REPORT ON RESTRAUNT CAFÉ CHATBOT USING NLP

TABLE OF CONTENTS

1. INTRODUCTION

This report describes the creation of a café chatbot driven by Natural Language Processing (NLP). We'll use Kaggle's "Café Chatbot Dataset" to get real-world chats, menu item. mappings, and comprehensive food information. This information will help the chatbot. comprehend consumer queries regarding the café's menu, services, feedback and operations.

Our objective is to develop a virtual assistant that improves the customer experience through simple and tailored chat conversations. The chatbot will help with reservations, ordering, menu access, customer support, and promotion notifications. The chatbot will grasp not just the surface meaning of user communications, but also underlying purpose of NLP and deep learning algorithms. The guarantees that replies are contextually appropriate and tailored to meet unique needs of individuals. This study describes our strategy to developing a user-friendly and intuitive chatbot that will change the café's customer experience.

Key Features of the Café Chatbot

Understanding Context : The chatbot will utilize NLP models to interpret and grasp the context of customer conversations comprehensively moving beyond individual keywords. This capability allows it to navigate through turn dialogues where the customers intentions may evolve or become clearer throughout the interaction.

Personalized Assistance : By examining interactions, preferences and the present context the chatbot can provide tailored suggestions, such, as recommending a menu item or proposing a new dish similar to what the customer typically enjoys.

Seamless Integration with Operational Systems : In addition, to handling customer interactions the chatbot can seamlessly integrate with the cafés reservation and ordering systems. This integration enables it to manage reservations efficiently process orders smoothly. Keep customers updated on their order status or waiting times.

2.ABSTRACT

The introduction of conversational agents,sometimes known as chatbot ,has transformed several businesses,including the restaurant and café industry.In this era of digital transformation,incorporating chatbots into restaurants and cafés has become more important for improving customer service,streamlining operations, and optimizing user experience.

This project suggests creating a chatbot built exclusively for restaurants and cafés,using the café chatbot dataset accessible on Kaggle.The dataset includes a wide range of interactions between users and a fictional café chatbot,offering a valuable source of conversational data for training and assessment.

The chatbot promises to provide consumers with a smooth and customized experience,allowing them to make reservations, inquire about menu items,place orders and request suggestions,among other features.By harnessing the chatbot which uses natural language

processing(NLP) methods,machine learning algorithms, and deep learning architecture to understand user queries ,identify intentions and provide suitable replies in real time.

The chatbot's architecture includes key components such as intent recognition,entity extraction,dialogue management,and answer creation.

The chatbot will be rigorously tested using quantitative and qualitative measures such as accuracy,response speed,user happiness and conversational fluency.Additionally,customer input and iterative changes will be critical in improving the chatbot's performance and usability in real-world restaurants and café settings.

Finally,the suggested chatbot has the potential to transform how customers engage with restaurants and cafes by providing a simple,fast and engaging route for obtaining services and information.Using the Café Chatbot Dataset and cutting edge NLP approaches,this project intends to promote conversational AI in the hospitality industry,promoting better customer experiences and operational efficiency.

3.OBJECTIVES

The Main Objective of Restaurant Café Using Natural Language Processing (NLP).The fundamental goal of using Natural Language Processing (NLP) in a restaurant café scenario is to improve and streamline many parts of the client experience, as well as to maximize operational efficiency.

The following are the major objectives:

1. **Improved Customer Interaction:** Implementing NLP enables the creation of conversational interfaces like as chatbots, which let consumers to communicate with the restaurant café using natural language inquiries. This allows for easier communication, more accessibility, and a more tailored experience for customers.

2. **Efficient Order Processing:** NLP may be used to interpret and process consumer orders submitted by text or speech. NLP systems may simplify operations by extracting key information like menu items, quantity, and special orders the ordering process, avoiding mistakes and lowering client wait times.

3.Menu Understanding and Recommendation: NLP systems may provide tailored suggestions by analyzing menu descriptions, client preferences, and previous order data. Understanding client preferences and dietary limitations allows the restaurant café to provide targeted suggestions, resulting in higher customer happiness and income.

4.Feedback Analysis: NLP approaches may be used to assess consumer comments and evaluations from a variety of sources, including social media, review sites, and questionnaires. Sentiment and topic modeling algorithms can extract important insights from unstructured text data, allowing the restaurant café to discover areas for development and answer customer problems in a proactive manner.

5.Operational efficiency: A side from customer-facing applications, natural language processing (NLP) may improve internal restaurant café operations. For example, it may automate routine processes like scheduling, inventory management, and employee communication, freeing up staff resources and increasing overall efficiency

6. Language support and accessibility: NLP technology can enable multilingual support, allowing the restaurant café to serve a broad clientele. By providing services in many languages, the institution may increase accessibility and provide a welcome environment for consumers with diverse linguistic origins.

Overall, the major goal of incorporating NLP into a restaurant café is to improve the client experience, streamline operations, and remain gain an advantage in an increasingly competitive sector by leveraging the power of language comprehension and automation.

4.DATASET INFORMATION:

The "Cafe Chatbot Dataset" is a useful resource for training and designing a chatbot designed particularly for café environments. It comprises of three core files, each of which provides crucial data and context for developing an efficient conversational agent.

1. **Conversation file:** This file comprises a series of conversations between users and the chatbot. These talks mimic real-world encounters that users may have while seeking assistance or making queries in a café setting. By studying these interactions, developers may teach the chatbot to comprehend different user intents, manage different sorts of requests, and create suitable replies.
2. **Item to ID File:** This file maps café products to their appropriate IDs. Each café item, such as drinks, snacks, or meals, is connected with a certain identification, which allows for more effective menu item processing and retrieval during chatbot conversations. This mapping allows

the chatbot to effectively read user queries linked to certain menu items and respond appropriately, such as placing orders or delivering information.

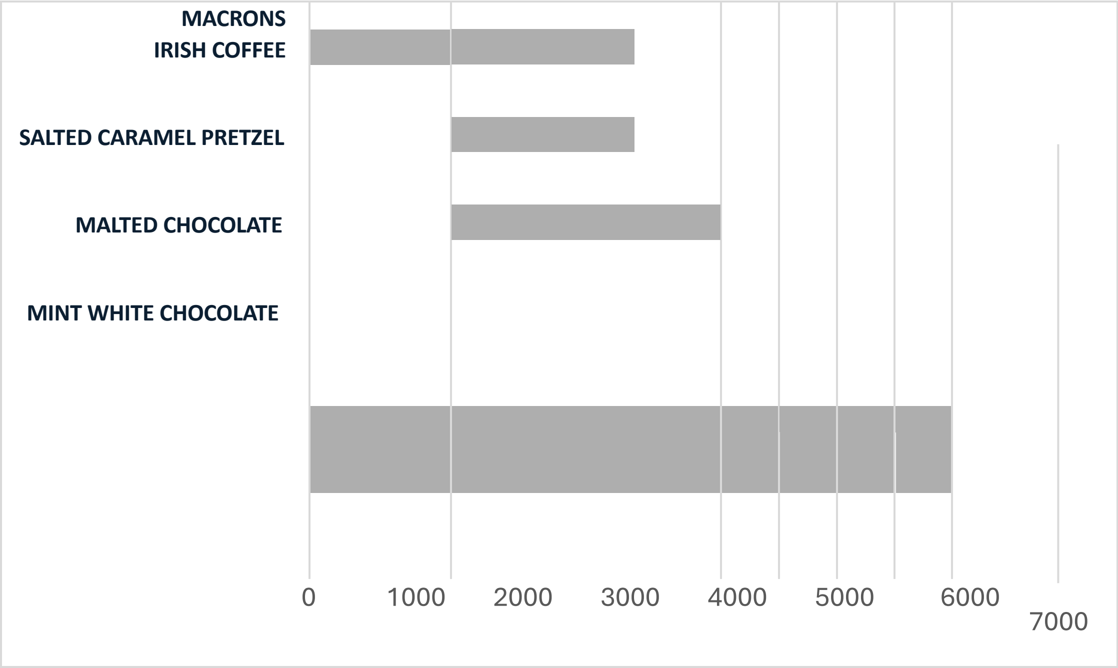
3.Food Files: The food file contains specific information on the café menu items. It contains features such as item names, descriptions, pricing, ingredients, and nutritional data. This extensive dataset enables the chatbot to provide detailed information about menu items, such as availability, ingredients, nutritional advice, and suggested combinations. Using this data, the chatbot can help consumers make smart food and beverage selections.

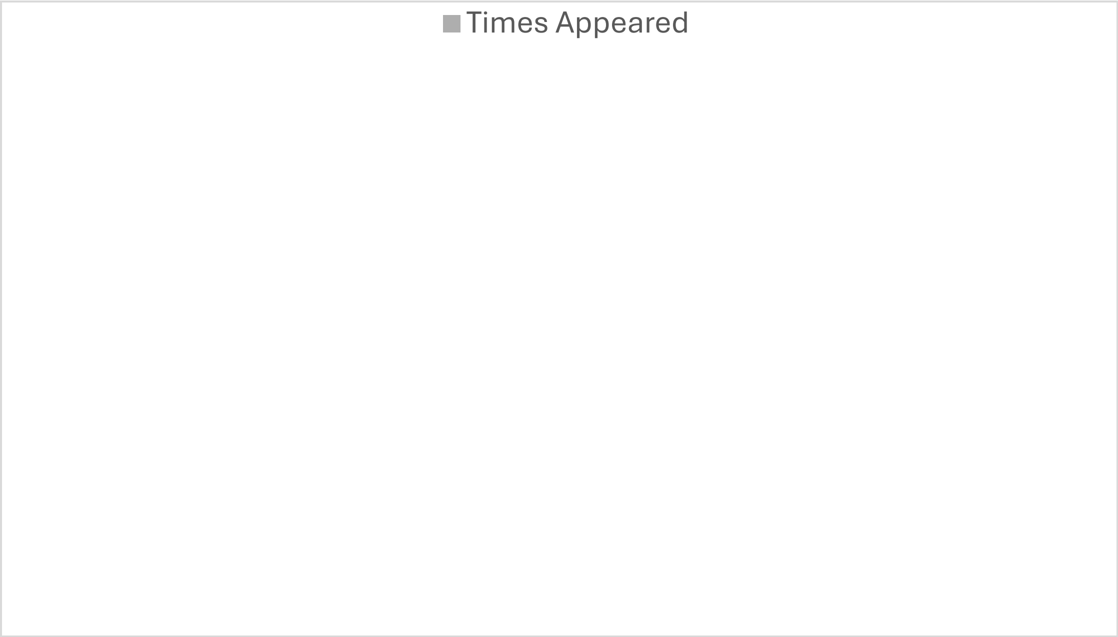
By integrating these files, developers may construct a café-oriented chatbot capable of comprehending customer questions and offering appropriate information regarding menu choices, help with orders and bookings, and provide a consistent client experience. The information enables the chatbot to learn from several encounters and tailor its replies to context, user preferences, and café-specific needs. Overall, the "Cafe Chatbot Dataset" allows developers to create intelligent conversational bots that improve customer service and engagement at cafes. **Here is the dataset link:**

https://www.kaggle.com/datasets/sonalibhoir/cafe-chatbot-dataset

5.EXPLORING DATASET ANALYSIS

Engaging in Exploratory Data Analysis (EDA) plays a role, in the data analysis journey as it helps us to understand the data, at hand spot trends and irregularities and develop theories and valuable insights.





The bar graph displays the "Top 5 Items, by Times Appeared" in a dataset indicating the frequency of menu items appearing in orders or mentions at a cafe or restaurant. Based on this chart here's an exploratory analysis using EDA :

1. **Mint White Chocolate:** This item appears frequently with over 4000 occurrences suggesting it could be an signature choice at the cafe.
2. **Malted Chocolate**: Coming in place it shows appearances compared to Mint White Chocolate but still holds its popularity.
3. **Salted Caramel Pretzel:** Positioned in the middle of the list indicating a level of popularity.
4. **Irish Coffee:** Frequent than the items hinting at being a niche or seasonal option.
5. **Macarons**: Among the 5 items this is seen often possibly indicating lower preference or being a newer addition to the menu.

For an EDA one would typically delve into :

Determining the frequency of each item to gauge their popularity accurately.

Comparing these items with others on the menu to assess their popularity, within the range offered.

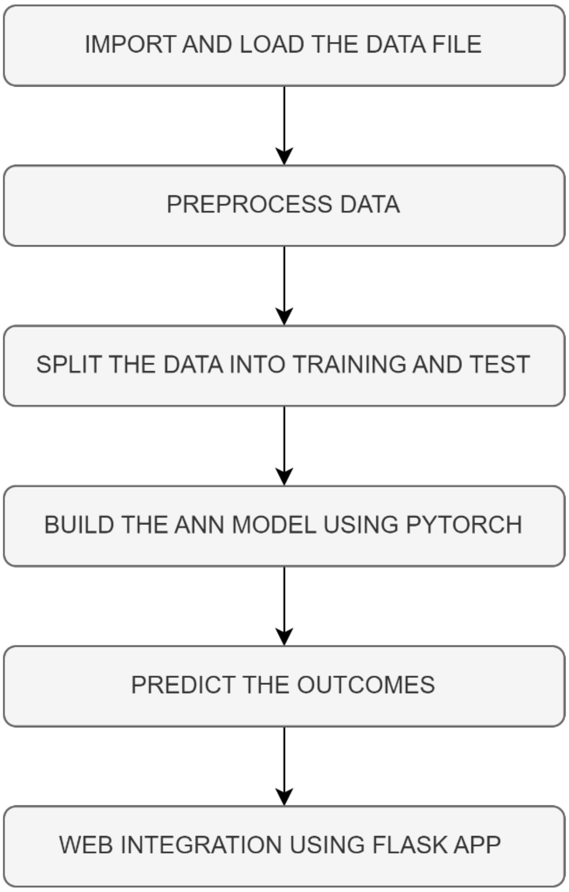
Examine the patterns over time to determine whether the demand, for these products remains steady grows or declines.

Compare these occurrences with influences like patterns, special offers or menu adjustments. This examination aids, in grasping customer choices organizing stock levels and developing marketing plans.

For example the coffee shop could increase its supply of Mint White Chocolate ingredients.

Focus on promoting Macarons vigorously to boost their appeal.

6.WORKFLOW


IMPORT AND LOAD THE DATA FILE
PREPROCESS DATA
SPLIT THE DATA INTO TRAINING AND TEST
BUILD THE ANN MODEL USING PYTORCH
PREDICT THE OUTCOMES
WEB INTEGRATION USING FLASK APP

a. IMPORT AND LOAD THE DATA FILE

During this stage we begin by bringing in the required libraries and importing the dataset from a file. This task is commonly carried out using tools such, as Pandas in Python which're capable of reading file types such, as CSV, Excel and others.

To work with the Cafe Chatbot dataset we need to import the information, from the CSV file into a Pandas DataFrame for handling.

b. PREPROCESS DATA

Data preprocessing includes tasks such, as tidying up the data (managing missing information eliminating duplicates) modifying it (normalizing, standardizing) converting variables selecting features and potentially enhancing features. The objective is to ready the dataset, for modeling purposes.

The Cafe Chatbot dataset probably contains language information so before processing may need to clean the text (such, as removing punctuation converting to lowercase and using stemming/lemmatization) encode it (convert text into data) address missing values and possibly extract features from the text .

clean and format the data for NLP tasks; create features like Bag of Words, N-grams, or word embeddings.

c. TESTING THE DATA

In a chatbot model trained on intents, each intent or "tag" is associated with a specific type of user input and has a corresponding response.

When a user types a message, like "Hi", the model analyzes it and predicts the intent behind the message.

It then selects the appropriate response linked to the predicted intent to reply to the user. This allows for a dynamic conversation where the chatbot can handle a wide range of inputs by mapping them to known intents.

d. BUILD THE ANN MODEL USING PYTORCH

Facilitates the creation of deep learning models for tasks like intent recognition and maintaining conversation context, enhancing the chatbot's responsiveness and accuracy. **e. PREDICT THE OUTCOMES**

When we have a trained network model we can make forecasts on fresh data. For instance in the case of a chatbot this might involve determining the purpose behind user messages or anticipating the word in a conversation. Here's the general process, for making predictions;

- Prepare Input Data : Any new data needs to go through preprocessing like the training data.
- Load the Model : Utilize Torch to load up the trained model.
- Generate Predictions : Use the model.predict() function to obtain the models output, which could either be a reply or an assessment of the input that can be linked to a response.

f. USING FLASK TO INTEGRATE THE CAFÉ CHATBOT

Deploying a model involves making it accessible, as a service through a web API allowing other software applications to utilize it. Flask, a web framework is commonly employed for this purpose. Here's an overview of the deployment process:

Getting Started with Flask and Understanding Flask:

Flask is often referred to as a "framework because of its nature and focus on providing only the essential tools and libraries necessary to kickstart a web application. This simplicity of Flask allows developers the flexibility to incorporate extensions tailored to their needs making it an excellent choice, for creating a cafe chatbot.

Enhanced Functionality:

One of Flasks strengths lies in its support for extensions that seamlessly integrate features into the application as if they were part of Flask itself. For instance developers can leverage extensions like Flask SQL Alchemy for database operations Flask WTF for form handling and Flask Login for user authentication enabling scalability of the chatbot.

Setting Up Your Flask Environment:

Installation and Configuration; To begin working with Flask start by installing it using pip and establish an environment to manage dependencies independently from Python projects. This approach ensures a clean and organized project structure.

Basic Project Layout: A standard structure for a Flask application typically consists of; app.py; This file serves as the core application, where the initialization of the Flask app occurs along, with defining routes.

Templates folder : Here you store HTML files utilized for rendering within the user interface. **Static folder :** Contains CSS stylesheets JavaScript scripts and image resources employed by the application.

In Flask routes are established using decorators that link URL patterns to corresponding Python functions.

These functions manage the rules of a business. React to user engagements. Incorporating Chatbot Logic, Dealing with Requests, Flask routes have the capability to receive user inputs, via GET and POST requests. In the case of a chatbot POST routes are utilized to accept user messages and provide replies.

Chatbot Responses: The logic for producing responses can be as straightforward as if conditions or more AI powered algorithms. Depending on the complexity level you may integrate natural language processing tools such as NLTK. Spacy to understand user inquiries and craft responses that mimic human speech patterns.

UI/UX Design: Constructing the Frontend; Utilize HTML for layout CSS for design and JavaScript for components. Flask can deliver HTML files containing forms for users to input their messages and buttons to submit them.

By using these instructions can create an engaging chatbot, with Flask designed to improve user interactions and handle tasks efficiently.

7.NATURAL LANGUAGE PROCESSING

Natural Language Processing(NLP) plays crucial role in the functionality of chatbots,such as those that would be built using the Café Chatbot dataset

Understanding NLP In Chatbots :

Chatbots Recognizing User Intent : A crucial aspect of chatbot operation is identifying what the user wants, which guides how the chatbot should reply. NLP algorithms are adept at categorizing user inputs into groups like "order a drink " "inquire about store hours ". Share feedback."

Managing Context: To offer responses a chatbot must grasp the conversations context. This can involve maintaining a state or utilizing techniques like context aware models.

Understanding Language: Fundamentally this involves breaking down sentences into words (tokenization) converting words into vectors (word embeddings) and comprehending sentence structures (parsing).

Handling Dialogues: NLP systems must determine the response based on the conversations status. This decision-making process may entail generating a response from scratch (using models) or selecting a reply, from predefined options (retrieval based models). Understanding the emotions conveyed by user inputs is crucial for chatbots to respond effectively in customer service situations.

In the field of **Natural language processing (NLP)** several techniques and tools are commonly employed:

Tokenization: Involves breaking down text into units, like words, characters or subwords serving as a step in many NLP tasks.

Frameworks: Python libraries like NLTK, spaCy, and PyTorch offer frameworks and pretrained models that streamline the development of NLP tools.

APPLYING NLP TO THE CAFÉ CHATBOT DATASET(USED TOKENISATION, STEMMING AND BAG OF WORDS)

When using a chatbot NLTK (Natural Language Toolkit) can be utilized to improve text processing functions by segmenting text, into tokens reducing words to their base form and employing a bag of words approach.

Here's the significance of each step :

TOKENIZATION : Tokenization refers to the act of dividing text into units known as tokens, which can range from words, to entire sentences.

In the realm of chatbots tokenization plays a role in segmenting user input into chunks for thorough examination and handling.

This process is essential, for grasping the meaning behind every word or expression entered by a user.

STEMMING : It is about simplifying words to their form. For instance "running" "runner" and "ran" could all be simplified to the root word "run“.
In the context of chatbots stemming aids in standardizing words, to their roots so that variations of a word are handled in a manner. This enhances the chatbots capacity to comprehend and interact with inputs effectively.

BAG OF WORDS : It breaks down text into words without taking into account grammar or sequence.
Use, in Chatbots , The bag of words model is used to transform text data into numerical feature vectors based on the frequency of words This allows the chatbot to analyze the text and determine the appropriate response based on the patterns learned from the training data. minimize it.

8.CLOUD SERVICES USED

For Deploying a chatbot like a Café Chatbot,Cloud services provide a range of capabilities that can enhance the bots functionality,scalability and manageability.The choice of cloud services can depend on various factors such as existing technology stacks,specific features and cost considerations.
Google Cloud Platform (GCP) provides a range of services tailored to cloud computing needs ensuring data management, application deployment and more. Here's an overview of some offerings :
BigQuery is a managed serverless data warehouse built for rapid SQL queries on large datasets. Utilizing Googles infrastructure it enables real time analytics making it ideal for in depth data analysis.
In addition to TensorFlow GCP boosts AI capabilities with tools like AutoML for automated machine learning processes the AI Platform for easy model deployment and pre trained APIs for vision, speech and language tasks.
Google App Engine delivers a managed serverless platform that eliminates infrastructure management requirements. Supporting programming languages such as Python, Java, Node.js and Go allows developers to concentrate on application development and deployment.

Cloud Spanner offers a horizontally scalable global database service. It empowers developers to build applications that offer worldwide availability and low latency data access.

The Google Kubernetes Engine, also known as GKE is a service that helps organize and oversee applications stored in containers making it easier to deploy, manage and scale them.

Google Cloud Platforms IAM features provide control measures that enable management of access, to specific resources and permissions for different actions thus enhancing security. Cloud Functions is a serverless computing service on Google Cloud Platform that allows running functions triggered by events without the need for managing servers. It supports programming languages. Is well suited for lightweight applications.

Google Cloud Platform offers networking options such as Dedicated Interconnect and Partner Interconnect to establish connections between on premises networks and the cloud. These cater to enterprises with networking requirements.

These services collectively enhance the flexibility, scalability and efficiency of cloud operations on Google Cloud Platform providing a solution, for IT needs.

Google Cloud Storage

A robust and flexible object storage service.

Data Storage : Save images, documents, settings and any type of file that chatbot requires.

Recovery : Maintain backups of the chatbots data and status for reliability and restoration needs.

Content Delivery : Deliver content to users with availability and global redundancy.

Google Cloud SQL and Firestore

Both are storage solutions for managing chatbot data each with features.

Cloud SQL: A managed relational database that supports MySQL, PostgreSQL and SQL Server. Suitable for data and complex queries.

Fire store : A NoSQL cloud database for storing and syncing data. It offers synchronization and offline support beneficial for real time chat applications.

9.THE DETAILS REQUIRED FOR UNDERSTANDING THE GCP STRUCTURE:

PROJECT ID : CLOUDPROJ-421621

COMMAND TO DEPLOY : GCLOUD INIT

To get started sign up for an account to access all Google Cloud services. Next download the Google Cloud SDK on your computer. This toolkit allows you to manage resources and applications hosted on Google Cloud.

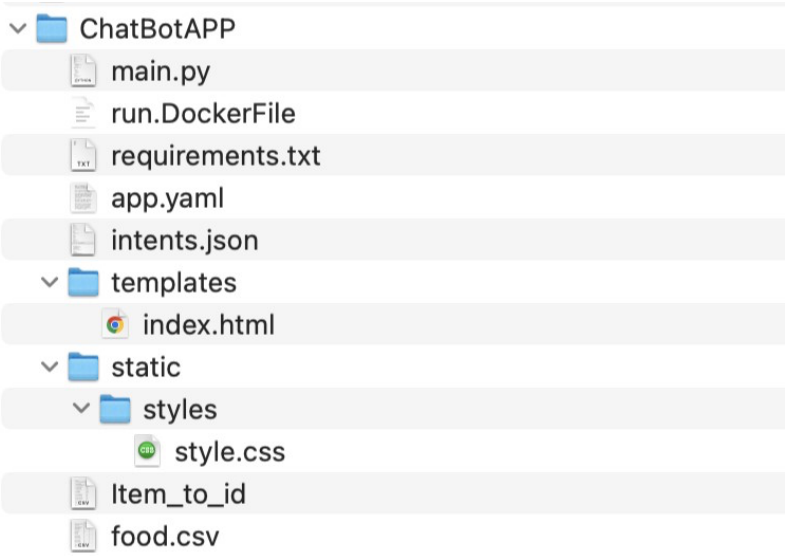
Once you have your account set up create a project, within it. This project will serve as a hub for all your application related resources. When setting up the project choose a name and ID to keep track of resources, billing and permissions.

In the project activate the App Engine API from the API library. App Engine is a managed platform that simplifies infrastructure management so you can concentrate on coding while it takes care of deployment, scaling and server management.

For Flask app structure :

Python Code (.py) : Develop the functionality of your Flask app using Python scripts. This includes defining routes, handling views and managing chatbot interactions.

App Engine Configuration (.yaml) : The app.yaml file plays a role, in App Engine deployments as it specifies the environment settings, runtime configurations and resource allocations.



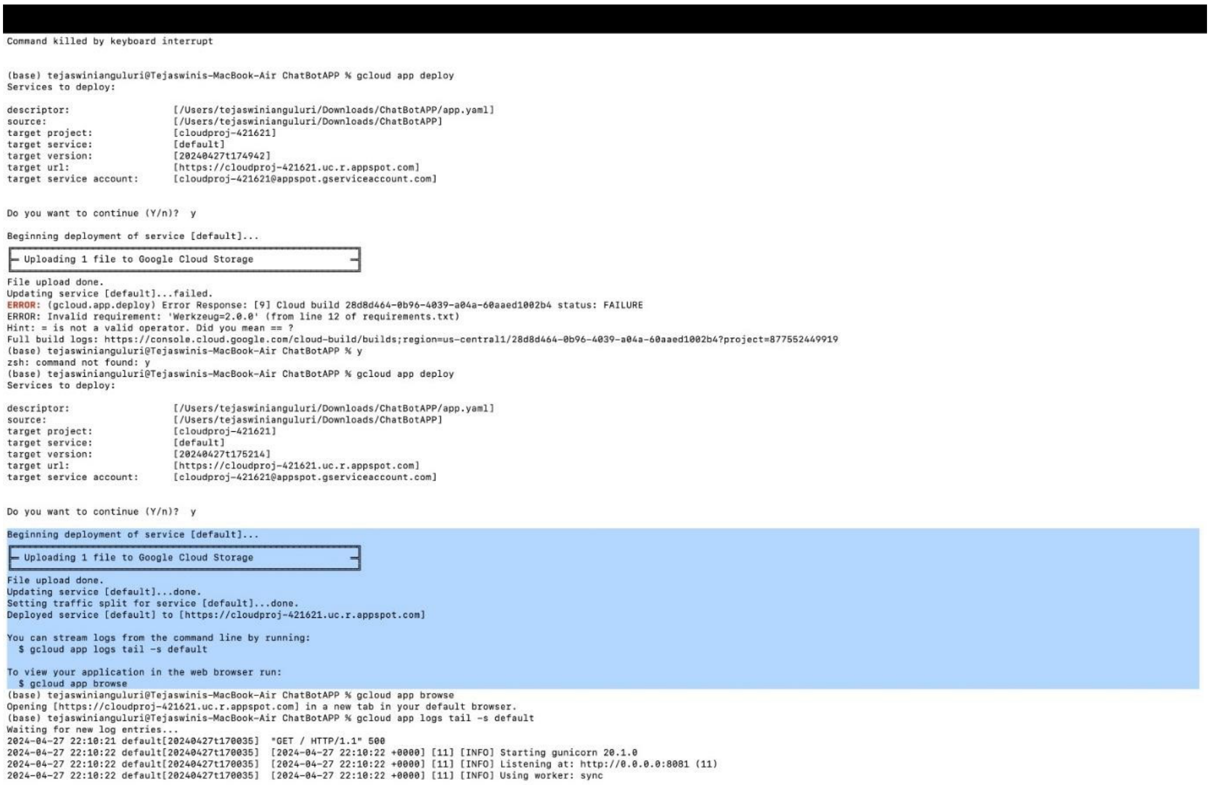
```
ChatBotAPP
    main.py
    run.DockerFile
    requirements.txt
    app.yaml
    intents.json
    templates
        index.html
    static
        styles
            style.css
    Item_to_id
    food.csv
```

Intents and Actions (.json): This file, in format contains a range of intentions linked to user interactions aiding the chatbot in understanding how to reply**.**

Website Templates and Design : The visual layout and style of your chatbots web interface are created using HTML and CSS files**.**

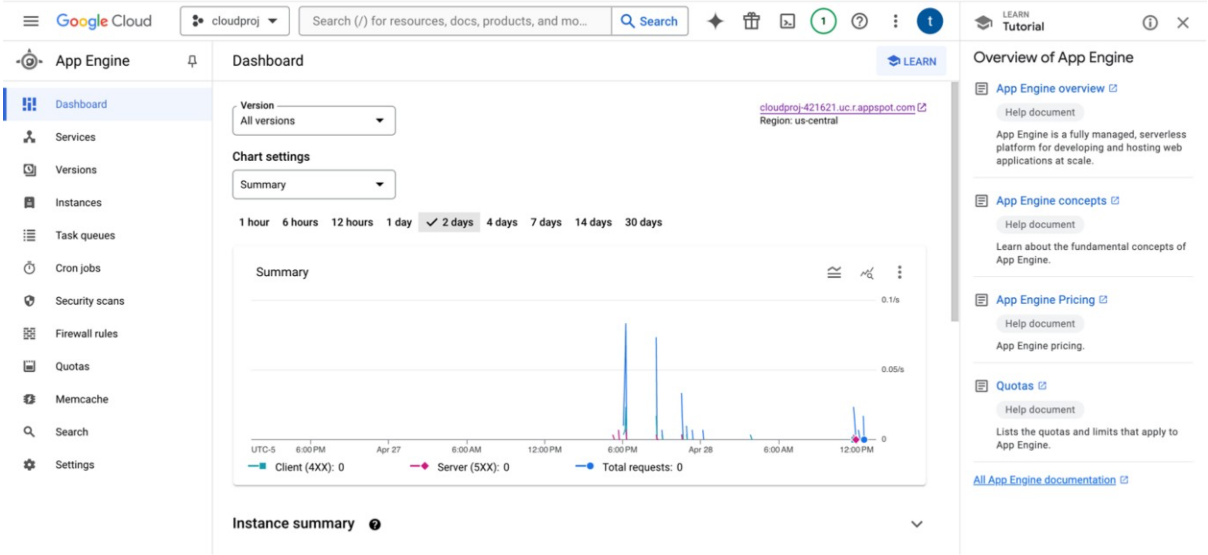
Required Libraries (requirements.txt) : A text document that enumerates all the Python libraries essential for your application, which GCP will utilize to replicate your apps environment in the cloud.

Data Storage (.csv): If your app utilizes CSV files, for storing data make sure to include them in your project structure**.**



[redacted]

```
Command killed by keyboard interrupt


(base) tejaswinianguluri@Tejaswinis-MacBook-Air ChatBotAPP % gcloud app deploy
Services to deploy:

descriptor:                  [/Users/tejaswinianguluri/Downloads/ChatBotAPP/app.yaml]
source:                      [/Users/tejaswinianguluri/Downloads/ChatBotAPP]
target project:              [cloudproj-421621]
target service:              [default]
target version:              [20240427t174942]
target url:                  [https://cloudproj-421621.uc.r.appspot.com]
target service account:      [cloudproj-421621@appspot.gserviceaccount.com]


Do you want to continue (Y/n)?  y

Beginning deployment of service [default]...
╔════════════════════════════════════════════════════════════╗
╠═ Uploading 1 file to Google Cloud Storage                 ═╣
╚════════════════════════════════════════════════════════════╝
File upload done.
Updating service [default]...failed.
ERROR: (gcloud.app.deploy) Error Response: [9] Cloud build 28d8d464-0b96-4039-a04a-60aaed1002b4 status: FAILURE
ERROR: Invalid requirement: 'Werkzeug=2.0.0' (from line 12 of requirements.txt)
Hint: = is not a valid operator. Did you mean == ?
Full build logs: https://console.cloud.google.com/cloud-build/builds;region=us-central1/28d8d464-0b96-4039-a04a-60aaed1002b4?project=877552449919
(base) tejaswinianguluri@Tejaswinis-MacBook-Air ChatBotAPP % y
zsh: command not found: y
(base) tejaswinianguluri@Tejaswinis-MacBook-Air ChatBotAPP % gcloud app deploy
Services to deploy:

descriptor:                  [/Users/tejaswinianguluri/Downloads/ChatBotAPP/app.yaml]
source:                      [/Users/tejaswinianguluri/Downloads/ChatBotAPP]
target project:              [cloudproj-421621]
target service:              [default]
target version:              [20240427t175214]
target url:                  [https://cloudproj-421621.uc.r.appspot.com]
target service account:      [cloudproj-421621@appspot.gserviceaccount.com]


Do you want to continue (Y/n)?  y

Beginning deployment of service [default]...
╔════════════════════════════════════════════════════════════╗
╠═ Uploading 1 file to Google Cloud Storage                 ═╣
╚════════════════════════════════════════════════════════════╝
File upload done.
Updating service [default]...done.
Setting traffic split for service [default]...done.
Deployed service [default] to [https://cloudproj-421621.uc.r.appspot.com]

You can stream logs from the command line by running:
  $ gcloud app logs tail -s default

To view your application in the web browser run:
  $ gcloud app browse
(base) tejaswinianguluri@Tejaswinis-MacBook-Air ChatBotAPP % gcloud app browse
Opening [https://cloudproj-421621.uc.r.appspot.com] in a new tab in your default browser.
(base) tejaswinianguluri@Tejaswinis-MacBook-Air ChatBotAPP % gcloud app logs tail -s default
Waiting for new log entries...
2024-04-27 22:10:21 default[20240427t170035]  "GET / HTTP/1.1" 500
2024-04-27 22:10:22 default[20240427t170035]  [2024-04-27 22:10:22 +0000] [11] [INFO] Starting gunicorn 20.1.0
2024-04-27 22:10:22 default[20240427t170035]  [2024-04-27 22:10:22 +0000] [11] [INFO] Listening at: http://0.0.0.0:8081 (11)
2024-04-27 22:10:22 default[20240427t170035]  [2024-04-27 22:10:22 +0000] [11] [INFO] Using worker: sync
```



Version: All versions

Chart settings: Summary

Client (4XX): 0 Server (5XX): 0 Total requests: 0

Overview of App Engine

App Engine overview
Help document
App Engine is a fully managed, serverless platform for developing and hosting web applications at scale.

App Engine concepts
Help document
Learn about the fundamental concepts of App Engine.

App Engine Pricing
Help document
App Engine pricing.

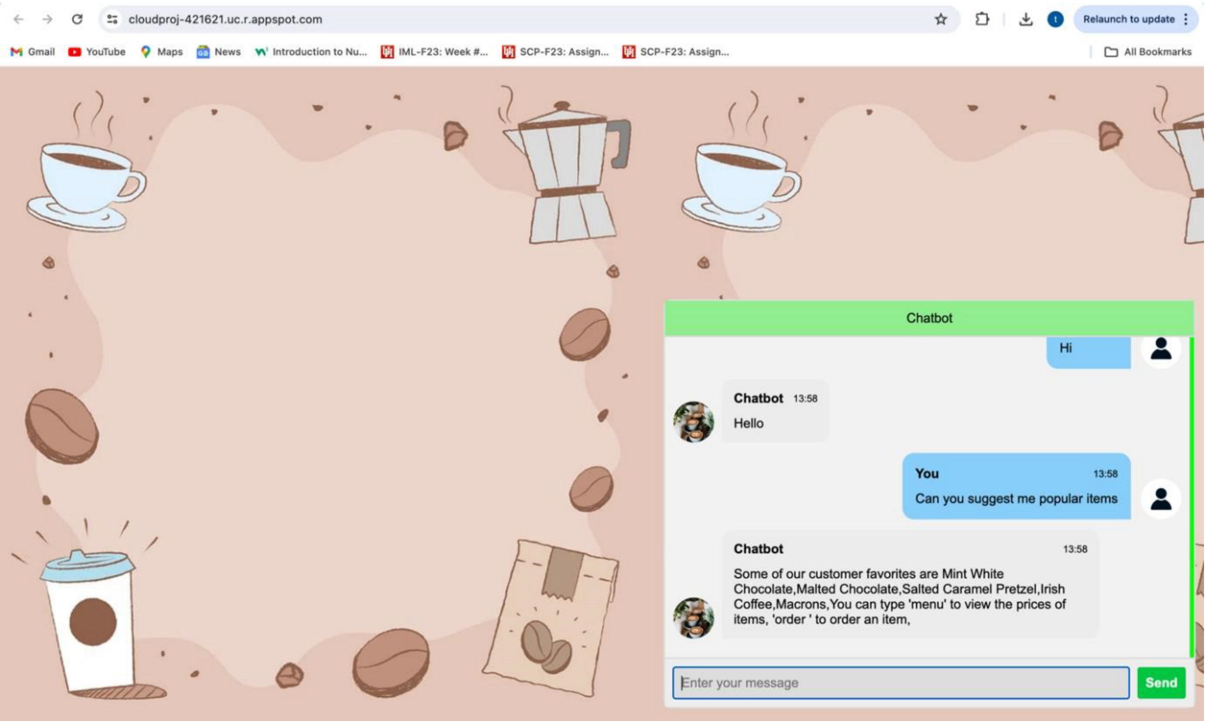
Quotas
Help document
Lists the quotas and limits that apply to App Engine.

All App Engine documentation

Access our chatbot using the URL provided once deployment is done

URL: https://cloudproj-421621.uc.r.appspot.com/

10.RESULTS



Hi

Chatbot 13:58

Hello

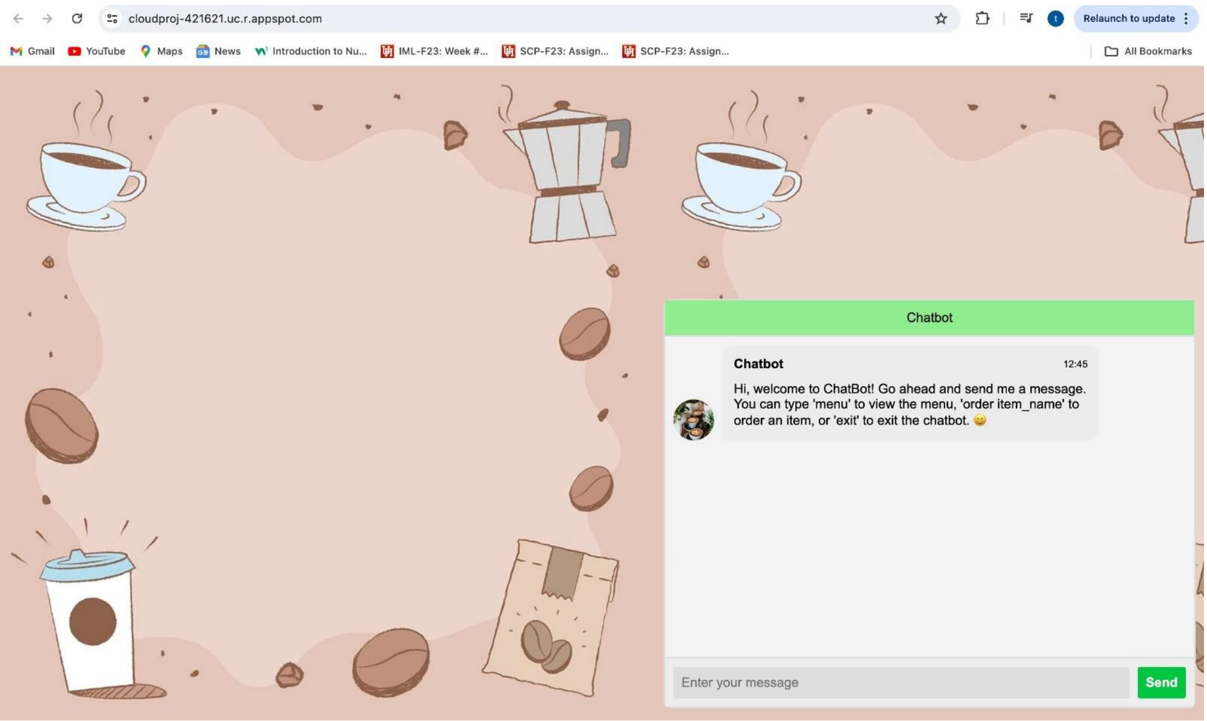
You 13:58

Can you suggest me popular items

Chatbot 13:58

Some of our customer favorites are Mint White Chocolate,Malted Chocolate,Salted Caramel Pretzel,Irish Coffee,Macrons,You can type 'menu' to view the prices of items, 'order ' to order an item,

Enter your message



Chatbot 12:45

Hi, welcome to ChatBot! Go ahead and send me a message. You can type 'menu' to view the menu, 'order item_name' to order an item, or 'exit' to exit the chatbot. 😀

Enter your message

11.CONCLUSION AND FUTURE WORK

Restaurant chatbots are changing the game in how restaurants engage with their customers. By using Natural Language Processing (NLP) these automated systems enhance customer service by simplifying order processes addressing questions and managing reservations. This technology boosts efficiency by relieving staff of tasks allowing them to focus on complex aspects of service.

As technology progresses restaurant chatbots are expected to advance offering features, like personalized suggestions based on orders communicating in multiple languages to cater to diverse clientele and handling challenging customer service situations with empathy and finesse.

Their increasing sophistication hints at a future where they play a role in hospitality than just being tools leading to deeper customer involvement and increased satisfaction levels. These chatbots are anticipated to become elements seamlessly integrated into restaurant operations and interactions, with customers.

In the advancements of restaurant chatbots the main focus will be, on improving customer personalization by integrating AI for making predictive suggestions. Emotional AI might be used to understand customer emotions and provide tailored responses. Moreover expanding to platforms and incorporating voice recognition and instant language translation will enhance the accessibility and versatility of chatbots. Furthermore introducing reality could transform the way menus are displayed making these systems crucial, for modernizing customer service in the hospitality sector.