

# Web Scraping

## Introduction

The provided Python script is designed to automate the collection, cleaning, and analysis of text data from web pages. This script utilizes several popular libraries such as BeautifulSoup for web scraping, NLTK for natural language processing (NLP), and scikit-learn for feature extraction via TF-IDF vectorization.

In this paper, I'll be covering Ratan Tata, a well-known businessman of the Tata Group conglomerate.

The Starter URL Link : [https://en.wikipedia.org/wiki/Ratan\\_Tata](https://en.wikipedia.org/wiki/Ratan_Tata)

To run the program you need to change the input\_directory according to the machine.

The ultimate goal of the script is to build a knowledge base from scraped web content by identifying and extracting the most relevant terms and sentences related to a specific topic or entity, in this case, Ratan Tata.

## Web Scraping and Data Collection

The script begins by scraping web pages for content. It uses the **requests** library to fetch pages and BeautifulSoup to parse the HTML content. The initial step involves scraping a starter URL, specifically a Wikipedia page, to collect URLs of related articles. These URLs are then written to a file for later processing.

Key Functions:

- **scrape\_and\_save(urls)**: This function iterates through a list of URLs, scrapes the content of each page, and saves the text of the page to a file. It extracts text specifically from paragraph tags, which are more likely to contain relevant information, and saves each document with a unique filename based on its order in the list.

---

```
https://en.wikipedia.org/wiki/Ratanji_Tata
https://en.wikipedia.org/wiki/Ratanji_Dadabhoy_Tata
https://en.wikipedia.org/wiki/Bombay
https://en.wikipedia.org/wiki/Bombay_Presidency
https://en.wikipedia.org/wiki/British_Raj
https://en.wikipedia.org/wiki/Mumbai
https://en.wikipedia.org/wiki/Maharashtra
https://en.wikipedia.org/wiki/India
https://en.wikipedia.org/wiki/Cornell_University
https://en.wikipedia.org/wiki/Bachelor_of_Architecture
https://en.wikipedia.org/wiki/Tata_Sons
https://en.wikipedia.org/wiki/Tata_Group
https://en.wikipedia.org/wiki/J._R._D._Tata
https://en.wikipedia.org/wiki/Cyrus_Mistry
https://en.wikipedia.org/wiki/Natarajan_Chandrasekaran
https://en.wikipedia.org/wiki/Naval_Tata
https://en.wikipedia.org/wiki/Tata_family
https://en.wikipedia.org/wiki/Order_of_Australia
https://en.wikipedia.org/wiki/Assam_Baibhav
https://en.wikipedia.org/wiki/Order_of_British_Empire
https://en.wikipedia.org/wiki/Padma_Vibhushan
End of crawler
```

---

## Data Cleaning

Once the content is saved, the script cleans the text files to prepare them for analysis. This involves converting text to lowercase, removing punctuation, tokenizing the text into words, removing stopwords, and lemmatizing the remaining words.

Key Functions:

- **clean\_text\_files(input\_directory):** Reads each text file, performs the aforementioned cleaning steps, and saves the cleaned text to a new file. This process prepares the data for the next step, which involves extracting meaningful insights through TF-IDF vectorization.

## Feature Extraction and Analysis

After cleaning, the script uses the TF-IDF vectorizer from scikit-learn to identify the most relevant terms across all documents. It computes the TF-IDF scores for each term in each document, aggregates these scores across all documents, and identifies the top terms.

Key Functions:

- **extract\_top\_terms\_from\_all\_files(directory, top\_n=40):** Identifies the top N terms with the highest TF-IDF scores across all cleaned text files. This step is crucial for understanding the key themes or topics present in the collected data.

---

```
Top 40 terms across all documents are: ['tata', 'india', 'mumbai', 'indian', 'state', 'company', 'bombay', 'mistr
y', 'son', 'british', 'award', 'order', 'city', 'cornell', 'university', 'ratanji', 'also', 'maharashtra', 'famil
y', 'chairman', 'year', 'government', 'architecture', 'first', 'assam', 'steel', 'group', 'australia', 'sir', 'rata
n', 'trust', 'two', 'honour', 'service', 'national', 'institute', 'world', 'presidency', 'million', 'member']
```

---

## Sentence Extraction and Knowledge Base Construction

Finally, the script searches for sentences containing the top identified terms within the original, non-cleaned content files. This is aimed at constructing a knowledge base where each key term is associated with sentences that provide context or information about that term.

Key Functions:

- **find\_sentences\_with\_terms\_in\_content\_files(directory, terms):** Finds and aggregates sentences containing each of the top terms from the original content

files. This results in a dictionary where each term is mapped to relevant sentences, encapsulating a knowledge base of extracted information.

'[tata': ['[16][206][207][208][17] Many of India's numerous conglomerates (including Larsen & Toubro, State Bank of India (SBI), Life Insurance Corporation of India (LIC), Tata Group, Godrej and Reliance),[149] and five of the Fortune Global 500 companies are based in Mumbai.'  
'[318]\n Electricity is distributed by the Brihanmumbai Electric Supply and Transport (BEST) undertaking in the island city, and by Adani Transmission,[319] Tata Power and the Maharashtra State Electricity Distribution Co. Ltd (Mahavitaran) in the suburbs.'  
'[325] Mobile phone coverage is extensive, and the main service providers are Vodafone Essar, Airtel, MTNL, Loop Mobile, Reliance Communications, Idea Cellular and Tata Indicom.'  
'Mumbai is also home to the National Institute of Industrial Engineering (NITIE), Jammalal Bajaj Institute of Management Studies (JBIMS), Narsee Monjee Institute of Management Studies (NMIMS), S P Jain Institute of Management and Research, Tata Institute of Social Sciences (TISS) and several other management schools.'  
'Mumbai is home to two prominent research institutions: the Tata Institute of Fundamental Research (TIFR), and the Bhabha Atomic Research Centre (BARC).'  
'[437] Prominent DTH entertainment services in Mumbai include Dish TV and Tata Sky.'  
'\n Ratanji Dadabhoi Tata (1856-1926) was an Indian businessman who played a pivotal role in the growth of the Tata Group in India.'  
'He was one of the partners in Tata Sons founded by Jamsetji Tata.'  
'Ratanji was the father of J. R. D.\n Tata.'  
'Under the name Tata & Co, Ratanji ran an opium importing business in China, which was legal at the time.',

## Saving the Knowledge Base

The knowledge base, a dictionary of terms and their associated sentences, is then serialized and saved using Python's **pickle** module. This serialized form can be easily loaded for future analysis or processing.

## Conclusion

This Python script demonstrates a comprehensive approach to web scraping, data cleaning, text analysis, and knowledge extraction. By leveraging powerful libraries and algorithms, it efficiently processes web content to construct a valuable knowledge base. Such a base can be instrumental in various applications, from content analysis and information retrieval to building datasets for machine learning models.

## 2. ChatBot

### System Description

The chatbot leverages various NLP techniques and algorithms to understand and respond to user inputs. Key components include:

The code outlines a sophisticated chatbot leveraging Natural Language Processing (NLP) to enhance user interaction through dynamic conversations. It employs advanced techniques like TF-IDF vectorization for generating contextually relevant responses from a pre-defined knowledge base. Additionally, the system manages user models to store preferences, likes, dislikes, and personal information, ensuring tailored interactions. Spacy's NLP capabilities are utilized for text analysis, including keyword extraction and personal information identification, which significantly improves the chatbot's ability to understand and respond to user queries accurately.

1. **Spacy for Text Processing:** Utilizes the **spacy** library for linguistic features extraction, including lemmatization, POS tagging to understand user inputs and extract personal information or preferences.

Which can be described as the response from the chat Bot.

It also uses the POS tagging and lemmatizations and linguistic feature extraction to update the user model for likes , dislikes and personal information.

2. **TF-IDF Vectorization for Response Selection:** Employs **TfidfVectorizer** from **scikit-learn** to transform text inputs into a TF-IDF matrix, enabling the identification of the most relevant responses from the knowledge base through cosine similarity.

When user gives some input it is pre-processing the text and extracts the keywords to find the cosine similarity from the knowledge base and returns the sentence base on the score.

3. **NLTK for Tokenization and Stopwords Removal:** Uses **nltk** for tokenizing user inputs and removing stopwords, facilitating cleaner and more focused text for processing.
4. **Personalized User Modeling:** Dynamically updates user models based on interactions to store and utilize personal information, likes, and dislikes for tailoring conversations.

When User enters the user id it recognises the user if the he/she is returning user and connects to its original json file as user model.

Otherwise it creates new json file for the new user on name of the User ID and stores the values accordingly in the corresponding json file

5. **Regex for Text Cleaning:** Implements regular expressions to preprocess and clean the chatbot's responses, ensuring clarity and relevance in communication.

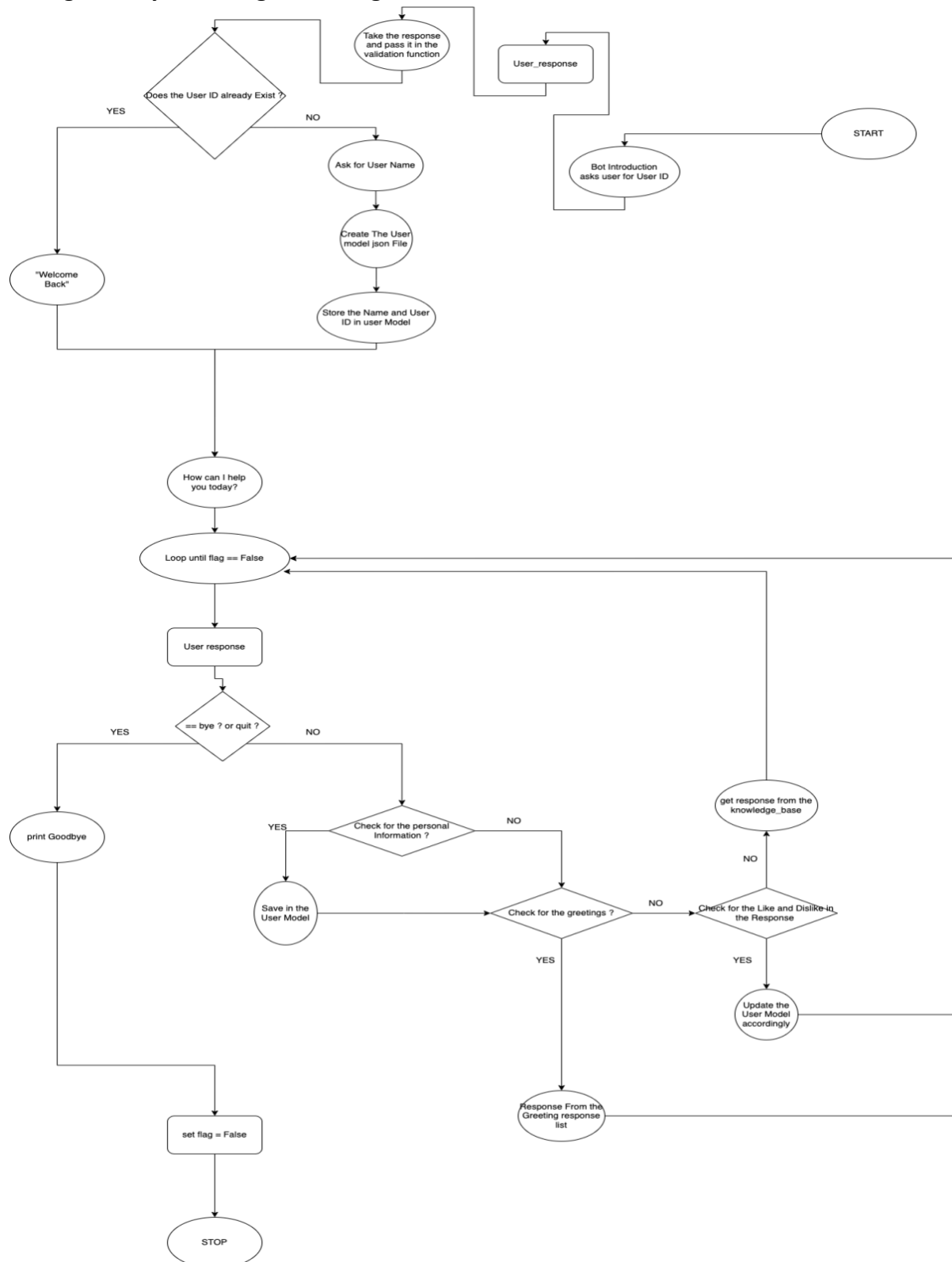
We have used some regex implementation to remove the unnecessary elements from the bot response.

Regex : `r'\[\d+\]`

6. **Named Entity Recognition (NER):** NER is utilized through the Spacy library, specifically using the `en_core_web_sm` model. This model is designed to recognize various types of entities in text, such as names of people (PERSON), organizations (ORG), locations (LOC), dates (DATE), products (PRODUCT), and more.

```
entity_labels = [  
    'PERSON', # People, including fictional  
    'GPE',   # Countries, cities, states  
    'ORG',   # Companies, agencies, institutions  
    'DATE',  # Absolute or relative dates or periods  
    'LOC',   # Non-GPE locations, mountain ranges, bodies of water  
    'PRODUCT', # Objects, vehicles, foods, not services  
    'EVENT', # Named hurricanes, battles, wars, sports events, etc.  
    'WORK_OF_ART', # Titles of books, songs, etc  
    'LANGUAGE', # Any named language  
    'FAC'    # Buildings, airports, highways, bridges, etc.  
]
```

A diagram of your dialog tree or logic:



## Dialogue Interaction:

```
(py) denish@denish21 Project1_3 % python chatbot.py
Welcome to the Chatbot Service!
Please enter your user ID or type 'new' to create a new session:
new
Enter a new user ID to start:
DK
Chatbot: Hi there! What's your name?
You: Denish
Denish: Hello
Chatbot: Good day to you! Denish!
Denish: Who is ratan tata ?
Chatbot: He is the father of Ratan Tata, Jimmy Tata and Noel Tata.
Denish: from where ratan tata graduated ?
Chatbot: Tata graduated from Elphinstone College in Bombay in 1858.
Denish: who runs tata steel ?
Chatbot: Sir Dorabji established the Tata Iron and Steel Company (TISCO), now known as Tata Steel in 1907.
Denish: where is the office of tata group situated ?
Chatbot: The Tata Group (/ˈtɑːtoː/) is a group of companies headquartered in Mumbai.
Denish: i like it
Chatbot: Noted. You like group, companies.
Denish: in which countries tata group is doing their business ?
Chatbot: Tata Sons is the owner of the Tata name and the Tata trademarks, which are registered in India and several other countries.
Denish: []
```

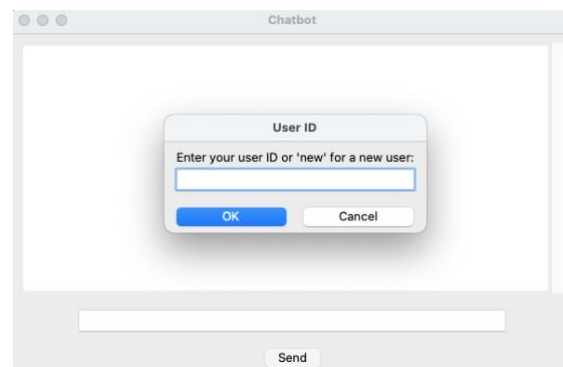
Here in this picture you can see that the chat bot greets me well first with my name. It also creates json file according to my user ID. Then I have asked several question regarding the topic of Ratan Tata and Tata Group and It has given me almost each and every answer for corresponding questions.

At last it greets well for the day and program ends:

Denish: quit

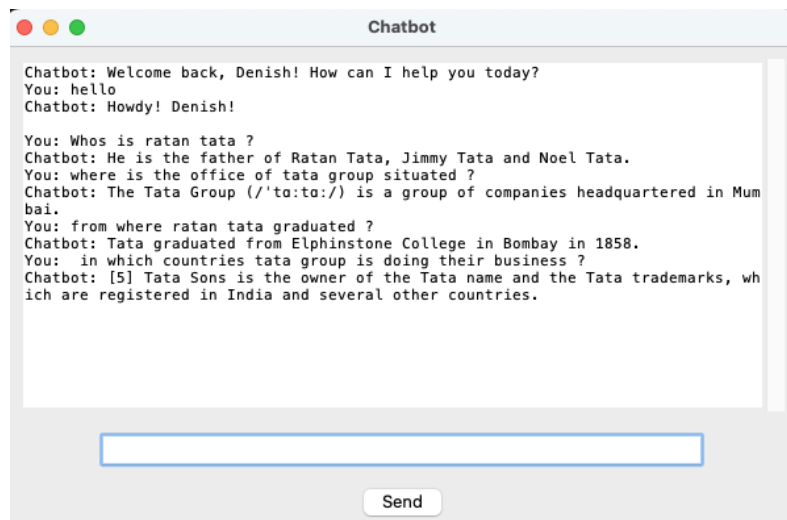
Chatbot: Bye Denish! Have a great day!

I have introduced the chat\_bot with GUI as well. To run the GUI we just need to comment out the last line of the code in chat\_bot.py named as start\_chat\_session(). This is the main running function of the chatbot.



**GUI** is made using tkinter library in Python. This GUI has the same functionality as the normal program chatbot.py has.

Also, It's dynamic look can be more interactive in terms of interaction with chatbot.



The chatbot's system processes user inputs, updates user models, and selects appropriate responses. The dialogue flow can be broadly outlined as follows:

1. **Greeting and User Identification:** The chatbot starts by greeting the user and identifying or creating a new user model based on the provided user ID.

Responses for the Greetings are Predefined and can be used when there are such trigger words are given.

2. **Input Processing and Personal Information Extraction:** Inputs are analyzed for personal details using NER, which are then stored in the user's model.

These NER and Information Systems are included to perform the operations on User model.

3. **Preference Learning:** Likes and dislikes are inferred from user inputs using keyword extraction and context analysis, updating the user model accordingly.
4. **Response Generation:** Utilizes the knowledge base and user model to generate relevant responses, employing TF-IDF vectorization and cosine similarity for content retrieval from the given knowledge\_base.
5. **Feedback Loop for Preferences:** Adjusts the user model based on the user's reactions to previous bot responses, refining future interactions.



An appendix for the knowledge base (and live lookup) you created with samples:

```
import pickle

# Replace 'your_pickle_file.pkl' with the actual file name
filename = 'knowledge_base.pkl'

# Open the file in binary read mode
with open(filename, 'rb') as file:
    data = pickle.load(file)

# Now, 'data' holds the deserialized object that was stored in the pickle file
print(data)
```

{'tata': ["[16][206][207][208][17] Many of India's numerous conglomerates (including Larsen & Toubro, State Bank of India (SBI), Life Insurance Corporation of India (LIC), Tata Group, Godrej and Reliance),[149] and five of the Fortune Global 500 companies are based in Mumbai.", '[318]\n Electricity is distributed by the Brihanmumbai Electric Supply and Transport (BEST) undertaking in the island city, and by Adani Transmission,[319] Tata Power and the Maharashtra State Electricity Distribution Co. Ltd (Mahavitaran) in the suburbs.', '[325] Mobile phone coverage is extensive, and the main service providers are Vodafone Essar, Airtel, MTNL, Loop Mobile, Reliance Communications, Idea Cellular and Tata Indicom.', 'Mumbai is also home to the National Institute of Industrial Engineering (NITIE), Jammalal Bajaj Institute of Management Studies (JBIMS), Narsee Monjee Institute of Management Studies (NMIIMS), S P Jain Institute of Management and Research, Tata Institute of Social Sciences (TISS) and several other management schools.', 'Mumbai is home to two prominent research institutions: the Tata Institute of Fundamental Research (TIFR), and the Bhabha Atomic Research Centre (BARC).', '[437] Prominent DTH entertainment services in Mumbai include Dish TV and Tata Sky.', '\n Ratanji Dadabhoi Tata (1856–1926) was an Indian businessman who played a pivotal role in the growth of the Tata Group in India.', 'He was one of the partners in Tata Sons founded by Jamsetji Tata.', 'Ratanji was the father of J. R. D.\nTata.', 'Under the name Tata & Co, Ratanji ran an opium importing business in China, which was legal at the time.', '[2]\n Tata Steel was conceived and commissioned by Jamsetji Tata.', 'Ratanji played an important role in the completion of the Tata Steel Project along with Jamsetji's son Dorab and thus Tata Steel was established in Jamshedpur.', 'However, after the war Tata Steel went through a difficult period in the 1920s as steel was dumped into India from Britain and Belgium.', 'Ratanji, along with other directors successfully sought protection for the Indian steel industry from the colonial government of the day and headed the creation of Tata Steel.', 'The first British-owned Tata Steel plant in India was established in 1911, on September 1911']

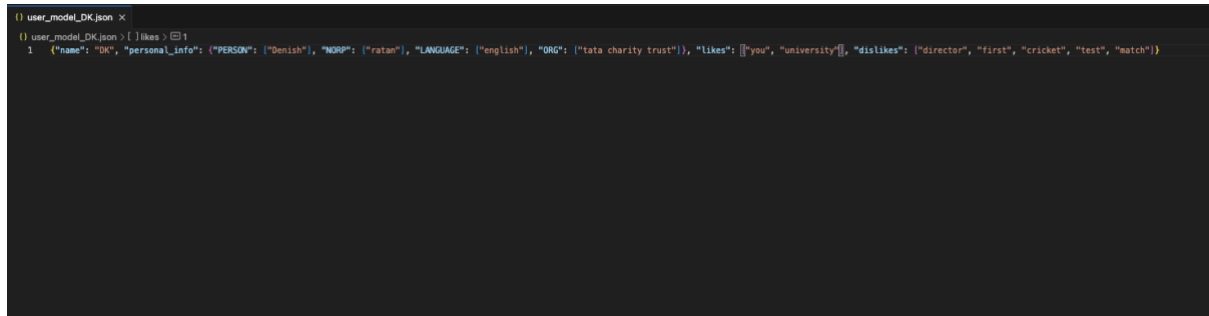
Here you can see that I have saved knowledge\_base as knowledge\_base.pkl file. It is stored for 15 key words according to my domain knowledge.

```
knowledge_base = ['tata', 'india', 'mumbai', 'indian', 'state', 'company', 'bombay', 'mistry', 'british', 'award', 'cornell', 'university', 'ratanji', 'maharashtra', 'family', 'chairman', 'steel', 'group', 'ratan']
```

The knowledge base constructed for the chatbot system is tailored to the domain centered around significant Indian entities and personalities, primarily focusing on the Ratan Tata, Tata Group of Company and its founder.

It encapsulates pivotal keywords such as 'Tata', 'India', 'Mumbai', and 'Ratanji', amongst others, which serve as anchors for retrieving detailed information relevant to user queries. This data-rich repository forms the backbone of the chatbot's response mechanism, empowering it to deliver precise and context-aware information across a spectrum of topics related to the Tata Group, its historical context within India, key locations, and notable figures associated with the company's legacy.

## An appendix for sample user models that were created :



If User already exist then it should also has its own json file and if the user is new then it generates the new json file according to the User ID.

In this figure we can see it generates the JSON file for the User Model in which it has. Stored the name, Personal Information List, Likes and Dislikes.

In the chatbot system I have used **Named Entity Recognition (NER)** which plays a crucial role in parsing and understanding user input.

NER implementation leverages the Spacy library, which is known for its efficiency and accuracy in identifying entities within text. By recognizing entities such as 'PERSON', 'GPE', 'ORG', and others, the chatbot can extract meaningful pieces of information from user dialogues.

This capability is pivotal for personalizing interactions, as it allows the chatbot to maintain context and reference relevant details that contribute to a more engaging experience.

The user models are structured to capture and store individual user data, including their name, personal likes and dislikes, and other relevant information.

This structured information enables your chatbot to deliver personalized content and responses, enhancing the user's experience over time for future development. As users interact with the chatbot, their models become more detailed, allowing the chatbot to anticipate needs and preferences, thus fostering a relationship and improving the overall effectiveness of service.

The NER component of your chatbot is key to these personalized experiences. When a user mentions something they like or dislike, or provides personal information, the chatbot can identify this information and update the user model accordingly. This continuous learning loop, powered by NER, is what allows the chatbot to evolve from a simple question-and-answer service to a personalized conversational agent.

## Evaluation and Analysis:

### Strengths:

- **Personalization:** The ability to learn user preferences and incorporate them into conversations using Likes and dislikes from the conversation.
- **Flexibility:** Utilizes a range of NLP techniques for robust understanding and response generation.
- **Scalability:** Designed to easily incorporate additional data sources and refine NLP capabilities.

### Weaknesses:

- **Context Understanding:** Limited ability to maintain context over long conversations.
- **Entity Recognition Limitations:** May not always accurately capture or utilize personal information.
- **Dependence on Quality of Knowledge Base:** The relevance of responses heavily relies on the pre-existing knowledge base content. And this knowledge base covers very limited amount of resources.

If It gets chance to expand the knowledge base then the performance and accuracy can be improved. Also, Integrating ML techniques can leverage the performance of the ChatBot.

### User Feedback (Survey Results of 10 people on average basis):

1. How easily can you interact with chatbot ?

**Ease of Interaction:** 4 / 5

2. How relevant information about Topic can you get from chatbot ?

**Relevance of Responses:** 4.5 / 5

3. How effective the personalisation feature of this chatbot ?

**Personalization Effectiveness:** 3.5 / 5

User Feedback : "I find the Chatbot idea really useful. We often search on Google to find relevant information and have to spend enormous time going through several links. The chatbot eases the hunt by providing the most relevant information on the specified key input. Additionally, the GUI option of the Chatbot gives a modern feel to the end user and allows us to stay interested in using the application more number of times. One feature I would like to see is the ability to save our chat conversations for later access, thus allowing us to avoid going through the same conversations again with the Chatbot. Overall, I would rate this product 4/5 where it excels in features, and ease of use, but could improve a bit by introducing conversation save feature."

Feedback indicates a positive reception to the chatbot's personalization and ease of use, though pointing out areas for improvement in understanding context and response relevance.

## Conclusion

This chatbot demonstrates the potential of NLP technologies in creating interactive, personalized user experiences. Future work will focus on improving context management, expanding the knowledge base, and refining entity recognition to enhance conversation quality and user satisfaction.