SENTIMENT-BASED HOTEL REVIEW ANALYSIS & RECOMMENDATION SYSTEM

“Transforming Opinions into Options:
Your Ultimate Hotel Companion.”

Problem Statement :

The primary objective of this data science project is to utilize Natural Language Processing (NLP), Machine Learning techniques, and Large Language Models (LLMs) to perform sentiment analysis on user reviews. In addition, it aims to deliver customized hotel suggestions, taking into account a user's intended purpose of visit and specific requirements. This automated system for sentiment analysis and hotel recommendations will provide valuable insights to luxury hotels by allowing them to identify areas for improvement and areas of excellence within their services.

Acknowledgements :

- The data was scraped from Booking.com. All data in the file is publicly available to everyone already. Please be noted that data is originally owned by Booking.com
- This dataset contains 515,000 customer reviews and scoring of 1492 luxury hotels across Europe. Meanwhile, the geographical location of hotels are also provided for further analysis.

The csv file contains 17 fields and 515,000 records. The description of each field is as below :

Hotel_Address: Address of hotel.

Review_Date: Date when reviewer posted the corresponding review.

Average_Score: Average Score of the hotel, calculated based on the latest comment in the last year.

Hotel_Name: Name of Hotel

Reviewer_Nationality: Nationality of Reviewer

Negative_Review: Negative Review the reviewer gave to the hotel. If the reviewer does not give the negative review, then it should be: 'No Negative'

ReviewTotalNegativeWordCounts: Total number of words in the negative review.

Positive_Review: Positive Review the reviewer gave to the hotel. If the reviewer does not give the negative review, then it should be: 'No Positive'

ReviewTotalPositiveWordCounts: Total number of words in the positive review.

Reviewer_Score: Score the reviewer has given to the hotel, based on his/her experience

TotalNumberofReviewsReviewerHasGiven: Number of Reviews the reviewers has given in the past.

TotalNumberof_Reviews: Total number of valid reviews the hotel has.

Tags: Tags reviewer gave the hotel.

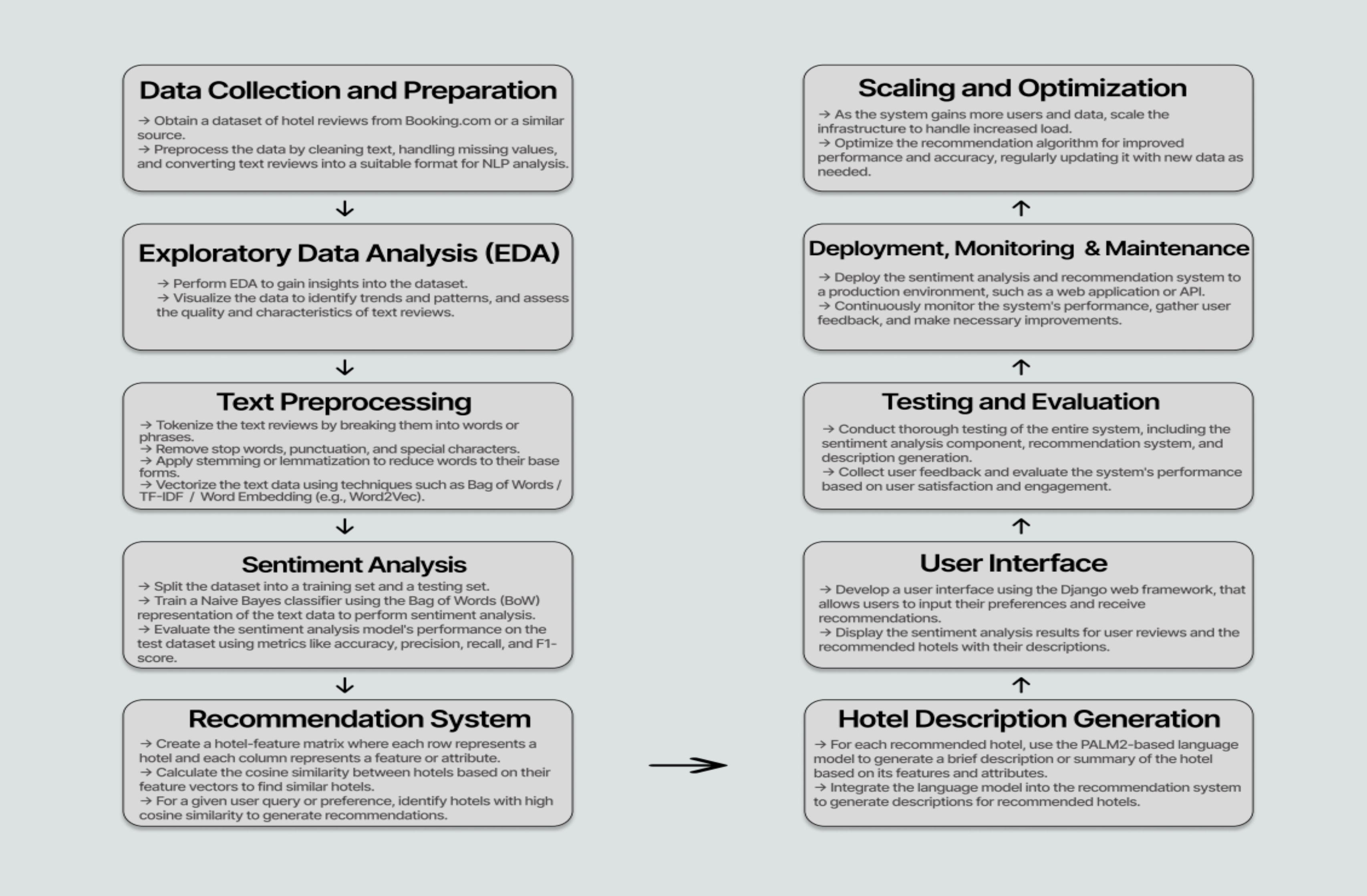
dayssincereview: Duration between the review date and scrape date.

AdditionalNumberof_Scoring: There are also some guests who just made a - scoring on the service rather than a review. This number indicates how many valid scores without review in there.

lat: Latitude of the hotel

lng: longtitude of the hotel

PROJECT WORKFLOW



Data Collection and Preparation

→ Obtain a dataset of hotel reviews from Booking.com or a similar source.
→ Preprocess the data by cleaning text, handling missing values, and converting text reviews into a suitable format for NLP analysis.

Exploratory Data Analysis (EDA)

→ Perform EDA to gain insights into the dataset.
→ Visualize the data to identify trends and patterns, and assess the quality and characteristics of text reviews.

Text Preprocessing

→ Tokenize the text reviews by breaking them into words or phrases.
→ Remove stop words, punctuation, and special characters.
→ Apply stemming or lemmatization to reduce words to their base forms.
→ Vectorize the text data using techniques such as Bag of Words / TF-IDF / Word Embedding (e.g., Word2Vec).

Sentiment Analysis

→ Split the dataset into a training set and a testing set.
→ Train a Naive Bayes classifier using the Bag of Words (BoW) representation of the text data to perform sentiment analysis.
→ Evaluate the sentiment analysis model's performance on the test dataset using metrics like accuracy, precision, recall, and F1-score.

Recommendation System

→ Create a hotel-feature matrix where each row represents a hotel and each column represents a feature or attribute.
→ Calculate the cosine similarity between hotels based on their feature vectors to find similar hotels.
→ For a given user query or preference, identify hotels with high cosine similarity to generate recommendations.

Hotel Description Generation

→ For each recommended hotel, use the PALM2-based language model to generate a brief description or summary of the hotel based on its features and attributes.
→ Integrate the language model into the recommendation system to generate descriptions for recommended hotels.

User Interface

→ Develop a user interface using the Django web framework, that allows users to input their preferences and receive recommendations.
→ Display the sentiment analysis results for user reviews and the recommended hotels with their descriptions.

Testing and Evaluation

→ Conduct thorough testing of the entire system, including the sentiment analysis component, recommendation system, and description generation.
→ Collect user feedback and evaluate the system's performance based on user satisfaction and engagement.

Deployment, Monitoring & Maintenance

→ Deploy the sentiment analysis and recommendation system to a production environment, such as a web application or API.
→ Continuously monitor the system's performance, gather user feedback, and make necessary improvements.

Scaling and Optimization

→ As the system gains more users and data, scale the infrastructure to handle increased load.
→ Optimize the recommendation algorithm for improved performance and accuracy, regularly updating it with new data as needed.

Tools & Methods used in the project :

- Pandas
- Numpy
- Matplotlib
- Seaborn
- WordCloud
- Sklearn
- NLTK
- Google.generativeai

Count Vectorizer

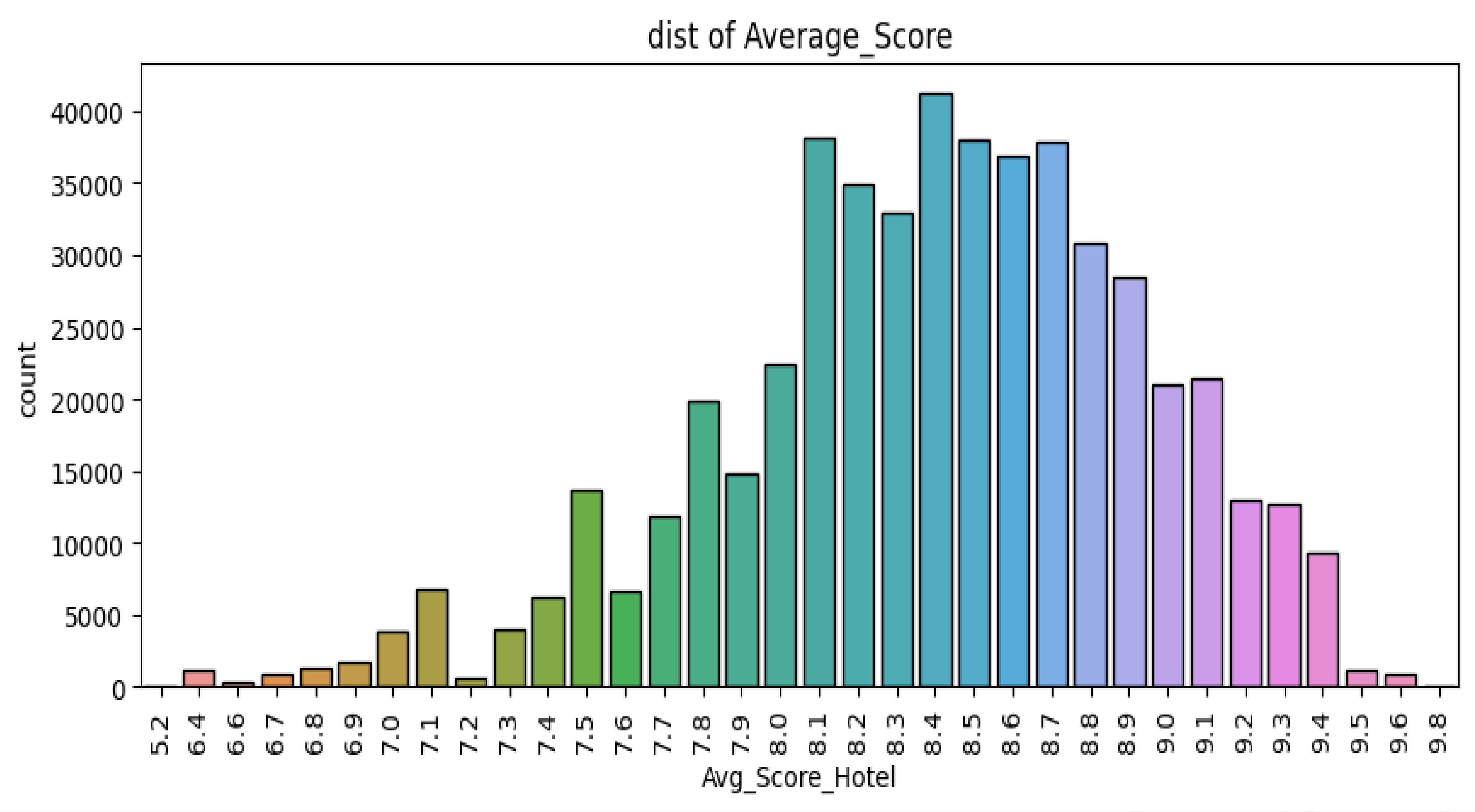
Bag-of-Words(BoW)

NaiveBayes Classifier

Cosine Similarity

Palm 2(LLM)

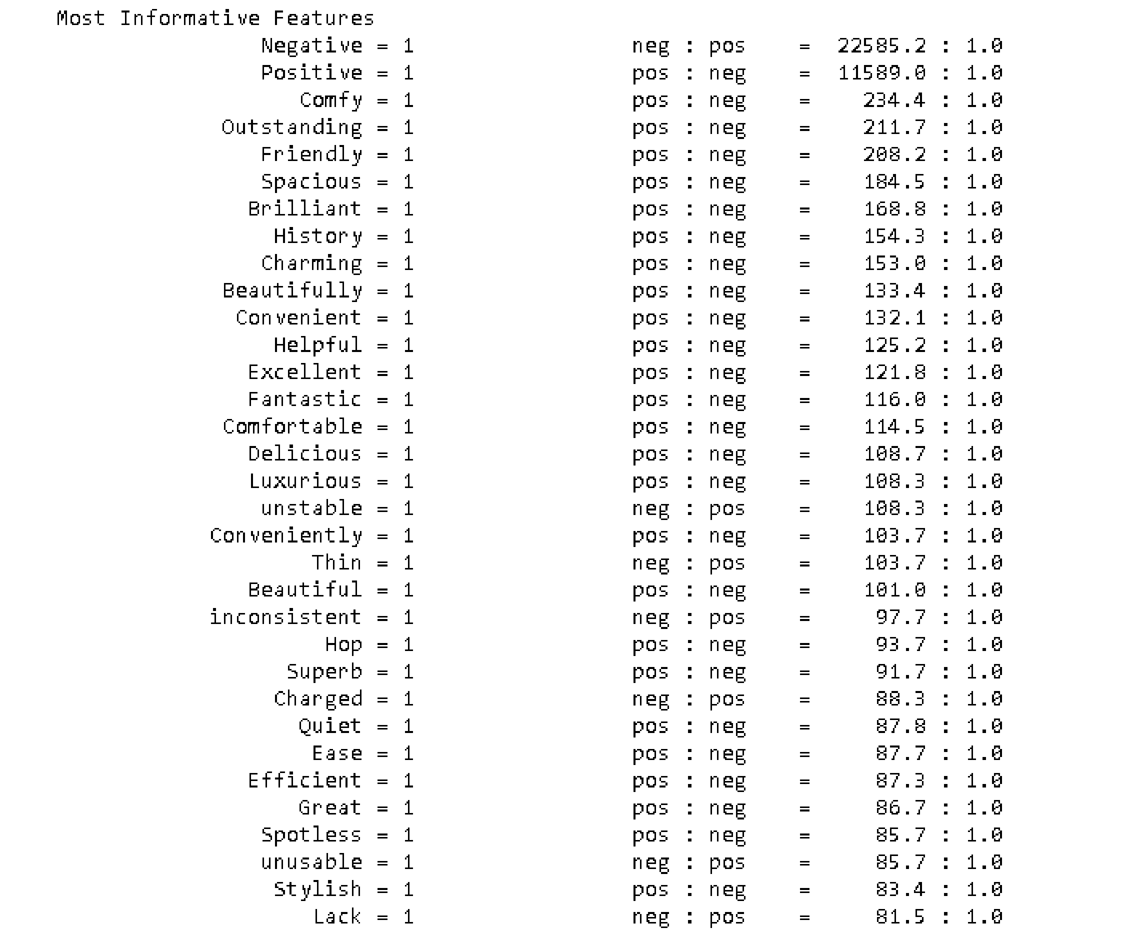
Few inferences from the project :



- After initially exploring into the dataset, I observed that the majority of the 1492 European luxury hotels have average scores that fall within the range of 7.6 to 9.2.

- The average scores span from 5.2 to 9.8, with a mean value of 8.4.

- I split the text into two groups, one with positive reviews and the other with negative reviews. Then, I used 80% of this data to teach the computer to tell them apart, and tested the computer's ability to do so on the remaining 20%.
- My research showed that we can use a technique called sentiment analysis in natural language processing (NLP) to predict whether reviews are positive or negative. Specifically, I used a bag-of-words model with a method called the Naive Bayes Classifier. This method was able to correctly identify positive and negative reviews with 93.5% accuracy during training and 92.5% accuracy during testing. These results are better than what a typical person could achieve (like around 80%), and it outperformed other machine learning algorithms in terms of accuracy. suggesting that the Naive Bayes Classifier is a good choice for this type of analysis.
- I found the most informative features, which are the words that best identify a positive or a negative review, or the words that had the greatest effect on the prediction accuracy.
- It is interesting to note that the most informative words for positive reviews tend to refer to the hotel staff (Friendly, Helpful, Efficient), room (Comfy, Spacious, Comfortable), and location (Convenient, Conveniently, Convenience), while the most informative words for negative reviews seem to refer mostly to problems with the facilities or amenities (unstable, Thin, Charged, Unusable, Lack, unreliable, damaged, Loud, Noisy, Smelly, Missing, loudly). This could be valuable insight for the reviewed hotels that are looking for areas to improve, in order to increase their ratings and attract more customers.



```
    Most Informative Features
                   Negative = 1              neg : pos    =  22585.2 : 1.0
                   Positive = 1              pos : neg    =  11589.0 : 1.0
                      Comfy = 1              pos : neg    =    234.4 : 1.0
                Outstanding = 1              pos : neg    =    211.7 : 1.0
                   Friendly = 1              pos : neg    =    208.2 : 1.0
                   Spacious = 1              pos : neg    =    184.5 : 1.0
                  Brilliant = 1              pos : neg    =    168.8 : 1.0
                    History = 1              pos : neg    =    154.3 : 1.0
                   Charming = 1              pos : neg    =    153.0 : 1.0
                Beautifully = 1              pos : neg    =    133.4 : 1.0
                 Convenient = 1              pos : neg    =    132.1 : 1.0
                    Helpful = 1              pos : neg    =    125.2 : 1.0
                  Excellent = 1              pos : neg    =    121.8 : 1.0
                  Fantastic = 1              pos : neg    =    116.0 : 1.0
                Comfortable = 1              pos : neg    =    114.5 : 1.0
                  Delicious = 1              pos : neg    =    108.7 : 1.0
                  Luxurious = 1              pos : neg    =    108.3 : 1.0
                   unstable = 1              neg : pos    =    108.3 : 1.0
               Conveniently = 1              pos : neg    =    103.7 : 1.0
                       Thin = 1              neg : pos    =    103.7 : 1.0
                  Beautiful = 1              pos : neg    =    101.0 : 1.0
               inconsistent = 1              neg : pos    =     97.7 : 1.0
                        Hop = 1              pos : neg    =     93.7 : 1.0
                     Superb = 1              pos : neg    =     91.7 : 1.0
                    Charged = 1              neg : pos    =     88.3 : 1.0
                      Quiet = 1              pos : neg    =     87.8 : 1.0
                       Ease = 1              pos : neg    =     87.7 : 1.0
                  Efficient = 1              pos : neg    =     87.3 : 1.0
                      Great = 1              pos : neg    =     86.7 : 1.0
                   Spotless = 1              pos : neg    =     85.7 : 1.0
                   unusable = 1              neg : pos    =     85.7 : 1.0
                    Stylish = 1              pos : neg    =     83.4 : 1.0
                       Lack = 1              neg : pos    =     81.5 : 1.0
```

- ➢ The most informative words indicating whether a review is positive or negative were identified for this dataset. Positive reviews predominantly emphasize the hotel staff and location, while highly negative reviews tend to focus on the facilities.
- ➢I did not find a significant correlation between the experience level of travelers and their review scores, nor between the nationality of the reviewers and their scores. It is possible that these relationships may become apparent in a larger dataset or for different types of hotels.

Observations [from positive reviews]

- These are the observations that would help hotels to improve their quality and service.
- These represent the aspects or services that customers found to their liking.

- Exceptional staff assistance and friendliness were highly appreciated by guests.
- Well-maintained and clean rooms garnered positive feedback.
- Guests found the bed's comfort to be exceptional.
- The hotel's favorable locations were well-received by visitors.
- The proximity to the city center was a notable advantage.
- Many guests were satisfied with the quality of the breakfast service.

Observations [from negative reviews]

- These are the observations that would help hotels to improve their quality and service.
- These are the services or aspects that did not meet the satisfaction of the customers.

- The provision and preparation of coffee and tea in-room hold significance.
- It appears that a considerable number of rooms are compact, including both the bed and bathroom. A more spacious room is desirable.
- There appear to be issues with the air conditioning, either due to its malfunction or excessive noise.
- The distance between the hotel and the city center is a concern.
- Problems with the availability of ironing boards have been noted.
- Concerns have been raised regarding breakfast pricing.
- Room availability tends to be limited due to high demand.
- Many customers express a preference for rooms on higher floors, rather than the ground floor.

- I have created a recommendation system that enables users to input their preferred country, purpose of visit, and travel preferences. This system then offers a list of the top 5 hotels in that country according to the user's specifications.
- To achieve this, I leverage cosine similarity to determine the most relevant recommendations, and I utilize the Palm2 language model to craft succinct descriptions for the recommended hotels. Additionally, I have employed the Django framework for the graphical user interface (GUI) development of this system.

References :

- "*Introduction to Machine Learning with Python*" by Andreas C. Müller and Sarah Guido
- "*Python for Data: Data Wrangling with Pandas, NumPy, and IPython*" by Wes McKinney
- Python packages : Matplotlib, Pandas, SciPy, Seaborn documentations
- StackOverflow for debugging code
- Popular blogs such as Analytics Vidhya, Towards Data Science, KDnuggets etc
- Youtube channels : StatQuest with Josh Starmer, Data School, Krish Naik etc