

{'role':'user', 'content':'Yes, can you remind me, What is my name?'} ]

response = get\_completion\_from\_messages(messages, temperature=1)

print(response)

0utput:



messages = [

{'role':'system', 'content':'You are friendly chatbot.'},

{'role':'user', 'content':'Hi, my name is Nandini'},

{'role':'assistant', 'content': "Hi Nandini! It's nice to meet you. \

Is there anything I can help you with today?"},

{'role':'user', 'content':'Yes, you can remind me, What is my name?'} ]

response = get\_completion\_from\_messages(messages, temperature=1)

print(response)

0utput:



def collect\_messages(\_):

prompt = inp.value\_input

inp.value = ''

context.append({'role':'user', 'content':f"{prompt}"})

response = get\_completion\_from\_messages(context)

context.append({'role':'assistant', 'content':f"{response}"})

panels.append(

pn.Row('User:', pn.pane.Markdown(prompt, width=600)))

panels.append(

pn.Row('Assistant:', pn.pane.Markdown(response, width=600, style={'background-color': '#F6F6F6'})))

return pn.Column(\*panels)

import panel as pn # GUI

pn.extension()

panels = [] # collect display

context = [{'role':'system', 'content':"""

You're ShoppingBot, your friendly assistant for all your shopping needs! \

I'm here to help you find everything you need. \

Just let me know what items you'd like to add to your cart, \

and I'll make sure you get the best deals. \

After you've finished shopping, I'll summarize your order \

and check if there's anything else you'd like. \

If you're ready to checkout, I'll guide you through the payment process. \

Let's get started! \

Here's what we have in store for you:

Clothing:

- T-shirts:

- Plain T-shirt: $12.95

- Graphic Tee: $10.00

- V-neck: $7.00

- Jeans:

- Regular Fit: $34.95

- Skinny Fit: $29.95

- Bootcut: $24.50

Accessories:

- Sunglasses: $19.95

- Watches:

- Digital: $29.99

- Analog: $39.99

Electronics:

- Smartphones:

- Android: $399.99

- iPhone: $799.99

- Laptops:

- Windows: $899.99

- MacBook: $1299.99

Toys:

- Action Figures: $14.99

- Dolls: $9.99

- Board Games:

- Monopoly: $24.99

- Scrabble: $19.99

Remember, if you need any specific sizes or colors, just let me know! \

And if you're unsure about anything, feel free to ask. Happy shopping!\

"""}]

inp = pn.widgets.TextInput(value="Hi", placeholder='Enter text here…')

button\_conversation = pn.widgets.Button(name="Chat!")

interactive\_conversation = pn.bind(collect\_messages, button\_conversation)

dashboard = pn.Column(

inp,

pn.Row(button\_conversation),

pn.panel(interactive\_conversation, loading\_indicator=True, height=300),

)

Dashboard

Output:

A screenshot of a computer

Description automatically generated

A white text on a white background

Description automatically generated

messages = context.copy()

messages.append({

'role': 'system',

'content': "Alright, let's create a summary of your shopping order in JSON format. Include the clothes names,price,size,description,and any additional information you find relevant."

})

response = get\_completion\_from\_messages(messages, temperature=0)

print(response)

Output:

