

# **HOTEL RECOMMENDATION TOOL USING WEBSCRAPING AND SENTIMENT ANALYSIS**

*Project report submitted  
in partial fulfilment of the requirement for the degree of*

**MTech Integrated Computer Science Engineering**

*By*

***SIDDARTH SR PILLAI - 21MIC0048***

***RAGUL G -21MIC0174***

***AKSHAJ ANIL -21MIC0186***

Under

**Web Mining and Social Network Analysis**

**CSI3033**

**J Component**



# **VIT<sup>®</sup>**

---

**Vellore Institute of Technology**

(Deemed to be University under section 3 of UGC Act, 1956)

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING  
VELLORE INSTITUTE OF TECHNOLOGY, VELLORE**

# TABLE OF CONTENTS

<b>TABLE OF CONTENTS.....</b>	
<b>ABSTRACT.....</b>	
<b>CHAPTER 1:INTRODUCTION.....</b>	
1.1    INTRODUCTION.....	
1.2    MOTIVATION.....	
1.3    BRIEF OVERVIEW OF PROBLEM.....	
1.4    SCOPE OF THE PROJECT.....	
1.5    SIGNIFICANT CONTRIBUTION.....	
<b>CHAPTER 2:REVIEW OF LITERATURE.....</b>	
2.1 REVIEWS.....	
<b>CHAPTER 3:PROBLEM DEFINITION.....</b>	
3.1 PROBLEM STATEMENT.....	
3.2 PROBLEM DEFINITION.....	
3.3 OBJECTIVE.....	
<b>CHAPTER 4:METHODOLOGY.....</b>	
4.1 INTRODUCTION.....	
4.2 APPROACHES USED TO ADDRESS THE PROBLEM.....	
4.3 STEPS/PHASES INVOLVED.....	
4.4 ALGORITHM DESCRIPTION.....	
4.5 TECHNIQUES USED FOR ANALYSIS.....	
<b>CHAPTER 5:DESIGN AND IMPLEMENTATION.....</b>	
5.1 INTRODUCTION.....	
5.2 DESIGN OF THE SYSTEM.....	
5.3 IMPLEMENTATION.....	
5.4 DETAILED DESCRIPTION OF CODE AND ALGORITHM.....	
<b>CHAPTER 6:RESULTS AND DISCUSSIONS.....</b>	
6.1 DESCRIPTION OF RESULT.....	
6.2 INTERPRETATION OF RESULT.....	

**CONCLUSION.....**

**REFERENCES.....**

**ANNEXURE.....**

# WEB SCRAPING AND SENTIMENT ANALYSIS ON HOTEL REVIEWS

## Abstract

This research paper aims to explore the landscape of hotel reviews using web scraping and sentiment analysis techniques. With the proliferation of online review platforms, understanding customer sentiment and preferences towards hotels is critical for both travelers and hospitality businesses. Using web scraping tools, we collect a comprehensive dataset of hotel reviews across multiple platforms. Through natural language processing and sentiment analysis, we assess the polarity and subjectivity of these reviews to determine the overall sentiment towards each hotel. In addition, we use advanced analytics to identify key features and attributes that influence customer satisfaction. Our findings not only contribute to the understanding of customer preferences in the hospitality sector, but also provide valuable insights for hotel management and marketing strategies. This research sheds light on top performing hotels and offers recommendations to improve customer experience and satisfaction in the hospitality industry.

## Chapter1:Introduction

### 1.1 Introduction

The hospitality industry in Tamil Nadu, India is a vibrant and diverse industry that caters to the needs of both domestic and international travelers. With its rich cultural heritage, picturesque landscapes and bustling urban centers, Tamil Nadu attracts a significant influx of tourists every year. Central to the traveler experience is the accommodation they choose, often influenced by reviews and ratings from other guests on various online platforms. In recent years, the proliferation of online review platforms has revolutionized the way travelers make decisions and the way hotels manage their reputations. Understanding the sentiments and preferences expressed in these reviews is paramount both for travelers looking for the best accommodation options and for hoteliers looking to improve their services. In this research paper, we embark on a journey to explore the landscape of hotel reviews in Tamil Nadu, using web scraping techniques and sentiment analysis to discern patterns, preferences, and perceptions among guests. Through systematic analysis of hotel reviews, we aim to identify top performing hotels, uncover key attributes that contribute to customer satisfaction, and provide insights useful to both travelers and the hospitality industry in Tamil Nadu.

### 1.2 Motivation:

Embarking on the journey of collecting hotel reviews from the web is an adventure into the realms of insight and empowerment. Each review encapsulates story, sentiment and perspective, offering a treasure trove of invaluable data waiting to be discovered. Through this project, you will become a beacon of excellence, championing superior standards and championing unique guest experiences within the hospitality industry. Your effort empowers decision makers by providing real-time, data-driven information that supports informed decisions and improves the quality of service offered by hotels. As you delve deeper into the world of hotel reviews, you will become a catalyst for innovation and drive transformational change by leveraging technology to understand and respond to guest feedback in new ways.

Ultimately, your project isn't just about collecting data—it's about increasing customer satisfaction, fostering a culture of continuous improvement, and shaping the future of hospitality by review.

### 1.3 Brief overview of the problem

From the customer's perspective, the challenges of obtaining hotel reviews underscore the need for authenticity, balance, and availability of information. Customers encounter countless reviews online, but the authenticity of these reviews is often questioned due to widespread fake or biased feedback. As customers move across platforms, they are looking for a comprehensive and diverse range of reviews to effectively inform their decision-making process. Timeliness is another critical factor, as customers require up-to-date information that reflects the current state of the hotel's services and facilities. In addition, user experience is paramount, with customers appreciating concise, informative and easily accessible reviews. Consistency across platforms is key to reducing confusion and instilling confidence in the review collection process. Customers prioritize data privacy and security and expect their personal data to be handled carefully and in accordance with regulations. Addressing these customer-facing challenges in sourcing hotel reviews is critical to promoting transparency, reliability and trust in online review platforms, ultimately enabling customers to make informed decisions about their lodging experience.

### 1.4 Scope of the project

**Selection of review platforms:** Determine which review platforms will be included in the scraping process. This may include popular platforms such as TripAdvisor, Booking.com, Google Reviews and others based on their relevance and popularity among travelers.

**Data Extraction:** Develop algorithms and scripts to extract relevant information from hotel review sites. This includes details such as review text, ratings, reviewer information (if available), review date and any other metadata deemed relevant to the analysis.

**Scalability:** Ensure the scraping process is scalable to handle a high volume of reviews across different hotels and platforms. Implement strategies to optimize performance and minimize resource consumption considering factors such as server load, bandwidth, and processing time.

**Data cleaning and preprocessing:** Clean and preprocess scraped data to ensure consistency, accuracy, and relevance. This may include removing duplicates, handling missing values, standardizing text formats, and filtering out spam or irrelevant content.

**Sentiment Analysis:** Use sentiment analysis techniques to categorize reviews as positive, negative, or neutral based on the sentiment expressed in the text. This analysis provides insight into overall customer satisfaction and sentiment trends over time.

**Data Visualization:** Visualize extracted data and analysis results using tables, graphs, and interactive dashboards to aid interpretation and decision making. Visualizations can include sentiment trends, review breakdowns, top-rated hotels, and benchmarking across properties.

**Documentation and Reporting:** Document the scraping methodology, data sources, preprocessing steps, analysis techniques, and findings in a comprehensive report. Effectively communicate project findings and implications to stakeholders, including hotel managers, marketing teams and hospitality industry decision makers.

### 1.5 Significant contribution

- 1) **insight into customer preferences:** By analyzing a large number of hotel reviews, the project offers valuable insights into customer preferences, priorities and issues. Understanding what guests appreciate and dislike about their hotel experiences allows hoteliers to tailor their services and amenities to better meet customer expectations, thereby improving overall guest satisfaction.
- 2) **Competitive Benchmarking:** The project allows hotels to benchmark themselves against the competition by comparing their own review scores, sentiment analysis results and key feature mentions to other hotels in the same region or category. This competitive intelligence helps hotels identify areas where they excel or lag, allowing them to improve their offering and remain competitive in the market.
- 3) **Data-driven decision-making:** By providing comprehensive data on guest feedback and sentiment, the project enables hotel managers and decision-makers to make informed decisions regarding resource allocation, service improvement and marketing strategies. Instead of relying solely on intuition or anecdotal evidence, hotels can use data-driven insights to prioritize initiatives that have the greatest impact on guest satisfaction and loyalty.
- 4) **Better customer engagement:** Armed with the learnings from the project, hotels can proactively engage with guests to address concerns, acknowledge positive feedback, and foster a sense of community and loyalty. By demonstrating responsiveness and attention to guest feedback, hotels can strengthen their relationships with customers and build a reputation for exceptional service and hospitality.
- 5) **A culture of continuous improvement:** The project promotes a culture of continuous improvement in the hospitality industry by encouraging hotels to continuously monitor and analyze guest feedback. By viewing guest reviews as a valuable source of actionable feedback rather than just passive comments, hotels can identify trends, patterns and opportunities for innovation that drive continuous improvement in the guest experience.
- 6) **Promoting transparency and trust:** By making hotel reviews more accessible and transparent to consumers, the project promotes trust and transparency in the hospitality industry. Guests can make more informed decisions about where to stay based on real, unbiased reviews, leading to more positive experiences and stronger relationships between hotels and their guests.

## CHAPTER 2: REVIEW OF LITERATURE

### 2.1 Reviews

#### *[1] Extracting hotel reviews from a review aggregation Website*

This work aims to develop a direct indicator for monitoring the satisfaction of hotel guests, which will complement the existing accommodation statistics provided by Statistics Finland. The research uses publicly available online reviews, especially from platforms such as TripAdvisor and Booking.com, and tries to collect data on customer satisfaction in a way that is consistent with regions and the frequency with which official statistics are updated. While previous studies have explored the use of online reviews to measure satisfaction and applied sentiment analysis techniques, they often offer snapshots of guest satisfaction and lack a clear procedure for systematically eliciting reviews.

The primary goal of this work is to create a simple indicator for monitoring hotel guest satisfaction, derived from a review dataset that can be updated monthly. To achieve this, the research seeks to automate the review download process with a focus on obtaining numerical review scores from websites hosting hotel customer reviews. Developed to download reviews, the program is initially focused on Finland, but is designed to be adaptable to other countries as well.

### ***[2] What do hotel guests really want? An analysis of online reviews using text mining***

This research delves into the field of hotel guest satisfaction through text mining and focuses on 21 five-star hotels in North Macedonia listed on Booking.com. As the hotel industry increasingly relies on advanced data analytics to improve guest satisfaction, text mining is emerging as a powerful tool for extracting insights from the vast pool of online reviews. As guest expectations rise in the digital age, personalized experiences are becoming paramount, forcing hotels to analyze online reviews in real time for better service delivery. The paper is divided into different parts: it starts with a theoretical background exploring hotel attributes, online guest reviews and text mining techniques. Subsequently, the research framework details the application of text mining methodologies, including inter-document matrix, LDA, and sentiment analysis, to the collected online reviews. The paper culminates with a comprehensive summary of the research findings, highlighting the benefits, limitations and recommendations for the future integration of text mining in hotel operations.

### ***[3] Text mining for aspect based sentiment analysis on customer review: A case study in the hotel industry***

This research delves into the pivotal role of Online Travel Agency (OTA) sites as Electronic Word Of Mouth (E-WOM) platforms, especially in the context of guesthouses in Malang. With a significant portion of bookings coming from OTAs, the study highlights the importance of leveraging online reviews for business continuity and increased customer satisfaction. The study acknowledges the limitations of current review processing methods and advocates sentiment analysis as a means of extracting valuable insights from review text. Based on established methodologies and previous research, the study proposes a sentiment analysis framework aimed at categorizing reviews into key aspects and distinguishing customer sentiment. Using machine learning classification methods, the study aims to provide stakeholders with useful insights to improve service quality and effectively solve customer problems. The paper is organized into structured sections, including Introduction, Literature Review, Methodology, Experiment, Analysis and Conclusion, which offer a comprehensive survey of the applications of sentiment analysis in the hospitality industry.

### ***[4] Sentiment classification of hotel service review on Traveloka sites using naïve bayes classifier (NBC) and binary logistic regression***

This study delves into the importance of online reviews in the hospitality industry, particularly focusing on the impact of Traveloka, an online travel services platform, on hotel marketing and performance evaluation. By analyzing visitor reviews from Gunawangsa MERR Hotel and Favehotel Rungkut, both of which use Traveloka for marketing purposes, the research aims to evaluate customer satisfaction and identify areas for improvement. Using text mining techniques, namely the Naïve Bayes Classifier (NBC) and binary logistic regression, the study classifies reviews into positive and negative sentiments to assess the overall guest experience. Through web scraping, the research collects data from reviews and uses lexicon dictionaries for sentiment analysis. A comparison of the classification performances between the NBC and binary logistic regression methods, visualized using Word Cloud, provides insight into the effectiveness of each approach. By identifying key aspects for improvement, this research seeks



to provide practical recommendations for Gunawangsa MERR Hotel and Favehotel Rungkut to optimize customer satisfaction and service delivery. Through this effort, the study contributes to the advancement of methodology for evaluating and improving hotel performance in the digital era.

#### ***[5] Application of social media analytics: a case of analyzing online hotel reviews***

This paper delves into the growing importance of social media analytics in the hotel industry, focusing on the analysis of online reviews from English-speaking customers in major Chinese cities. As social media continues to have a significant influence on consumer behavior and decision-making, businesses are increasingly using computational methods such as text mining and sentiment analysis to extract actionable insights from the vast amount of user-generated data. The study, involving 58 three- to five-star hotels selected through TripAdvisor, aims to identify the preferred hotel attributes and main concerns of foreign tourists visiting China. Through a comprehensive case study, the paper illustrates the use of natural language preprocessing, text mining, and sentiment analysis techniques to analyze online textual content, revealing valuable insights for hotel managers. The study demonstrates various visualization techniques to facilitate decision making and strategic planning in the hospitality sector and shed light on the correlation between review ratings and sentiment scores. Additionally, the analysis reveals common themes between satisfied and dissatisfied customers and highlights the importance of key categories such as food, location, rooms, service and staff. The paper concludes with implications and insights from the case study that underline the potential of social media analytics for informed decision-making and customer satisfaction in the hotel industry.

#### ***[6] Improving Naïve Bayes in Sentiment Analysis For Hotel Industry in Indonesia***

This study deals with the growing need to effectively process and analyze online hotel reviews, particularly focusing on the traveloka.com platform in Indonesia. With the proliferation of textual information online, users are looking for efficient ways to sift through vast amounts of testimonials to make informed decisions. Recognizing the limitations of traditional approaches, the study delves into sentiment analysis and topic modeling techniques to discern underlying sentiments and themes in hotel testimonials. By breaking down reviews into satisfaction measurement categories such as cleanliness, comfort, food, location and service, the research aims to provide users with valuable insights to help them make decisions. The study uses generative techniques and expertise to create a corpus for effective sentiment analysis and ranking of hotel reviews. Through the application of classification methods, the research seeks to streamline the process of evaluating hotel reviews, offering users a more effective and insightful means of evaluating hotel services and making informed booking decisions. This study contributes to advanced methodologies for sentiment analysis and topic modeling in the context of online hotel reviews, thereby improving user experience and supporting informed decision making in the tourism industry.

#### ***[7] Comparison of Feature Extraction Methods on Sentiment Analysis in Hotel Reviews***

This study investigates the use of sentiment analysis, particularly using the Support Vector Machine (SVM) algorithm, to analyze hotel reviews obtained from social media platforms. Given the overwhelming amount of reviews available, sentiment analysis offers a method to systematically categorize opinions as positive, negative, or neutral, providing valuable insights into customer sentiment. Before applying the SVM algorithm, three feature extraction methods are compared: Bag Of Words, TF-IDF and the improved TF-IDF approach. The selection of these methods takes into account the impact of repeated word features within each review. Through comparative analysis, TF-IDF is shown to be the most effective feature extraction



method, providing a remarkable accuracy rate of 71.75%, precision of 78.66%, recall of 71.91%, and F1 score of 70.08%. These results underscore the importance of verbal features in the analysis of hotel review data and highlight the effectiveness of sentiment analysis techniques in extracting meaningful insights from a large number of social media reviews.

#### ***[8] Using Online Hotel Customer Reviews to Improve the Booking Process***

This research delves into the evolving e-commerce landscape in the hotel industry, where online booking services have become key business models and guest experiences. With an increasing number of hotel companies integrating online booking services, along with the ability for customers to leave comments on affiliate sites, there is the potential to use customer reviews to improve search engines. The study focuses on connecting customer reviews with online hotel reservation search tools, particularly categorizing customers based on their travel intentions. Through opinion gathering, customer reviews are obtained to identify frequently mentioned features of the hotel. Subsequently, the goal of the analysis of these functions is to enrich the reservation process by integrating new characteristics in accordance with customer preferences. The research seeks to improve the online hotel booking experience by developing a tailored tool that uses customer reviews from platforms such as Agoda and aligns them with user preferences derived from surveys.

#### ***[9] The Value of Web Data Scraping: An Application to TripAdvisor***

This study delves into the emerging field of smart tourism consulting and focuses on the use of Web Data Discovery (WDS) and Website Discovery tools to analyze TripAdvisor's social pages throughout 2022, a period marked by post-pandemic conditions and signs of economic growth recovery. As tourism is significantly affected by government and mobility restrictions, understanding new trends and insights from social media platforms becomes paramount for tourism management and forecasting. Using Fanpage Karma (FpK) as a web scraper, the research aims to identify relevant metrics, unanswered trends and levels of data scraping, offering valuable lessons for tourism companies and users alike. The paper begins with an exploration of the potential of WDS and an initial assessment of the web scraper's capabilities, followed by an analysis of TA's social pages to reveal key trends, issues and management insights. The comparison of WDS tools further underlines the importance of choosing the right solution with implications discussed for tourism stakeholders in the Algarve and beyond. Concluding remarks highlight further trends, implications and avenues for future research in the dynamic environment of smart tourism consultancy.

#### ***[10] A comparison of hotel ratings between verified and non-verified online review platforms.***

This study delves into the field of electronic word of mouth (eWOM) in the tourism industry, focusing on the role of online reviews in shaping consumer decisions and the reliability of review platforms. The proliferation of online review platforms such as TripAdvisor and Booking.com has raised concerns about the authenticity of reviews, with instances of fake or misleading reviews undermining consumer trust. The research systematically compares the rating structures and dynamics of hotels on different platforms, distinguishing between verified and unverified rating systems. The findings reveal that ratings on unverified platforms tend to be more inflated and volatile, suggesting a higher prevalence of biased or questionable reviews. The study contributes new knowledge by examining the dynamics of reviews at the individual hotel level across platforms and offers practical recommendations to increase the transparency and reliability of reviews. Organized into sections focusing on literature review, research design, empirical findings and discussion, the paper provides valuable contributions to understanding and addressing issues in online rating systems in the tourism industry.

## **CHAPTER 3:PROBLEM DEFINITION**

### **3.1 Problem statement**

Web hotel reviews aim to collect, analyze and extract actionable insights from various online platforms to enhance guest satisfaction and operational excellence in the hospitality industry.

### **3.2 Problem definition**

The problem involves the effective aggregation, analysis, and utilization of hotel reviews sourced from various online platforms within the hospitality industry. Specifically, the challenge lies in developing a robust web scraping methodology to systematically gather review data and implementing sentiment analysis techniques to extract meaningful insights from the collected data. Thus, the problem definition revolves around creating a scalable and accurate framework for web scraping hotel reviews to facilitate data-driven decision-making and continuous improvement within the hospitality sector.

### **3.3 Objectives**

- 1)The main objective of our system is to make the user more convenient to identify the reviews from multiple sources across the web, providing a more comprehensive view of the hotel's strengths and weaknesses as perceived by past guests.
  - 2) The project aims to develop a web application for hotel search, integrating sentiment analysis to gauge customer feedback.
  - 3) Using web mining techniques, it retrieves and analyzes data from Booking.com to provide users with comprehensive information about hotels in a specified location. The objective is to enhance user decision-making by offering insights derived from both quantitative ratings and qualitative sentiment scores of hotel reviews.
  - 4) We have done sentimental analysis algorithms such as VADER to analyze the sentiment of the reviews.
  - 5)The output will be name of the hotel,location,rating and sentiment score
- Along with it the graphical comparison of the sentimental scores will be displayed.

## **CHAPTER 4: Methodology**

### **4.1 Introduction**

The research methodology used in this project implements a Flask web application that allows users to search for hotels in a specific location. The methodology involves several steps: Upon receiving a user input for the location, the application sends a request to Booking.com to fetch hotel data using web scraping techniques via the BeautifulSoup library. It extracts relevant information such as hotel names, locations, ratings, and URLs. It then utilizes the Nominatim API to obtain latitude and longitude coordinates for each hotel location. Additionally, the application fetches hotel review data from the provided URLs and performs sentiment analysis using the VADER sentiment analyzer from the NLTK library to calculate the average sentiment score for each hotel. Finally, the hotels are sorted based on their

sentiment scores, and the results are displayed to the user. The application ensures robustness by handling potential errors, such as failed requests, gracefully. Overall, this methodology integrates web scraping, API usage, sentiment analysis, and Flask web development to provide users with a personalized and informative hotel search experience.

## **4.2 Approaches used to address the problem**

### **1) Web Scraping with BeautifulSoup:**

- BeautifulSoup: to scrape hotel data from Booking.com.
- It identifies and extracts relevant information from the HTML response, such as hotel names, locations, ratings, and URLs.

### **2) NLP and Sentiment Analysis:**

- VADER sentiment analysis a tool from the NLTK library to analyze hotel reviews.
- It calculates the sentiment scores of the reviews to gauge the overall sentiment towards each hotel.

### **3) API Integration:**

- The code integrates with the Nominatim API to obtain latitude and longitude coordinates for each hotel location.
- This allows for the visualization of hotel locations on a map.

### **4) Flask Web Development:**

- The project is structured as a Flask web application, utilizing routes and templates to handle user requests and render HTML pages dynamically.
- It employs Flask's `render_template` function to render search results and handle form submissions.

### **5) Data Processing and Sorting:**

- Here we process and organize hotel data, including sentiment scores and coordinates, to present the most relevant information to users.
- It sorts hotels based on sentiment scores to display the most positively reviewed options first.

## **4.3 Steps/phases involved**

- **User Input and Request Handling:** The application starts by waiting for user input, typically in the form of a location entered in a search field. When the user submits the form, the Flask route `/search` is triggered, initiating the search for hotels in the specified location.
- **Web Scraping Hotel Data:** Upon receiving the user's search query, the application sends an HTTP GET request to Booking.com using the `requests` library, with specific headers to mimic a browser. The response HTML is then parsed using BeautifulSoup to extract relevant hotel information, such as names, locations, ratings, and URLs. This phase involves identifying and scraping hotel data from the Booking.com search results page.
- **Coordinate Retrieval:** After extracting hotel locations, the application utilizes the Nominatim API to obtain latitude and longitude coordinates for each hotel's address.

This phase involves sending a request to the Nominatim API with the hotel location as a query parameter and parsing the JSON response to extract the coordinates.

- **Sentiment Analysis of Reviews:** The application fetches hotel review data by following the URLs extracted during web scraping. It then extracts review text using BeautifulSoup and performs sentiment analysis using the VADER sentiment analyzer from the NLTK library. The sentiment scores for each review are averaged to calculate an overall sentiment score for each hotel.
- **Data Processing and Sorting:** Once hotel data, coordinates, and sentiment scores are collected, the application organizes this information into a structured format suitable for presentation to the user. The hotels are sorted based on their sentiment scores in descending order to prioritize positively reviewed hotels.
- **Rendering Results to the User:** Finally, the sorted hotel data, along with the user's search location, are passed to a Flask template for rendering. The template dynamically generates an HTML page, displaying the search results to the user, typically in the form of a list or a map with markers representing hotel locations. If no hotels are found for the specified location, an appropriate message is displayed to the user.

#### **4.4 Algorithm Description:**

##### **Web Scraping Algorithm:**

- **Description:** The web scraping algorithm is implemented using BeautifulSoup, a Python library for parsing HTML and XML documents. It follows these steps:
- Send an HTTP GET request to Booking.com to retrieve the HTML content of the search results page for the specified location.
- Parse the HTML content using BeautifulSoup to extract relevant information about each hotel, such as name, location, rating, and URL.
- Iterate through the extracted data and store it in a structured format for further processing.

##### **Sentiment Analysis Algorithm:**

- **Description:** Sentiment analysis is performed on hotel reviews using the VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment analyzer from the NLTK library. The algorithm proceeds as follows:
- Retrieve hotel review data by following the URLs extracted during web scraping.
- Extract review text from the HTML content of each hotel's review page.
- Apply sentiment analysis using VADER to assign sentiment scores to each review, which quantify the positivity or negativity of the text.

- Calculate the average sentiment score for each hotel by aggregating the sentiment scores of its reviews.
- The sentiment scores are then used to rank hotels, with higher scores indicating more positively reviewed hotels.

#### **Geocoding Algorithm:**

- Description: Geocoding is the process of translating addresses or location names into geographic coordinates (latitude and longitude) that can be used to pinpoint the location on a map. The algorithm proceeds as follows:
- Construct a request URL for the geocoding service (in this case, the Nominatim API) with the address or location name as a query parameter.
- Send an HTTP GET request to the geocoding service with the constructed URL.
- Receive a response from the geocoding service, typically in JSON format, containing the geographic coordinates corresponding to the provided address or location name.
- Parse the JSON response to extract the latitude and longitude coordinates.
- Return the extracted coordinates to the calling function or use them for further processing, such as displaying the location on a map.

#### **4.5 Techniques used for analysis**

The code employs several techniques for analysis, primarily focusing on extracting insights from hotel reviews and presenting them to users. These techniques include web scraping, sentiment analysis, and data visualization. Web scraping is utilized to gather hotel data, including names, locations, ratings, and review URLs, from Booking.com. Sentiment analysis, powered by the VADER sentiment analyzer from the NLTK library, evaluates the positivity or negativity of hotel reviews to calculate average sentiment scores for each establishment. These scores are then used to rank hotels, providing users with a curated list of positively reviewed accommodations. Additionally, the application leverages geocoding techniques through the Nominatim API to obtain latitude and longitude coordinates for hotel locations, facilitating data visualization on a map. Through these techniques, users can make informed decisions based on the sentiment of reviews and the geographic proximity of hotels to their desired location.

### **Chapter 5: Design and implementation**

#### **5.1 Introduction**

The design approach used in the provided application focuses on building a Flask web application for hotel sentiment analysis and visualization. The application follows a client-server architecture, where the server-side code (Python) handles data retrieval, analysis, and processing, while the client-side code (HTML, CSS, JavaScript) is responsible for presenting the results to the user in a web browser. The Flask framework is utilized to define routes and handle user requests, such as searching for hotels. The application leverages web scraping techniques with BeautifulSoup to extract hotel data from Booking.com, including names, locations, ratings, and review URLs. Sentiment analysis of hotel reviews is performed using the VADER sentiment analyzer from the NLTK library to calculate average sentiment scores for each hotel. The application also integrates geocoding techniques through the Nominatim API to obtain latitude and longitude coordinates for hotel locations, enabling data visualization on a map. The user interface is designed with simplicity and usability in mind, featuring clean

and responsive HTML templates styled with CSS for a visually appealing presentation of search results and sentiment analysis findings. Additionally, interactive features like map markers and a bar chart enhance the user experience, providing intuitive ways to explore hotel data. Overall, the design approach prioritizes functionality, performance, and user engagement to deliver a comprehensive and informative hotel search and sentiment analysis tool.

## **5.2 Design of the system**

### **5.2.1 Design modules**

- Web Application Framework (Flask)
- HTML Templates
- Web Scraping (BeautifulSoup)
- HTTP Requests (requests)
- Sentiment Analysis (NLTK)
- Geocoding (Nominatim API)
- Data Processing and Sorting
- Data Visualization (Leaflet, Chart.js)

### **5.2.2 Core modules**

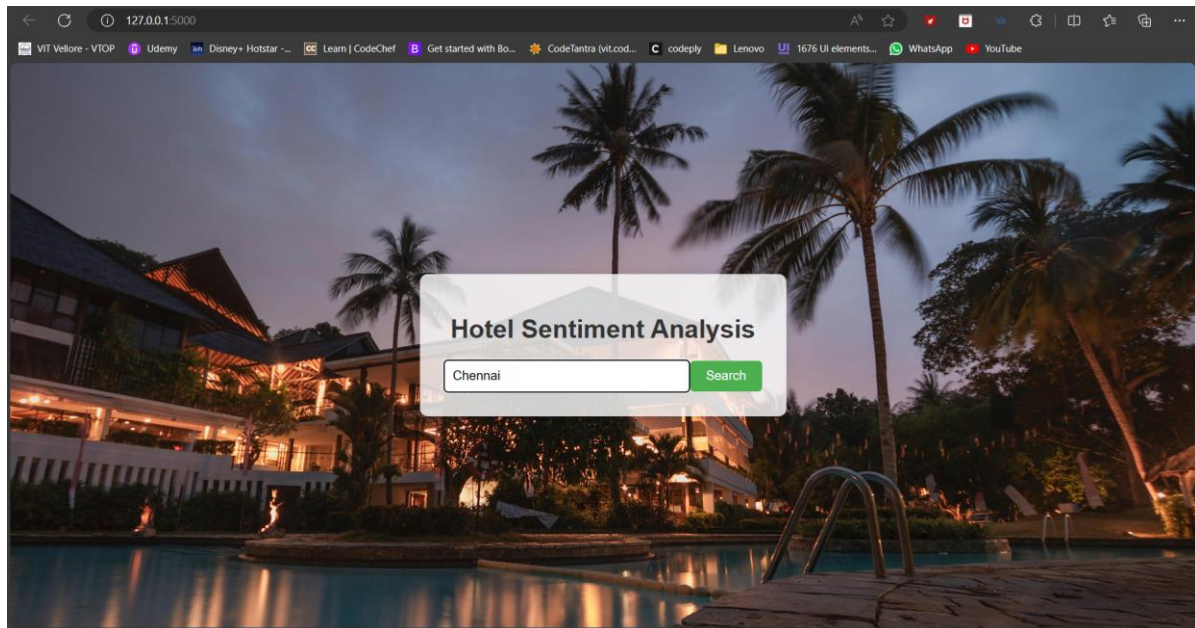
- Flask
- Requests
- Beautiful soap
- NTLK(Natural language tool kit)



## 5.3 Implementation

### Landing Page:

Enter the location



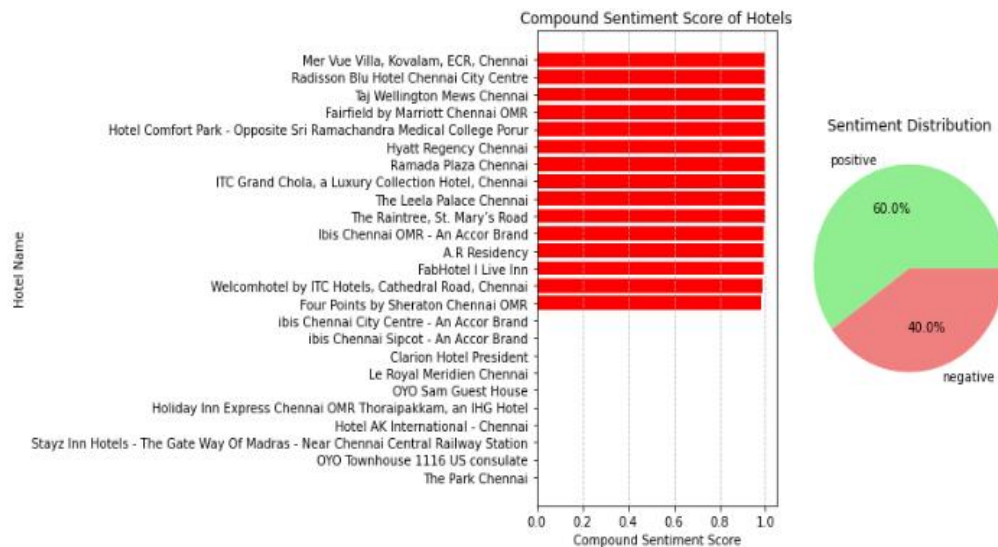
### Implementation in jupyter notebook:

```
name \
0 Hotel Comfort Park - Opposite Sri Ramachandra ...
1 FabHotel I Live Inn
2 Mer Vue Villa, Kovalam, ECR, Chennai
3 Ibis Chennai OMR - An Accor Brand
4 ibis Chennai Sipcot - An Accor Brand
5 Ramada Plaza Chennai
6 ibis Chennai City Centre - An Accor Brand
7 Radisson Blu Hotel Chennai City Centre
8 ITC Grand Chola, a Luxury Collection Hotel, Ch...
9 Taj Wellington Mews Chennai
10 A.R Residency
11 Hyatt Regency Chennai
12 The Park Chennai
13 The Leela Palace Chennai
14 Clarion Hotel President
15 Le Royal Meridien Chennai
16 Four Points by Sheraton Chennai OMR
17 OYO Sam Guest House
18 Welcomhotel by ITC Hotels, Cathedral Road, Che...
19 Holiday Inn Express Chennai OMR Thoraipakkam, ...
20 Hotel AK International - Chennai
21 Stayz Inn Hotels - The Gate Way Of Madras - Ne...
22 Fairfield by Marriott Chennai OMR
23 OYO Townhouse 1116 US consulate
24 The Raintree, St. Mary's Road
```



	location	rating \
0	Chennai	8.2
1	South Chennai, Chennai	7.8
2	Chennai	9.1
3	Sholinganallur, Chennai	7.7
4	SIPCOT IT Park, Chennai	7.4
5	South Chennai, Chennai	8.2
6	Central Chennai, Chennai	7.1
7	Egmore-Nungambakam, Chennai	8.3
8	Guindy, Chennai	8.4
9	South Chennai, Chennai	8.7
10	T - Nagar, Chennai	7.7
11	Central Chennai, Chennai	7.5
12	T - Nagar, Chennai	6.3
13	South Chennai, Chennai	8.7
14	Mylapore, Chennai	6.2
15	Guindy, Chennai	6.9
16	Old Mahabalipuram Road, Chennai	7.5
17	Triplicane, Chennai	7.7
18	Central Chennai, Chennai	8.5
19	Thoraipakkam, Chennai	6.3
20	Central Chennai, Chennai	6.6
21	Egmore-Nungambakam, Chennai	4.0
22	Old Mahabalipuram Road, Chennai	7.8
23	Central Chennai, Chennai	7.2
24	Central Chennai, Chennai	8.7

	name	compound_score \
2	Mer Vue Villa, Kovalam, ECR, Chennai	0.9990
7	Radisson Blu Hotel Chennai City Centre	0.9984
9	Taj Wellington Mews Chennai	0.9977
22	Fairfield by Marriott Chennai OMR	0.9976
0	Hotel Comfort Park - Opposite Sri Ramachandra ...	0.9971
11	Hyatt Regency Chennai	0.9970
5	Ramada Plaza Chennai	0.9967
8	ITC Grand Chola, a Luxury Collection Hotel, Ch...	0.9965
13	The Leela Palace Chennai	0.9962
24	The Raintree, St. Mary's Road	0.9955
3	Ibis Chennai OMR - An Accor Brand	0.9948
10	A.R Residency	0.9944
1	FabHotel I Live Inn	0.9939
18	Welcomhotel by ITC Hotels, Cathedral Road, Che...	0.9891
16	Four Points by Sheraton Chennai OMR	0.9835
6	ibis Chennai City Centre - An Accor Brand	0.0000
4	ibis Chennai Sipcot - An Accor Brand	0.0000
14	Clarion Hotel President	0.0000
15	Le Royal Meridien Chennai	0.0000
17	OYO Sam Guest House	0.0000
19	Holiday Inn Express Chennai OMR Thoraipakkam, ...	0.0000
20	Hotel AK International - Chennai	0.0000
21	Stayz Inn Hotels - The Gate Way Of Madras - Ne...	0.0000
23	OYO Townhouse 1116 US consulate	0.0000
12	The Park Chennai	0.0000



## 5.4 Detailed description of the code and algorithm

### Flask Setup:

The code imports the Flask framework and other necessary modules, including `render_template` for rendering HTML templates and `request` for handling HTTP requests. Additionally, it imports modules for web scraping (`BeautifulSoup`), making HTTP requests (`requests`), and sentiment analysis (`SentimentIntensityAnalyzer` from `NLTK`).

### Flask App Initialization:

An instance of the Flask class is created and assigned to the variable `app`. The app instance is initialized with the name `_name_`, which typically refers to the current module's name.

### Route Definitions:

Two routes are defined: `'/'` and `'/search'`.

The `'/'` route renders the `index.html` template when the root URL is accessed.

The `'/search'` route handles POST requests submitted from a form in the `index.html` template. It extracts the location entered by the user, performs web scraping on Booking.com to retrieve hotel data based on the location, calculates sentiment scores for each hotel using reviews, and renders the `results.html` template with the hotel data.

### Web Scraping and Sentiment Analysis:

The `search_hotels()` function is responsible for performing web scraping on Booking.com to retrieve hotel data based on the user-entered location. It extracts hotel names, locations, ratings,

and URLs. For each hotel, it calculates a sentiment score by extracting and analyzing reviews using the VADER sentiment analyzer.

### **Server Execution:**

Finally, the script checks if it's being executed as the main program (if `_name_ == '_main_'`) and starts the Flask development server with debugging enabled (`app.run(debug=True)`). This allows the Flask application to be run directly from this script, making it accessible through a local web server.

### **HTML Structure:**

The HTML structure defines the layout and content of the web page. It includes a header displaying the search location, a table to list hotel information, a map section to visualize hotel locations, and a bar chart to represent sentiment scores.

### **CSS Styling:**

CSS styles are applied to customize the appearance of the web page, including font styles, table formatting, map size, and chart container.

### **JavaScript Functionality:**


JavaScript code provides interactive features to the web page. It initializes a Leaflet map with markers representing hotel locations and binds pop-up information to each marker. Additionally, it generates a bar chart using Chart.js to visualize the sentiment scores of the hotels.

## Chapter 6: Results and discussions

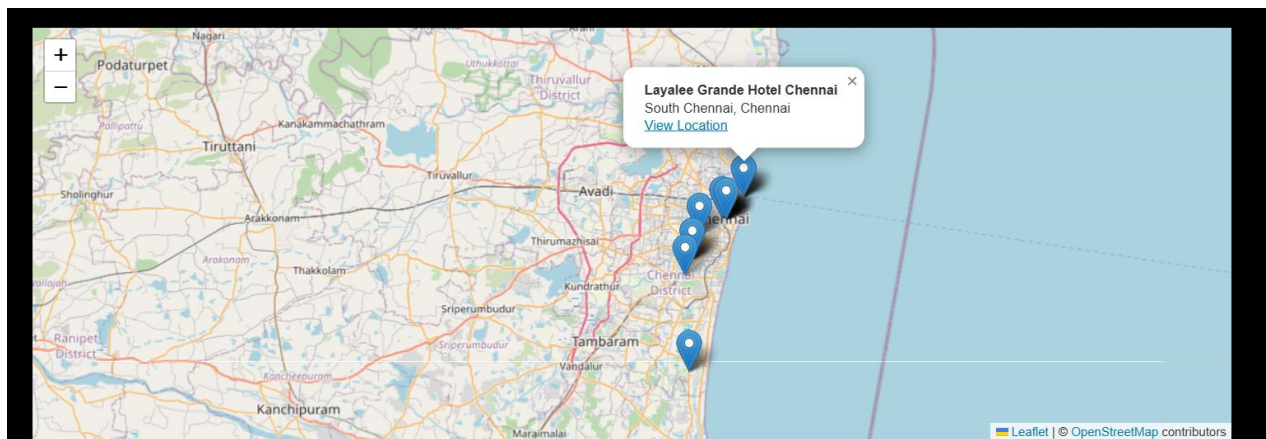
### 6.1 Description of result

#### Sentiment Scores of Hotel in Chennai:

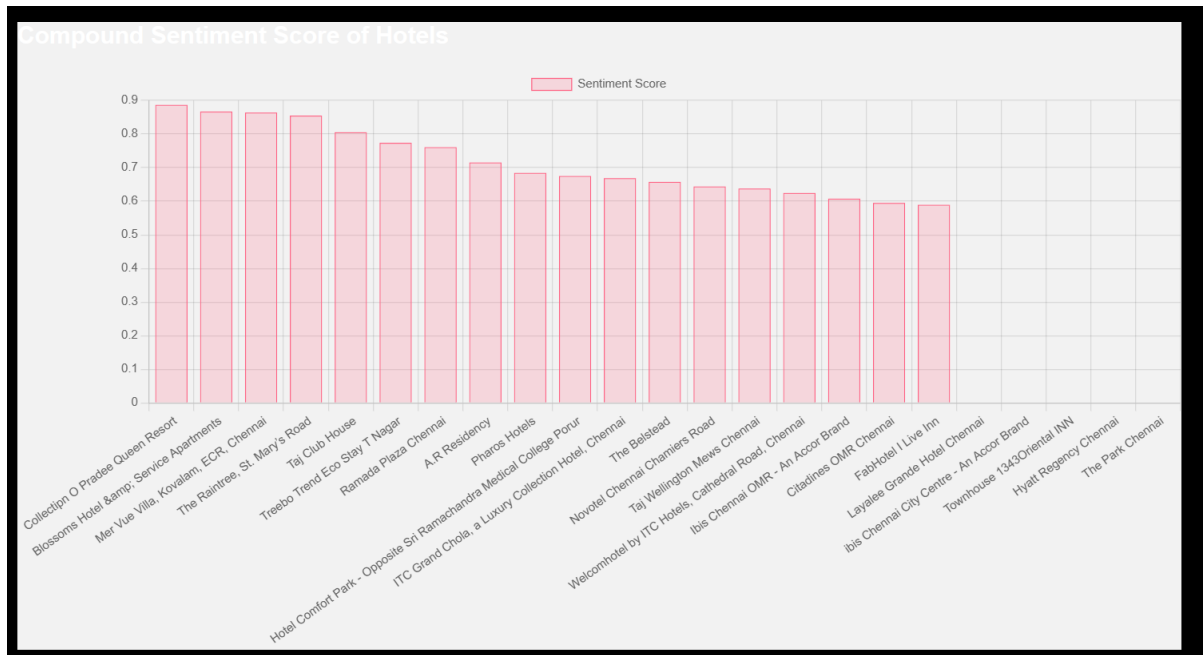
Hotels in T - Nagar, Chennai			
Name	Location	Rating	Sentiment Score
Collectipn O Pradee Queen Resort	Chennai	8.2	0.8869111111111111
Blossoms Hotel & Service Apartments	T - Nagar, Chennai	8.6	0.867
Mer Vue Villa, Kovalam, ECR, Chennai	Chennai	9.2	0.8638899999999999
The Raintree, St. Mary's Road	Central Chennai, Chennai	8.7	0.8547499999999999
Taj Club House	Central Chennai, Chennai	7.9	0.8057500000000001
Treebo Trend Eco Stay T Nagar	T - Nagar, Chennai	9.2	0.7739571428571429
Ramada Plaza Chennai	South Chennai, Chennai	8.1	0.76116
A.R Residency	T - Nagar, Chennai	7.7	0.71577
Pharos Hotels	Nungambakkam, Chennai	8.0	0.68463
Hotel Comfort Park - Opposite Sri Ramachandra Medical College Porur	Chennai	8.2	0.6753899999999999
ITC Grand Chola, a Luxury Collection Hotel, Chennai	Guindy, Chennai	8.5	0.6688400000000001
The Belstead	Nungambakkam, Chennai	7.9	0.65804
Novotel Chennai Chamiers Road	Central Chennai, Chennai	8.1	0.6443099999999999
Taj Wellington Mews Chennai	South Chennai, Chennai	8.7	0.63795
Welcomhotel by ITC Hotels, Cathedral Road, Chennai	Central Chennai, Chennai	8.7	0.62502
Ibis Chennai OMR - An Accor Brand	Sholinganallur, Chennai	7.6	0.6079600000000001
Citadines OMR Chennai	Sholinganallur, Chennai	8.4	0.59531
FabHotel I Live Inn	South Chennai, Chennai	7.8	0.58974
Layalee Grande Hotel Chennai	South Chennai, Chennai	7.0	0.0
ibis Chennai City Centre - An Accor Brand	Central Chennai, Chennai	7.1	0.0
Townhouse 1343Oriental INN	Central Chennai, Chennai	4.9	0.0
Hyatt Regency Chennai	Central Chennai, Chennai	7.4	0.0
The Park Chennai	T - Nagar, Chennai	6.3	0.0



#### Locations of hotel in map:



## Bar chart:



## 6.2 Interpretation of results:

This Flask application facilitates hotel searching by users, utilizing data from Booking.com. Upon receiving a location input, it queries Booking.com to retrieve hotel information such as names, locations, and ratings. Additionally, it employs sentiment analysis on customer reviews for each hotel using the VADER sentiment analyzer. The results are then presented on a web page, sorted by sentiment score in descending order, enabling users to prioritize hotels with more positive reviews. Accompanying the hotel listings are a map showing the geographical distribution of the hotels and a bar chart visualizing the sentiment scores, providing users with comprehensive insights to aid in decision-making. This approach not only offers users a convenient means of exploring available accommodations but also empowers them to make informed choices based on both quantitative ratings and qualitative sentiment analysis of customer feedback.

## CONCLUSION:

In conclusion, this Flask application effectively combines web scraping, sentiment analysis, and data visualization techniques to provide users with a robust platform for hotel searching and decision-making. By integrating data from Booking.com, sentiment analysis of customer reviews, and interactive visualizations, the application empowers users to make informed choices when selecting accommodations. The user-friendly interface, coupled with the ability to sort and visualize hotel data based on sentiment scores, enhances the overall user experience and enables users to prioritize hotels based on both quantitative ratings and qualitative sentiments. This application serves as a valuable tool for travelers seeking accommodations, offering them a comprehensive solution that integrates data-driven insights with user-friendly functionality.

## APPENDIX:

### Flask(frame work):

```
from flask import Flask, render_template, request
from bs4 import BeautifulSoup
import requests
from nltk.sentiment.vader import SentimentIntensityAnalyzer
import re

app = Flask(__name__)

# Initializing the VADER sentiment analyzer
sid = SentimentIntensityAnalyzer()

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/search', methods=['POST'])
def search_hotels():
    location = request.form['location']
    url =
f'https://www.booking.com/searchresults.html?ss={location}%2C+India&ssne=Ooty&
ssne_untouched=Ooty&efdco=1&label=gen173nr-
1FCAQoggJCDWNpdHlflTIxMTQ40DhIM1gEaGyIAQGYATG4ARfIAQzYAQHoAQH4AQ0IAgGoAg04At--
3K8GwAIB0gIkMGRhNWE1ZmItY2I2My00Y2MyLWI1YWItMjQ2ZGVhMzFhM2Rm2AIF4AIB&aid=30414
2&lang=en-
us&sb=1&src_elem=sb&src=searchresults&dest_type=city&group_adults=1&no_rooms=1
&group_children=0'

    headers = {
        'User-Agent': 'Mozilla/5.0 (X11; CrOS x86_64 8172.45.0)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.64 Safari/537.36',
        'Accept-Language': 'en-US, en;q=0.5'
    }

    # Sort hotels based on sentiment score
    hotels_data.sort(key=lambda x: x['sentiment_score'], reverse=True)

    if hotels_data:
        return render_template('results.html', hotels=hotels_data,
location=location)
    else:
        return "No hotels found for the given location"
    else:
        return "Failed to retrieve data from Booking.com"
```

```

def get_coordinates(location):
    # URL for the Nominatim API
    url =
f"https://nominatim.openstreetmap.org/search?format=json&q={location}"

    # Sending a GET request to the API
    response = requests.get(url)

    response = requests.get(url, headers=headers)
    if response.status_code == 200:
        soup = BeautifulSoup(response.text, 'html.parser')
        hotels = soup.findAll('div', {'data-testid': 'property-card'})

        hotels_data = []

        for hotel in hotels:
            name_element = hotel.find('div', {'data-testid': 'title'})
            name = name_element.text.strip() if name_element else None

            location_element = hotel.find('span', {'data-testid': 'address'})
            location = location_element.text.strip() if location_element else
None

            rating_element = hotel.find(
                'div', {'class': 'a3b8729ab1 d86cee9b25'})
            rating_text = rating_element.text.strip() if rating_element else
None

            # Extracting only the numerical part from the rating text using
            regular expressions
            if rating_text:
                rating_match = re.search(r'\d+\.\d+', rating_text)
                rating = float(rating_match.group()) if rating_match else None
            else:
                rating = None

            link_element = hotel.find('a', class_='a78ca197d0')
            hotel_url = link_element['href'] if link_element else None

            if location and hotel_url:
                lat, lon = get_coordinates(location)
                if lat is not None and lon is not None:
                    sentiment_score = get_sentiment_score(hotel_url)
                    hotels_data.append({
                        'name': name,
                        'location': location,

```



```

        'rating': rating,
        'sentiment_score': sentiment_score,
        'lat': lat,
        'lon': lon,
        'link': hotel_url
    })

# Checking if the request was successful
if response.status_code == 200:
    # Parsing the JSON response
    data = response.json()
    if data:
        # Extracting latitude and longitude from the response
        lat = float(data[0]['lat'])
        lon = float(data[0]['lon'])
        return lat, lon
    # Return None if coordinates cannot be obtained
    return None, None

def get_sentiment_score(url):
    response_hotel = requests.get(url)
    if response_hotel.status_code == 200:
        soup_hotel = BeautifulSoup(response_hotel.content, 'html.parser')
        review_elements = soup_hotel.findAll(
            'div', {'data-testid': 'featuredreview-text'})
        reviews = [review_element.text.strip()
                     for review_element in review_elements]

        # Perform sentiment analysis
        compound_scores = [sid.polarity_scores(
            review)['compound'] for review in reviews]
        if compound_scores:
            average_score = sum(compound_scores) / len(compound_scores)
        else:
            average_score = 0.0
        return average_score
    else:
        return 0.0

if __name__ == '__main__':
    app.run(debug=True)

```

index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Hotel Sentiment Analysis</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      background-image: url("/static/bg.jpg");
      background-size: cover;
      background-position: center;
      height: 100vh;
      display: flex;
      justify-content: center;
      align-items: center;
    }

    .container {
      background-color: rgba(255, 255, 255, 0.8);
      padding: 30px;
      border-radius: 10px;
      box-shadow: 0 0 20px rgba(0, 0, 0, 0.1);
      text-align: center;
      max-width: 400px;
      width: 90%;
    }

    h1 {
      color: #333;
      margin-bottom: 20px;
    }

    form {
      display: flex;
      justify-content: center;
      align-items: center;
      margin-top: 20px;
    }

    input[type="text"] {
      padding: 10px;
      font-size: 16px;
      border: 2px solid #ccc;
      border-radius: 5px;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Hotel Sentiment Analysis</h1>
    <form>
      <input type="text" value="Search for a hotel name" />
      <button type="submit">Search</button>
    </form>
  </div>
</body>
</html>
```

```

        flex: 1;
    }

    button[type="submit"] {
        padding: 10px 20px;
        font-size: 16px;
        background-color: #4CAF50;
        color: white;
        border: none;
        border-radius: 5px;
        cursor: pointer;
        transition: background-color 0.3s ease;
    }

    button[type="submit"]:hover {
        background-color: #45a049;
    }
</style>
</head>
<body>
    <div class="container">
        <h1>Hotel Sentiment Analysis</h1>
        <form action="/search" method="POST">
            <input type="text" id="location" name="location"
placeholder="Enter Location" required>
            <button type="submit">Search</button>
        </form>
    </div>
</body>
</html>

```

### Result.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Search Results</title>
    <link rel="stylesheet" href="https://unpkg.com/leaflet/dist/leaflet.css"
/>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #000000; /* Black background */

```

```

        color: #ffffff; /* White text */
    }

    h1 {
        text-align: center;
    }

    table {
        width: 80%;
        margin: 0 auto;
        border-collapse: collapse;
        color: #ffffff; /* White text */
    }

    table, th, td {
        border: 1px solid white; /* White border */
    }

    th {
        background-color: #f2f2f2;
        color: black; /* Black text */
    }

    #map {
        height: 400px;
        width: 80%;
        margin: 20px auto;
    }

    .chart-container {
        width: 80%;
        margin: 20px auto;
    }
</style>
</head>
<body>
<h1>Hotels in {{ location }}</h1>
<table>
    <thead>
    <tr>
        <th>Name</th>
        <th>Location</th>
        <th>Rating</th>
        <th>Sentiment Score</th>
    </tr>
    </thead>
    <tbody>
    {% for hotel in hotels %}

```

```

        <tr>
            <td>{{ hotel.name }}</td>
            <td>{{ hotel.location }}</td>
            <td>{{ hotel.rating }}</td>
            <td>{{ hotel.sentiment_score }}</td>
        </tr>
    {% endfor %}
</tbody>
</table>

<div id="map"></div>

<div class="chart-container" style="background-color: #f2f2f2;">
    <h2>Compound Sentiment Score of Hotels</h2>
    <canvas id="barChart"></canvas>
</div>

<script src="https://unpkg.com/leaflet/dist/leaflet.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/3.7.0/chart.min.js"></scr
ipt>
<script>
    var map = L.map('map').setView([{{ hotels[0].lat }}, {{ hotels[0].lon }}],
13);

    L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
        attribution: '&copy; <a
href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'
    }).addTo(map);

    {% for hotel in hotels %}
    L.marker([{{ hotel.lat }}, {{ hotel.lon }}]).addTo(map)
        .bindPopup('<b>{{ hotel.name }}</b><br>{{ hotel.location }}<br><a
href="{{ hotel.link }}" target="_blank">View Location</a>');
    {% endfor %}

    // Bar Chart
    var ctxBar = document.getElementById('barChart').getContext('2d');
    var barChart = new Chart(ctxBar, {
        type: 'bar',
        data: {
            labels: [{% for hotel in hotels %}"{{ hotel.name }}",{% endfor
%}],
            datasets: [{
                label: 'Sentiment Score',
                data: [{% for hotel in hotels %}{{ hotel.sentiment_score }},{%
endfor %}],
                backgroundColor: 'rgba(255, 99, 132, 0.2)',

```

```

        borderColor: 'rgba(255, 99, 132, 1)',
        borderWidth: 1
    }],
    },
    options: {
        scales: {
            y: {
                beginAtZero: true
            }
        }
    }
});
</script>
</body>
</html>

```

## References:

- [1] Thomas, D. M., & Mathur, S. (2019, June). Data analysis by web scraping using python. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 450-454). IEEE.
- [2] Hajba, G. L. (2018). Website Scraping with Python. *Berkeley: Apress*.
- [3] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, 5(4), 1093-1113.
- [4] Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined xpproach. *Journal of Informetrics*, 3(2), 143-157.
- [5] Mouthami, K., Devi, K. N., & Bhaskaran, V. M. (2013, February). Sentiment analysis and classification based on textual reviews. In *2013 international conference on Information communication and embedded systems (ICICES)* (pp. 271-276). IEEE.
- [6] Gräbner, D., Zanker, M., Fliedl, G., & Fuchs, M. (2012). Classification of customer reviews based on sentiment analysis. In *Information and communication technologies in tourism 2012* (pp. 460-470). Springer, Vienna.