Introduction to AI COM727 Software Prototype and with Technical Report (AE2)

Your student number + Name:	Q15817059 Sathya Priya Mahendran
Your team member's name if applicable (for the implementation of the prototype)	Ann Mariya Tom Uma Mani Nair Utsavpuri Kiranpuri Goswami Yash Yogeshkumar Vaghela

Contents

Introduction

Artificial Intelligence is an advanced technology where machines are trained to learn, memorize and mimic the human intelligence. As everyone predicts, the future will completely be an AI-enabled modern world. AI technology is providing lot of benefits but inevitably there are negative significances too.

One of the major legal and ethical issue that confront society due to AI is privacy and surveillance, it amplifies the ability to use peculiar data and privacy interests by raising scrutiny to new levels of power and speed. Next is AI bias which is caused while developing in-built algorithms, in order to prevent it, the algorithms should be developed and trained responsibly. The foremost professional issue by AI is loss of certain jobs (i.e.) there may be a chance of machines taking over the jobs of people. Our Train Bot is the software product which has been built by our group and it does not have negative significances in flaw of the culture. It is an independent chatbot which can analyze user's requests and provide appropriate responses.

Need for your prototype

The Train Bot chatbot is particularly designed for the passengers who are in search of data like upcoming train timings, ticket fare and stations between arrival and departure points. Our chatbot will be able to communicate these data to the users which will be very helpful for them before starting their train travel. This is an easy-to-use chatbot, mainly used be passengers who are in hurry and does not need to deep dive-in the complete railway website for search of these significant info instead can interact with our chatbot and get responses instantly.

Statement of the problem

Train Bot respond to tedious tasks with any degree of dissatisfaction to the users and also achieve speed-up response times. They're programmed to do the task and fulfill user's expectations. Our chatbot eliminates deep and tiresome search throughout the website for significant information like upcoming train timings, ticket fare and stations between arrival and departure points whereas it will respond to the user's queries specifically.

Aims and Objectives

The purpose of our project is to reduce the stress for the user who are in the hurry to catch next train and reduce the time spending for the search in website. Also, our AI chatbot symbolizes like a human, interact with the users and provides real-time data without direct human control via text channels through website.

Proposed Solution – Train Bot

We have designed the Train Bot mainly focusing on the passengers using railway transportation. Our chatbot will be providing very speedy and crisp mechanism. It is a user-friendly bot. It gives approval of the user query reducing confusion. The conversational interface of a chatbot is appropriate for fetching the information of the Railways and there is a lot of space for improving and fine tuning the services provided.

Prototype Design

Previously, I used to book a ticket in IRCTC (Indian Railway) chatbot and felt that it is not a user-friendly, because it was developed completely with rule-based chatbot where I felt it as robotic without any static data. Normally, I expect only the accurate and concise output from the chatbot but when I use IRCTC Disha 2.0 bot, it is literally leading to deep-diving conversations which is time consuming. Considering those potential

problems, we have created the Train Bot which provides accurate and instant output by analyzing the input data. Our prototype design (Figure.1. System Architecture), we have explained the list of packages we used in this project.

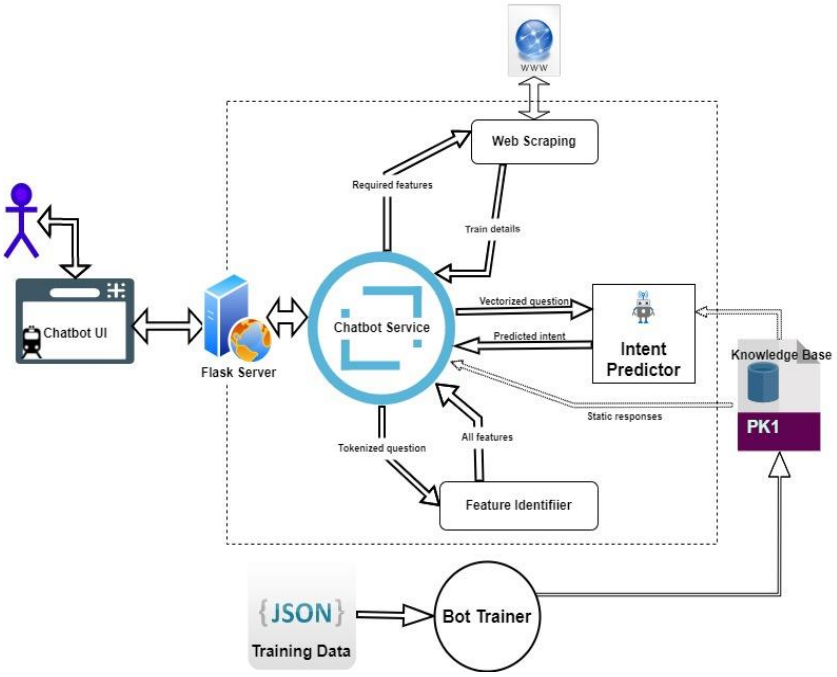

Figure.1. System Architecture

Prototype Development and AI Algorithms used

Our project "Train Bot" was developed by using the machine learning algorithm Multinomial Naive Bayes. This procedure is a Bayesian method for implementing Natural Language Processing (NLP). This algorithm presumptions an identifier the writing, via Bayes theorem. It estimates respective identifier's probability for a specified input and results to identifier with an utmost possibility. Using Bayes theorem to calculate the probabilities between two fruits: class Banana itself is accessible, we estimate the probability of class Apple. It's based on the formula below: P(Apple|Banana) = P(Apple) * P(Banana|Apple)/P(Banana). Multinomial

distribution generally involves number feature amounts. However, in practice, fractional counts such as Tfidftransformer (tf-idf) may also execute. The following steps refers to the implementation of the model:

- To develop our Train Bot website friendly, we used Graphical User Interface (GUI) with the help of flask server to host our chatbot service. Flask server is a software server which is expert in executing HTTP requests on the public world wide web, private LAN and WANs. Flask session support Server-side Session to your application. The information that is essential to be preserved in the cookie session for a momentary storage on the machine. In our Train Bot, flask session saves user's data for 5 minutes in order to provide output from the previous user request. We are also using the static method to pull the data from the JSON via pickle module, so there is no need of identification mechanism for the static set of questions.

```
from flask import Flask, render_template, request, session
from flask_session import Session
import json
import time
from datetime import timedelta

from chat_bot import ChatBot# Importing from chat_bot.py

# API configurations begins
chatbot_app = Flask(__name__, template_folder="templates", static_folder="static")
chatbot_app.config["SESSION_PERMANENT"] = True
chatbot_app.config['PERMANENT_SESSION_LIFETIME'] = timedelta(minutes=5)
chatbot_app.config["SESSION_TYPE"] = "filesystem"
Session(chatbot_app)
```

Figure.2. Flask Session Package

- Web scraping is the scheme of congregation and scrutinizing raw data from the web, and the Python community has come up with some dominant web scraping tools. From that, we have used the packages requests and beautiful soup for scraping and parsing data from the web. In our prototype, we have used web scraping mechanism to extract data from the live railway website https://ojp.nationalrail.co.uk/service/planjourney/search

```
import requests
import time
from bs4 import BeautifulSoup

from flask import session
```

Figure.3. Web Scraping Tools

- Feature identifier is used to identify the key words from the text so it will be easier for the system to predict the question for providing the response. NumPy random package is used to choose multiple random items from a list without repetition or duplicates whereas in our bot, it is used to pull the responses randomly without any repetitions.
- NLTK is a leading platform developed for executing with NLP on Python. It offers audience numerous input processing libraries with a huge number of test scenarios. A variability of method can be executed using NLTK such as tokenizing, parse tree visualization, etc. We have used Wordnet Lemmatizer in our project whereas we first tokenize the sentence into words using word tokenize and then we will call lemmatizer on each word. This can be done in a list command. We have also installed stop words from NLTK package in order to safely ignore the words which does not add much meaning to the sentence and also this can be done without sacrificing the meaning of the sentence. Such words are already saved in package named corpus.

  Scikit-learn is undoubtedly great library for machine learning in Python. The sklearn library have a number of actual tools for machine learning and arithmetical modeling including classification, regression, clustering and dimensionality reduction. In our plan, we have used vectorization to convert the text data into numerical vectors for checking the details from the pickle module in order to utilize the NLTK and it makes the system to understand the machine language easier.

```
import nltk
from nltk.stem import WordNetLemmatizer as Lemmatizer
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
```

Figure.4. NLTK and Scikit-learn Packages

```
import pickle

responses = pickle.load(open("responses.pk1", "rb"))  # Read the static responses for greetings and gratitude
```

Figure.5. Pickle Module

- In Intent Predictor, we have imported the word tokenize method from the NLTK package to split a sentence into tokens or words which is helpful to match the keywords and provide response to the user. The datetime is a package which is used to provide functions to deal with dates, times and time intervals whereas this package plays a major role in our project. Also, we have included dateutil module to display time zone using the datetime module which does not support as it would be done in the time module.

```
from nltk import word_tokenize
```

Figure.6. Tokenize Method

```
date_time = dparser.parse(question, fuzzy=True)
date = date_time.date().strftime("%d%m%Y")
time = date_time.time().strftime("%H%M")
features["travel_date"] = date
features["travel_time"] = time
```

Figure.7. Date and Time Features

- Training data are stored using JavaScript Object Notation (JSON) which consists of pre-defined set of data from user inputs and repose to query in quicker way because of the JSON.

Evaluation

We tested our project by using the below testing scenarios, this will lead our system to provide the accurate live data.

- Once the program is executed, you will able to see Train Bot web page with bot icon.

Figure.8. Output

- When you clicked the bot icon, you are welcomed with welcome message and waiting for your input. When you provide the same input frequently output might vary from our chatbot.

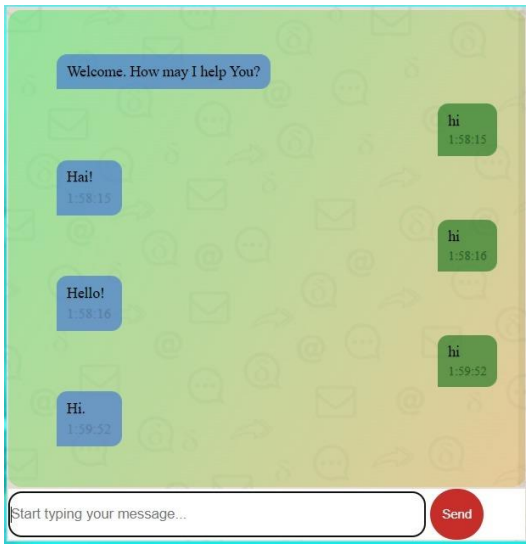

Figure.9. Chatbot showing Static Information

- Also, based on our question, system starts to provide live update by using the web scrapping. Right now, our system is designed to process only the Begin and End greeting, gratitude, time for the destination train, ticket price and stops between.

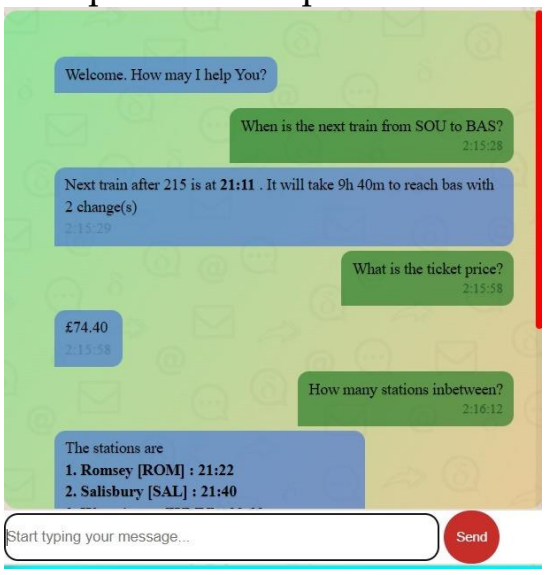

Figure.10. Train Bot

Limitation

- As our prototype Train Bot is programmed in a way that they only know what they are trained and cannot recognize humans all context which may lead to an annoy the passenger and still need to include more set of question to our system to provide answer to all set of queries. We still need to update some architecture to hold the large set of data.
- The challenge we faced during our project is web scrap integration we couldn't able to populate the respective output properly because we didn't provide the proper feature identification to the system and it took long way to sort out the correct identification to get the required output from the web scrap.

Conclusion

This report fruitfully enlightens and executed Train bot which can be used to get the significant statistics from the Railway. By using the user frequent question, we still add lot many futures, we will sort out the existing bugs and include the railway suggestions and planning to include booking data via bot. The key to active chatbot use is to make sure they enrich your user's satisfaction. This chatbot will value the users by valid time, reduce stress and unwanted internet usage.

References

1) Androutsopoulou, A., Karacapilidis, N., Loukis, E. and Charalabidis, Y., 2019. Transforming the communication between citizens and government through AI-guided chatbots. *Government Information Quarterly*, 36(2), pp.358-367.
2) B, A. and E, A., 2015. Alice chatbot: Trials and outputs. Computaci´on y Sistemas, pp.625–632.

3) Bernard Marr. 2022. What Are the Negative Impacts of Artificial Intelligence (AI)? | Bernard Marr. [online] Available at: https://bernardmarr.com/what-are-the-negative-impacts-of-artificial-intelligence-ai/.
4) Bird, S. and Loper, E., 2022. NLTK: The Natural Language Toolkit. [online] ACL Anthology. Available at: <https://aclanthology.org/P04-3031>
5) C. Müller, A. and Guido, S., 2017. Introduction to Machine Learning with Python. 1st ed. CA: O'Reilly Media, Inc.,
6) Communication and Applications. (n.d.). Springer, pp.561–571.
7) Conference on Information and Education Technology. (n.d.). pp.111–119.
8) F, C., M, D., M, L., F, P., A, P. and S, L., 2018. Chatbot fore-learning: A case of study. 7th ed., International Journal of Mechanical Engineering and Robotics Research, pp.528–533.
9) Juan, A. and Ney, H., n.d. Reversing and Smoothing the Multinomial Naive Bayes Text Classifier.
10) Khan, V. and Meenai, T.A. (2021). Pretrained Natural Language Processing Model for Intent Recognition (BERT-IR). Human-Centric Intelligent Systems, 1(3-4), p.66. doi:10.2991/hcis.k.211109.001.
11) Nurshatayeva, A., Page, L., White, C. and Gehlbach, H., 2020. Proactive student support using artificially intelligent conversational chatbots: The importance of targeting the technology.
12) S, B. and E, L., 2004. NLTK: The natural language toolkit, in 'Proceedings of the ACL Interactive Poster and Demonstration Sessions', pp.214–217.
13) S, B. and E, L., 2004. NLTK: the natural language toolkit, Association for Computational Linguistics.
14) S-T, L., F-Y, L. and J-W, L. (n.d.). A banking chatbot security control procedure for protecting user data security and privacy. International Conference on Broadband and Wireless Computing.

15) Su, J., Sayyad-Shirabad, J. and Matwin, S., n.d. Large Scale Text Classification using Semi-supervised Multinomial Naive Bayes.
16) Techwithtim.net. 2022. [online] Available at: https://www.techwithtim.net/tutorials/ai-chatbot/.
17) W, M., T, P., K, P., S, P., P, C. and T, T. (2019). A survey on evaluation methods for chatbots, in 'Proceedings of the 2019'.
18) W, W., 2018. Artificial intelligence modules (cmp6040, 7028) coursework specification.