

Group project

For your group project, design and implement analytic bots for the company. Assume you are advising an online retailer or a company on customer service.

Requirements

The chatbot must respond to the queries of the users based on their inputs. The range of questions asked by users should be restricted based on the context. Includes:

- At least 4 conversation scenarios.
- Each scenario should consist of at least 3 questions and potential answer pairs.
- You should suggest at least one scenario which could integrate data analytics aspects for the company or retailers. At least one scenario should cover how apply one of the data analytics libraries or API.
- To increase flexibility in handling data, incorporate Pandas into the retailer's data management system (e.g., menu, price, etc.).
- In addition, suggest at least one data analytics related Python library or API (excluding Pandas) to further develop your chatbot functionality.

1. Proposal

Format: .pptx (less than 3 pages) Problem Statement

- Brief description of the company or retailer
- The rationale for developing a chatbot for the retailer

Deadline: 7th Aug

2. Code

Format: .ipynb Submit your Jupyter notebook file:

- Beautify your code – make clear documentation.
- To increase flexibility in handling data, incorporate Pandas into the retailer's data management system (e.g., menu, price, etc.). If you have already incorporated Pandas from assignment 1, you are exempted from this task.
- Logically describe the rule and implement the rule.
- In addition, suggest at least one data analytics related Python library or API (excluding Pandas) to further develop your chatbot functionality.

Deadline: 14th Aug

3. Final report

Format: .pptx (less than 20 pages)

- Brief description of the company or retailer
- The rationale for developing a chatbot for the retailer - **What's something new from the current service that they are offering?**
- Scenarios - Describe your chatbot and scenarios.
- Describe each scenario that you considered briefly.
- *If the chatbot replicates the current services that the retailer offers from other online sources (e.g., web order, uber eats, etc.), mark this as the **replicated services**.*
- *If you have suggested any additional services that retailers are not offering, mark this scenario as the **new scenario**. Include the rationale for considering this scenario and how this scenario can help users' experience with the retailers. [Note] If you have made the*

changes from your first group assignment, describe how you changed the scenarios or what scenario you have considered in the Appendix.

2. Describe python libraries (for data analytics) or API (1 page)

- Suggest how you would like to integrate analytics Python libraries or API.
- Include the rationale for why the libraries are useful for the retailer's service.
- Future suggestions for the code extension (e.g., additional python libraries). You don't need to write the code. You can share your idea and the proper python library.
- Be creative!

[Extra Point] – Use the library to extend your code. [Note] If you have made the changes from your first group assignment other than applying data analytics related python libraries or API (excluding pandas), describe how you changed the code in the Appendix. [Extra Point] Explore more than 1 libraries or APIs. [Extra Point] Share Database Schema to visualize your data collection plan 3. Suggestion for the retailers, and future suggestion

- Describe how the chatbot will help the retailer to improve your business.

Deadline: 21st Aug / 23rd Aug

4. Presentation

- Prepare a presentation as the version for pitching your ideas to the clients (company).
- Assume that you had a proposal for your initial idea and 2nd round meetings with the company. Scenarios that you have considered.
- Describe the Python libraries or API that you used for the group assignment 2.
- Explain rationale for incorporating specific python libraries & API.
- If you modify the code (e.g., consider Pandas for data analytics library). Describe the major changes that you made from the previous code. [Note] You don't need to share the demo of your chatbot

Proposal

Book retailer: Argo Bookshop

1. Description

Argo Bookshop is the oldest English-language independent bookstore in the city of Montreal, located in the Shaughnessy Village since 1966. They offer a well-curated selection of fiction and non-fiction, including literature, poetry, science fiction and fantasy, science, philosophy, history, and biography. The bookstore is known for its curated selection of books, and its commitment to promoting local authors and diverse voices.

2. Rationale for developing a chatbot

2.1. 24/7 Customer Service:

A chatbot can provide round-the-clock customer service, answering queries and providing information even outside of business hours. This ensures that customers can receive immediate responses to their inquiries at any time, enhancing their overall experience.

2.2. Enhancing Online Presence:

While Argo Bookshop is primarily an offline retailer, a chatbot can help bridge the gap between the physical and digital world. It can provide a platform for customers to engage with the store online, increasing its visibility and reach.

2.3. Efficient Customer Support:

Chatbots can handle multiple inquiries simultaneously, reducing wait times and increasing efficiency. They can provide instant responses to common questions about store hours, book availability, upcoming events, and more, freeing up staff time to focus on more complex customer needs.

2.4. Personalized Recommendations:

With machine learning capabilities, the chatbot can analyze customer preferences and behavior to provide personalized book recommendations. This not only enhances the customer experience but also drives sales.

2.5. Data Collection and Analysis:

Chatbots can collect and analyze data from customer interactions, providing valuable insights into customer behavior, preferences, and trends. This data can be used to make informed decisions about inventory, marketing strategies, and other business aspects.

In conclusion, implementing a chatbot for Argo Bookshop is a strategic move towards enhancing customer service, improving operational efficiency, and driving business growth. It aligns with the current digital transformation trends and positions Argo Bookshop as a forward-thinking retailer in the competitive book market.

Scenarios

1. Scenario 1: Book Availability Inquiry

Question 1:

User: "Do you have '1984' by George Orwell in stock?"

Chatbot: "Let me check that for you."

The chatbot would then use the Pandas API to search the inventory database for the book.

Chatbot: "Yes, '1984' by George Orwell is currently in stock."

Question 2:

User: "How many copies are available?"

Chatbot: "One moment, I'll check the quantity for you."

The chatbot would use the Pandas API to retrieve the quantity of the book in the inventory.

Chatbot: "We currently have 5 copies of '1984' by George Orwell in stock."

Question 3:

User: "What is the price of the book?"

Chatbot: "Let me find that information for you."

The chatbot would use the Pandas API to retrieve the price of the book from the inventory.

Chatbot: "'1984' by George Orwell is priced at \$15."

Question 4:

User: "Can you reserve a copy for me?"

Chatbot: "Sure, could you please provide your name and contact information?"

Upon receiving the user's information, the chatbot would use the Pandas API to update the inventory and reserve the book for the user.

Chatbot: "Thank you. I have reserved a copy of '1984' by George Orwell for you. You can pick it up at our store within the next 48 hours."

2. Scenario 2: Book Recommendation

Question 1:

User: "Can you recommend a good mystery novel?"

Chatbot: "Sure, do you have a preference for any particular author or series?"

Question 2:

User: "I enjoyed books by Agatha Christie."

Chatbot: "Great! Based on your preference for Agatha Christie, you might enjoy 'The Murder of Roger Ackroyd'. It's one of her most popular works."

Question 3:

User: "Sounds interesting. Do you have it in stock?"

Chatbot: "Let me check that for you."

The chatbot would then use the Pandas API to search the inventory database for the book.

Chatbot: "Yes, 'The Murder of Roger Ackroyd' is currently in stock."

Question 4:

User: "What is the price of the book?"

Chatbot: "Let me find that information for you."

The chatbot would use the Pandas API to retrieve the price of the book from the inventory.

Chatbot: "'The Murder of Roger Ackroyd' is priced at \$12."

3. Scenario 3: Store Information Inquiry

Question 1:

User: "What are your store hours?"

Chatbot: "Our store is open from 10 AM to 6 PM from Monday to Saturday. On Sundays, we are open from 12 PM to 5 PM."

Question 2:

User: "Where are you located?"

Chatbot: "We are located at 1915 Saint Catherine St W, Montreal, QC H3H 1M3, Canada."

Question 3:

User: "Do you have any upcoming events?"

Chatbot: "Let me check our events calendar."

The chatbot would use the Pandas API to retrieve the upcoming events from the events database.

Chatbot: "Yes, we have a book signing event with a local author this coming Saturday at 2 PM."

Question 4:

User: "Do you have a parking facility?"

Chatbot: "I'm sorry, we do not have a dedicated parking facility. However, there is street parking available nearby."

4. Scenario 4: Order Status Inquiry

Question 1:

User: "I'd like to check the status of my order. The order number is 12345."

Chatbot: "Let me find that information for you."

The chatbot would use the Pandas API to retrieve the order status from the order database.

Chatbot: "Your order is currently being processed and will be ready for pickup in 2-3 days."

Question 2:

User: "Can I change the pickup date?"

Chatbot: "Sure, when would you like to pick up your order?"

Question 3:

User: "I'd like to pick it up next Monday."

Chatbot: "Alright, I have updated your pickup date to next Monday."

The chatbot would use the Pandas API to update the pickup date in the order database.

Chatbot: "Your pickup date has been successfully updated. You can pick up your order next Monday."

Question 4:

User: "Can I add another book to my order?"

Chatbot: "Sure, which book would you like to add?"

Upon receiving the user's response, the chatbot would use the Pandas API to update the order in the database.

Chatbot: "I have added the book to your order. It will be ready for pickup along with your other items next Monday."

5. Scenario 5: Handling Complaints

Question 1:

User: "I have a complaint about a book I purchased."

Chatbot: "I'm sorry to hear that. Could you please provide more details about the issue?"

Question 2:

User: "The book I received has some torn pages."

Chatbot: "I apologize for the inconvenience. Could you please provide your order number so I can assist you further?"

Question 3:

User: "My order number is 67890."

Chatbot: "Thank you for the information."

The chatbot would use the Pandas API to retrieve the order details from the order database.

Chatbot: "I found your order. We can replace the book for you. Would you like to come to the store for the replacement or should we ship it to you?"

Question 4:

User: "Please ship it to me."

Chatbot: "Alright, we will ship a replacement book to the address on file. You should receive it within 5-7 business days."

The chatbot would use the Pandas API to update the order status in the database.

Chatbot: "Your replacement order has been processed. You will receive a confirmation email with the tracking details shortly."

Additional content

1. Packages

1. gunthercox/ChatterBot: ChatterBot is a machine learning, conversational dialog engine for creating chat bots ([github.com](https://github.com/gunthercox/ChatterBot))
2. Natural Language Processing With Python's NLTK Package – Real Python

2. Replicated Services

1. **Book Availability Inquiry:** This service is typically offered by most online retailers, including web order services. The chatbot replicates this service by providing real-time information about book availability and quantity.
2. **Order Status Inquiry:** This is another common service offered by online retailers. The chatbot replicates this service by providing real-time updates on order status and allowing users to modify their orders.

3. New Scenarios

1. **Book Recommendation:** While some online retailers offer book recommendations, they are often based on purchase history or popular trends. The chatbot enhances this service by providing personalized recommendations based on user preferences and interactions. This can enhance the user experience by helping them discover new books that align with their interests.

Rationale: Personalized recommendations can increase customer satisfaction and drive sales. They can also differentiate Argo Bookshop from other retailers by providing a more personalized and interactive shopping experience.

2. **Handling Complaints:** While most retailers have a process for handling complaints, the chatbot offers a more immediate and interactive way to resolve issues. It can instantly process complaints, provide solutions, and update the order database, providing a seamless experience for the user.

Rationale: Quick and effective complaint resolution can significantly enhance customer satisfaction and loyalty. It also reduces the workload for customer service staff, allowing them to focus on more complex issues.

3. **Store Information Inquiry:** While store information is typically available on the retailer's website, the chatbot provides this information in a more interactive and accessible way. It can provide real-time updates on store hours, location, and upcoming events based on the store's database.

Rationale: Providing store information through the chatbot can enhance the user experience by providing immediate responses to their queries. It can also drive foot traffic to the store by promoting upcoming events.

Database schema

Books

- BookID (Primary Key)
- Title
- Author
- Genre
- Price
- QuantityInStock

Orders

- OrderID (Primary Key)
- CustomerID (Foreign Key)
- BookID (Foreign Key)
- OrderDate
- PickupDate
- Status

Events

- EventID (Primary Key)
- EventName
- EventDate
- EventDescription

Customers

- CustomerID (Primary Key)
 - FirstName
 - LastName
 - Email
 - Phone
-

Sample code

```
class BookstoreBot:
    def __init__(self, db_path):
        self.chatbot = ChatBot('BookstoreBot')
        trainer = ChatterBotCorpusTrainer(self.chatbot)
        trainer.train('chatterbot.corpus.english')
        self.conn = sqlite3.connect(db_path)
        self.inventory = pd.read_sql_query("SELECT * FROM Books", self.conn)
        self.orders = pd.read_sql_query("SELECT * FROM Orders", self.conn)
        self.customers = pd.read_sql_query("SELECT * FROM Customers", self.conn)
        self.events = pd.read_sql_query("SELECT * FROM Events", self.conn)

        # Add methods for each scenario here
        # ...

    def respond(self, user_input):
        # Decide which scenario to use based on the user's input
        # This is a simplified example, in a real application you would want to use NLP techniques to
        understand the user's intent
        if 'availability' in user_input:
            return self.check_book_availability(user_input)
        elif 'recommend' in user_input:
            return self.recommend_book(user_input)
        elif 'order status' in user_input:
            return self.check_order_status(user_input)
        elif 'complaint' in user_input:
            return self.handle_complaint(user_input)
        elif 'store information' in user_input:
            return self.provide_store_information(user_input)
        else:
            return str(self.chatbot.get_response(user_input))
```
