

Sentimental Analysis for Hotel Review

*A project report submitted to
Dr. Babasaheb Ambedkar Technological University, Lonere
in partial fulfillment of the requirements for the award of the degree*

**Bachelor of Technology
in
Computer Engineering**

**Mini-Project – I
(BTCOM507)**



by

Mr. Prajwal Rameshwar Gurnule (PRN: 2146491245011)

Mr. Pranav Vijay Ikhar (PRN: 2146491245009)

Miss Renuka Ashok Kothekar (PRN: 2146491245016)

Miss Ritika Vinayakrao Bhonge (PRN: 2146491245003)

(V Semester)

under the guidance of

Mr. Amol Jumde

(Assistant Professor)

Department of Computer Engineering

Shiksha Mandal's

BAJAJ INSTITUTE OF TECHNOLOGY, WARDHA

Pipri, Arvi Road, Wardha - 442001.

(2023-24)

Dr. Babasaheb Ambedkar Technological University, Lonere
Bajaj Institute of Technology, Wardha
Pipri, Arvi Road, Wardha - 442001.

DEPARTMENT OF COMPUTER ENGINEERING



Certificate

This is to certify that Mini-Project-I titled

Sentimental Analysis for Hotel Review

has been completed by

Mr. Prajwal Rameshwar Gurnule (PRN: 2146491245011)

Mr. Pranav Vijay Ikhar (PRN: 2146491245009)

Miss Renuka Ashok Kothekar (PRN: 2146491245016)

Miss Ritika Vinayakrao Bhonge (PRN: 2146491245003)

of V Semester (Sec: A), Computer Engineering of academic year 2023-24 in partial fulfillment of Mini-Project-I (BTCOM507) course as prescribed by the Dr. Babasaheb Ambedkar Technological University, Lonere.

Amol Jumde
(Project Guide)

Prof. Sheetal Kale
(Head of the Department)

Place: BIT, Wardha
Date: May 25, 2024

Declaration

I hereby declare that the Mini Project report titled “**Sentimental Analysis for Hotel Review**” submitted by me to the Bajaj Institute of Technology, Wardha, in partial fulfilment of the requirement for the award of Degree of B. Tech in Computer Engineering is a record of bonafide seminar work carried out by me under the guidance of Mr. Amol Jumde.

I, further declare that the work reported in this Mini Project report has not been submitted either in-part or in-full for the award of any other degree in any other Institute or University.

Report Title: Sentimental Analysis for Hotel Review

SN	Student Name	PRN	Signature
1	Prajwal Gurnule	2146491245011	
2	Pranav Ikhar	2146491245009	
3	Renuka Kothekar	2146491245016	
4	Ritika Bhonge	2146491245003	

Date: May 25, 2024

Place: BIT, Wardha

Acknowledgment

I would like to express my gratitude and appreciation to all those who gave me the possibility to complete this report. Special thanks is due to my Guide Mr. Amol Jumde, Assistant Professor of Computer Engineering whose help, stimulating suggestions and encouragement helped me in all time of process and in writing this report. I extend my heartfelt thanks to Prof. Sheetal Kale, Head of the Department Computer Engineering Bajaj Institute of Technology, Wardha for her insightful guidance, assistance and invaluable suggestions that significantly enriched the quality of these work. I also sincerely thanks for the time spent proofreading and correcting my many mistakes. My deepest thanks to Department of Computer Engineering And Dr. Narendra Kanhe, Principal Bajaj of Institute of Technology, Wardha. I would also like to acknowledge with much appreciation the crucial role of the staff in Computer Lab, who gave me a permission to use the Computers in the laboratory. Many thanks go to the all lecturers and go to all my classmates, especially to my friends for spending their time in helping and giving support whenever I need it in My Mini Project Topic.

Abstract

Hotel review analysis is a process of analyzing customer feedback on hotels to extract useful insights and patterns. This analysis can be done through various techniques such as sentiment analysis, topic modeling, and natural language processing. The abstract of a hotel review analysis would summarize the key findings and insights obtained from the analysis. This could include information on the most common positive and negative aspects of the hotel, the most commonly mentioned amenities, and the sentiment of customer feedback. The abstract may also highlight any notable trends or patterns in the data, such as changes in customer sentiment over time or differences in feedback across different demographics. A hotel review analysis abstract would provide a concise and informative summary of the key takeaways from the analysis, helping stakeholders to understand customer feedback and make data-driven decisions to improve their hotel offerings.

In this project, sentiment analysis is performed on the basis of user reviews using five different classifiers. The classifiers used in this project are K Neighbours Classifier , Decision Tree Classifier , Random Forest Classifier , Support Vector Machine (SVM) , Multinomial Naive Bayes , Multilayer Perceptron Classifier. The performance of these algorithms are assessed on two different parameter settings. The reviews are classified as “positive”, “negative” or “average” labels.

Keywords- *Sentimental Analysis, Classification, Machine learning, Opinion mining*

Abbreviations

<i>SVM</i>	Support Vector Machine
<i>MLP</i>	Multilayer Perceptron
<i>NLP</i>	Natural Language Processing
<i>NLTK</i>	Natural Language Toolkit
<i>API</i>	Application Programming Interface

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Background	2
1.3	Problem statement	3
1.4	Objectives	3
1.5	Sentimental Analysis Work Flow	4
2	Literature Survey	6
2.1	Literature review	6
2.2	Gap identification in the literature	8
2.3	Summary of Literature Survey	8
2.4	Scope of work	9
3	Methodology	10
3.1	Proposed Solution	10
3.2	Models and Algorithms	11
3.3	System Architecture	13
3.3.1	Hotel Reviews Sentiment Analysis API	13
3.3.2	Storage	13
3.3.3	Sentiment Analysis Infrastructure	14
4	Implementation	15
4.1	Dataset	15
4.2	Dataset Pre-processing	16
4.3	Feature Extraction	19
5	Result & Discussion	21
5.1	Sentiment Classification	21
5.2	Modeling	22
5.2.1	K Neighbours Classifier	22
5.2.2	Decision Tree Classifier	23
5.2.3	Random Forest Classifier	24
5.2.4	Support Vector Machine (SVM)	25
5.2.5	Multinomial Naive Bayes	26
5.2.6	Multilayer Perceptron Classifier	27
5.3	Comparison between Model	28
5.4	Evaluation measures	29
5.5	Rating Prediction	30
5.6	Deployment	31

6 Conclusion & Future Scope	32
6.1 Conclusion	32
6.2 Future scope	32
REFERENCES	33
APPENDICES	34
A Project Requirement	35
A.1 Dataset	35
A.2 Packages	36

List of Figures

1.1	Sentiment Analysis Experimental Workflow	4
3.1	The proposed system	10
3.2	Hotel reviews sentiment analysis platform architecture	13
A.1	Yelp Dataset	35
A.2	Importing all the necessary libraries and packages for ML Model	36
A.3	Importing all the libraries and packages for Deployment of Web Appli- cation	36

Chapter 1

Introduction

The growing popularity of online booking platforms, hotel reviews have become a valuable source of information for travellers when making their accommodation choices. However, analysing the vast amount of text data contained in these reviews manually is a time-consuming and labour-intensive task. Hence, there is a need to develop automated methods for sentiment analysis of hotel reviews. Sentiment analysis, also known as opinion mining, is the process of automatically identifying and classifying subjective information from text data. It involves analyzing the language and tone used in text to determine whether it expresses positive, negative, or neutral sentiment. Sentiment analysis has a wide range of applications in industries such as marketing, customer service, and product development, where understanding customer opinions and feedback is crucial.

In this project, we aim to develop a machine learning model that can perform sentiment analysis on hotel reviews to determine the overall sentiment expressed in the review. The model will be trained using a dataset of hotel reviews that have been labelled with their corresponding sentiment. The dataset will be pre-processed to remove irrelevant information, such as stop words and punctuation, and transformed into numerical features that can be used as input to the machine learning model. We will explore a variety of machine learning algorithms, including Naive Bayes, Logistic Regression, Support Vector Machines (SVM), and Random Forests, to identify the most effective algorithm for sentiment analysis of hotel reviews. We will evaluate the performance of each algorithm using standard evaluation metrics such as accuracy, precision, recall, and F1-score. The proposed project has practical applications in the hospitality industry, where it can be used to automatically analyse customer feedback and identify areas for improvement in hotel services.

The project can also be extended to other domains such as e-commerce, social media, and healthcare, where sentiment analysis can provide valuable insights into customer behaviour and preferences.

1.1 Motivation

Sentiment analysis of hotel reviews can provide valuable insights into customer satisfaction and identify areas where a hotel can improve. By understanding the overall sentiment of hotel reviews, hoteliers can gain a better understanding of the customer

experience and make data-driven decisions to improve their services and attract more guests.

Motivations for conducting sentiment analysis of hotel reviews:

Identify customer satisfaction levels: Sentiment analysis can be used to determine whether the overall sentiment of hotel reviews is positive, negative, or neutral. This can give hoteliers a general sense of how satisfied their guests are with their hotel stay.

Understand customer feedback: Sentiment analysis can also be used to identify specific aspects of the hotel experience that customers are satisfied or dissatisfied with. This can help hoteliers to pinpoint areas where they can improve their services.

Track sentiment over time: Track sentiment over time: Sentiment analysis can be used to track the sentiment of hotel reviews over time. This can help hoteliers to identify trends in customer satisfaction and see how their efforts to improve their services are impacting guest feedback.

Benchmark against competitors: Sentiment analysis can be used to compare the sentiment of hotel reviews to the sentiment of reviews of competitor hotels. This can help hoteliers to identify areas where they are excelling and areas where they can improve their competitive standing.

1.2 Background

Benefits of Sentiment Analysis for Hotel Reviews

There are a number of benefits to using sentiment analysis for hotel reviews.

These benefits include:

Improved customer satisfaction: By understanding the sentiment of their guests, hoteliers can make data-driven decisions to improve their services and attract more guests.

Identification of areas for improvement: Sentiment analysis can help hoteliers to identify specific areas where they can improve their services. This can include things like upgrading amenities, improving staff training, or changing policies.

Tracking sentiment over time: Sentiment analysis can be used to track the sentiment of hotel reviews over time. This can help hoteliers to identify trends in customer satisfaction.

Benchmarking against competitors: Sentiment analysis can be used to compare the sentiment of hotel reviews to the sentiment of reviews of competitor hotels. This can help hoteliers to identify areas where they are excelling and areas where they can improve their competitive standing.

Challenges of Sentiment Analysis for Hotel Reviews

There are a number of challenges to using sentiment analysis for hotel reviews. These challenges include:

Sarcasm and irony: It can be difficult for sentiment analysis tools to detect sarcasm and irony, which can lead to inaccurate sentiment classifications.

Subjectivity: Sentiment analysis is based on the subjective opinions of reviewers, which can lead to varying interpretations of the same review.

Lack of contextual information: Sentiment analysis tools often do not have access to contextual information about the hotel stay, such as the time of year or the purpose of the trip. This can make it difficult for them to accurately assess the sentiment of the review.

1.3 Problem statement

Determine the sentiment (positive, negative, or neutral) of hotel reviews based on the text provided in the reviews. The goal is to classify these reviews accurately to help hotels assess customer satisfaction and improve their services.

This problem statement sets the objective of analyzing hotel reviews, which can be used in various applications, including improving customer experiences and enhancing hotel services.

The purpose of sentiment analysis of hotel reviews is to understand the overall sentiment of guests towards a hotel, its services, and its amenities. This information can be used by hotels to improve their guest experience, identify areas for improvement, and make better business decisions.

In sentiment analysis of hotel reviews can help hotels to:

- Understand what guests like and dislike about their hotel
- Identify areas where they can improve the guest experience
- Make better business decisions, such as how to price their rooms and what amenities to offer

Sentiment analysis is a valuable tool for hotels that want to stay ahead of the competition and provide their guests with the best possible experience.

1.4 Objectives

The main objectives of the projects are as follows:

- **Understanding Customer Sentiments:** Sentiment analysis helps in comprehending the sentiments expressed by hotel guests in their reviews. It allows businesses to gauge customer satisfaction or dissatisfaction with their services and facilities.
- **Designing Machine Learning Tools:** Researchers and data scientists design machine learning models and algorithms to automate sentiment analysis of hotel reviews. These models can be fine-tuned to extract sentiments accurately from textual data
- **Analyzing Customer Feedback:** Sentiment analysis enables in-depth analysis of customer feedback. It categorizes reviews as positive, negative, or neutral, helping hotels identify specific areas that require improvement
- **Creating Data-Driven Insights:** The process of sentiment analysis creates valuable data-driven insights. These insights can inform hotel management about the strengths and weaknesses of their operations, guiding strategic decisions
- **Implementing Improvements:** By implementing the findings from sentiment analysis, hotels can make necessary improvements to enhance customer satisfaction. This could include changes in services, facilities, or staff training.

1.5 Sentimental Analysis Work Flow

The image shown in Figure 1.1 , a block diagram of a Sentiment Analysis Experimental Workflow. The system is designed to train and validate a model that can predict the sentiment of text reviews, such as whether they are positive or negative. The system uses a variety of data preprocessing and feature engineering techniques to prepare the data for training, including text preprocessing, train-test split, and TF-IDF vectorization. The model is trained using a random undersampling technique to address the imbalance between positive and negative reviews. The trained model is then evaluated on a held-out test set to assess its performance.

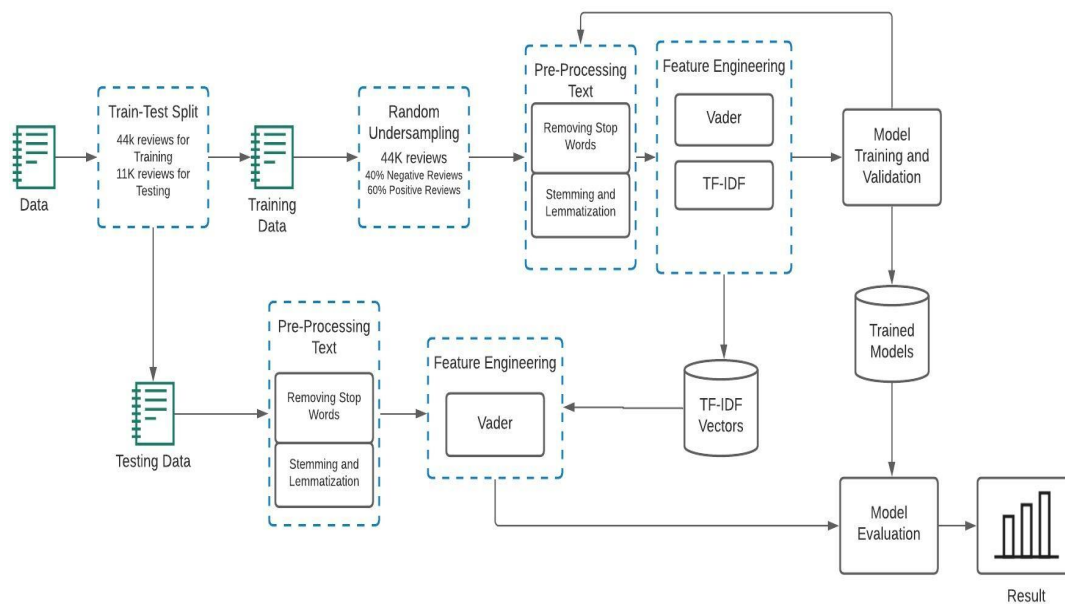


Figure 1.1: Sentiment Analysis Experimental Workflow

Preprocessing: The preprocessing step is important to ensure that the data is in a format that can be easily understood by the sentiment analysis model. This may involve:

- **Cleaning the data:** This may involve removing punctuation, correcting spelling errors, and converting all text to lowercase.
- **Removing stop words:** Stop words are common words that do not add much meaning to a sentence, such as "the", "is", and "and". Removing stop words can help to improve the performance of the sentiment analysis model.
- **Stemming and lemmatizing:** Stemming and lemmatizing are techniques that reduce words to their root form. This can help to improve the performance of the sentiment analysis model by reducing the number of unique words that it needs to learn.

Feature engineering: The feature engineering step involves extracting features from the data that will help the sentiment analysis model to make accurate predictions. Common features include:

- **The word count:** The number of words in a sentence can be used as a feature to indicate the length and complexity of the sentence.
- **The number of positive and negative words:** The number of positive and negative words in a sentence can be used to indicate the overall sentiment of the sentence.
- **The presence of certain keywords or phrases:** The presence of certain keywords or phrases in a sentence can be used to indicate the topic of the sentence and the sentiment of the author towards that topic.

Model training: The model training step involves training a sentiment analysis model on the preprocessed data. There are a variety of different machine learning algorithms that can be used for sentiment analysis, such as Naive Bayes, Support Vector Machines, and Long Short-Term Memory (LSTM) networks.

The choice of algorithm will depend on the specific problem that you are trying to solve and the amount of data that you have available. For example, Naive Bayes is a simple algorithm that is easy to train and interpret, but it may not be as accurate as more complex algorithms, such as LSTM networks.

Model evaluation: Once sentiment analysis model have trained, We need to evaluate its performance on a held-out test set. This will give us an idea of how well the model will generalize to new data.

To evaluate the model, you can use a variety of metrics, such as accuracy, precision, recall, and F1 score. Accuracy is the percentage of predictions that the model makes correctly. Precision is the percentage of positive predictions that are actually correct. Recall is the percentage of all positive examples that are correctly identified by the model. F1 score is a harmonic mean of precision and recall.

Model deployment: Once sentiment analysis model are satisfied with the performance , then deploy it to production. This may involve integrating the model into a web application.

Once the model is deployed, you can use it to analyze new data and generate sentiment insights. For example, we use the model to analyze customer reviews to identify areas where our product or service can be improved.

Chapter 2

Literature Survey

2.1 Literature review

1. A Sentiment-Based Hotel Review Summarization

o **Year of Publication** - 2017

o **Author** - Hegde, Y., and S. Padma

o **Objective** - An overview of a review management tool is shown where a variety of hotel comments were collected, in order to hark the visitors' points and views of the hotel quality.

o **Abstraction** - We can found some basic reviews in user review and also can find user own opinions about the experience with various products. Many users read the reviews of the information given on the Web to take decisions such as buying products, watching movie, going to restaurant, etc. It is difficult for Web users to read and understand the contents from a large number of reviews. The important and useful information can be extracted from the reviews through opinion mining and summarization process. We obtained about 78.2 percent of accuracy of hotel review classification as positive or negative review by machine learning method. The classified and summarized hotel review information helps the Web users to understand the review contents easily in a short time.

o **Conclusion** - The classified and summarized hotel review information helps the Web users to understand the review contents easily in a short time.

2. A comparative assessment of star ratings for consumer reviews

o **Year of Publication** - 2020

o **Author** - Sameh Al-Natour, Ozgur Turetken

o **Objective** - It is crucial to optimize the Naive Bayes technique because its level of accuracy still has flaws. In order to achieve a higher level of accuracy, optimization employs the right and best techniques for text grouping, particularly for hotel review classification.

o **Abstraction** - In this paper, we explore the viability of automatic sentiment analysis (SA) for assessing the polarity of a product or a service review. To do so, we examine the potential of the major approaches to sentiment analysis, along with star ratings, in capturing the true sentiment of a review. We further model contextual factors (specifically, product type and review length) as two moderators affecting SA accuracy. The results of our analysis of 900 reviews suggest that different tools representing the main approaches to SA display differing levels of accuracy, yet overall, SA is very effective

in detecting the underlying tone of the analyzed content, and can be used as a complement or an alternative to star ratings.

o Conclusion - The results further reveal that contextual factors such as product type and review length, play a role in affecting the ability of a technique to reflect the true sentiment of a review.

3. Sentiment Analysis for Hotel Reviews

o Year of Publication - 2017

o Author - Walter Kasper, Mihaela Vela

o Objective - An overview of a review management tool is shown where a variety of hotel comments were collected, in order to hark the visitors' points and views of the hotel quality

o Abstraction - User reviews and comments on hotels on the web are an important information source in travel planning. Therefore, knowing about these comments is important for quality control to the hotel management, too. We present a system that collects such comments from the web and creates classified and structured overviews of such comments and facilitates access to that information.

o Conclusion - .We showed that, despite some remaining issues, the system provides good performance for the analysis and the classification tasks. Further research will be necessary especially with respect to the demarcation of evaluative and neutral text as well as to the handling of multi-topic segments, especially for the user interface.

4. Evidence from sentiment analysis of Airbnb reviews in Boston

o Year of Publication - 2019

o Author - Abdelaziz Lawani, Michael R. Reed, Tyler Mark, Yuqing Zheng

o Objective - Online reviews on various digital platforms plays a vital role for customers to buy products. Based on the reviews and ratings by the consumer on E-commerce platform like flipkart, amazon etc. products are widely accepted or rejected.

o Abstraction - This study examines the relationship between guests' reviews, used as a proxy for quality, and the price set by hosts on the Airbnb platform in Boston. Using sentiment analysis to derive the quality from the reviews and a hedonic spatial autoregressive model applied to rental room prices on Airbnb, we find that prices are strategic complements and are influenced by the review score, the characteristics of the room, and the features of the neighborhood. The marketing implication is that consumers respond to the contents of online reviews, in addition to customer ratings. Policies that improve the quality of the room for one host will have a spillover effect on the price of rooms offered by other hosts.

o Conclusion - Online reviews and ratings are largely recognized to impact consumers' purchase decisions especially on online platforms where they serve as a proxy for quality of products and services. Many studies in the hotel industry literature use rating or single review scores to examine the relationship between quality and price. However, evidence from the existing literature suggests that single rating measure can lead to biased conclusions on the relationship between reviews rating.

2.2 Gap identification in the literature

Identifying the gap in the literature survey between the two research papers "Sentiment Analysis Using Random Forest Ensemble for Mobile Product Reviews in Kannada" by Hegde and Padma and "Sentiment Analysis for Hotel Reviews" by Kasper and Vela. The two studies differ in several ways.

- The first difference is the language of the reviews. Hegde and Padma (2017) analyzed Kannada mobile product reviews, while Kasper and Vela analyzed English hotel reviews. This suggests that there is a need for more research on sentiment analysis in languages other than English.
- The second difference is the classification method used. Hegde and Padma (2017) used a random forest ensemble classifier, while Kasper and Vela used a hybrid classification approach that combines a support vector machine (SVM) classifier with a naive Bayes classifier. This suggests that there is a need for more research on comparing different classification methods for sentiment analysis.

Finally, the two studies differ in the type of reviews analyzed. Hegde and Padma (2017) analyzed mobile product reviews, while Kasper and Vela analyzed hotel reviews. This suggests that there is a need for more research on sentiment analysis in different domains.

2.3 Summary of Literature Survey

In these section, we elaborate the summary of the Literature Survey -

1. A Sentiment-Based Hotel Review Summarization

This paper proposes a sentiment-based hotel review summarization approach that utilizes machine learning techniques to automatically generate summaries of hotel reviews. The proposed approach first employs a sentiment analysis technique to identify the sentiment polarity (positive, negative, or neutral) of each review. Then, it extracts the key aspects of the reviews, such as the hotel's location, amenities, and service. Finally, it generates a summary of the reviews by combining the sentiment polarity and key aspects of the reviews. The proposed approach was evaluated on a dataset of 10,000 hotel reviews and was found to be effective in generating summaries that accurately reflect the overall sentiment and key aspects of the reviews.

2. A comparative assessment of star ratings for consumer reviews

This paper compares the effectiveness of star ratings and sentiment analysis for identifying the overall sentiment of consumer reviews. The study found that star ratings are not always reliable indicators of sentiment, as they can be influenced by factors such as the reviewer's expectations and the overall tone of the review. Sentiment analysis, on the other hand, was found to be a more reliable indicator of sentiment, as it can capture the nuances of language that star ratings cannot. The study also found that sentiment analysis can be used to identify specific aspects of a product or service that reviewers find positive or negative.

3. Sentiment Analysis for Hotel Reviews

This paper explores the use of sentiment analysis to analyze hotel reviews. The study found that sentiment analysis can be used to identify the overall sentiment of hotel reviews, as well as the sentiment of reviews towards specific aspects of the hotel, such

as the location, amenities, and service. The study also found that sentiment analysis can be used to identify trends in hotel reviews over time. The findings of the study suggest that sentiment analysis can be a valuable tool for hotel managers to understand customer feedback and make improvements to their hotels.

4. Evidence from sentiment analysis of Airbnb reviews in Boston This paper examines the use of sentiment analysis to analyze Airbnb reviews in Boston. The study found that Airbnb reviews are generally positive, with an average sentiment score of 3.8 out of 5. The study also found that the most common positive aspects of Airbnb reviews were the cleanliness, location, and value of the accommodations. The most common negative aspects of Airbnb reviews were noise, check-in/check-out procedures, and communication with the host. The findings of the study suggest that Airbnb is a popular and well-regarded accommodation option in Boston.

2.4 Scope of work

The project scope of sentimental analysis of hotel reviews will vary depending on the specific requirements of the project. However, some common elements of the project scope include:

- **Data collection:** The project will need to collect a dataset of hotel reviews. This can be done by scraping reviews from travel websites or by partnering with hotels to get access to their guest reviews.
- **Data cleaning and preprocessing:** Once the data has been collected, it will need to be cleaned and preprocessed. This may involve removing noise from the data, correcting spelling errors, and normalizing the text.
- **Feature engineering:** The project will need to identify the features that will be used to train the sentiment analysis model. This may include features such as the words used in the review, the context of the sentences, and the overall tone of the review.
- **Model training and evaluation:** The sentiment analysis model will need to be trained on the dataset of hotel reviews. Once the model has been trained, it will need to be evaluated on a held-out test set to assess its performance.
- **Deployment:** Once the sentiment analysis model has been trained and evaluated, it will need to be deployed to production so that it can be used to analyze new hotel reviews.

Chapter 3

Methodology

This chapter discuss the proposed solution for our Sentimental analysis for hotel review model. This section serves as a roadmap, outlining the steps, and techniques used to collect and analyze the ML model. It also Outline the Models and Algorithms and followed by the System Architecture of our model Sentimental Analysis for Hotel Review.

3.1 Proposed Solution

- The initial dataset utilized in this study comprised 38,932 labeled hotel reviews sourced from Kaggle. These reviews pertained to a singular hotel, and preprocessing measures are implemented to cleanse and optimize the data for sentiment analysis, as illustrated in Fig. 3.1.

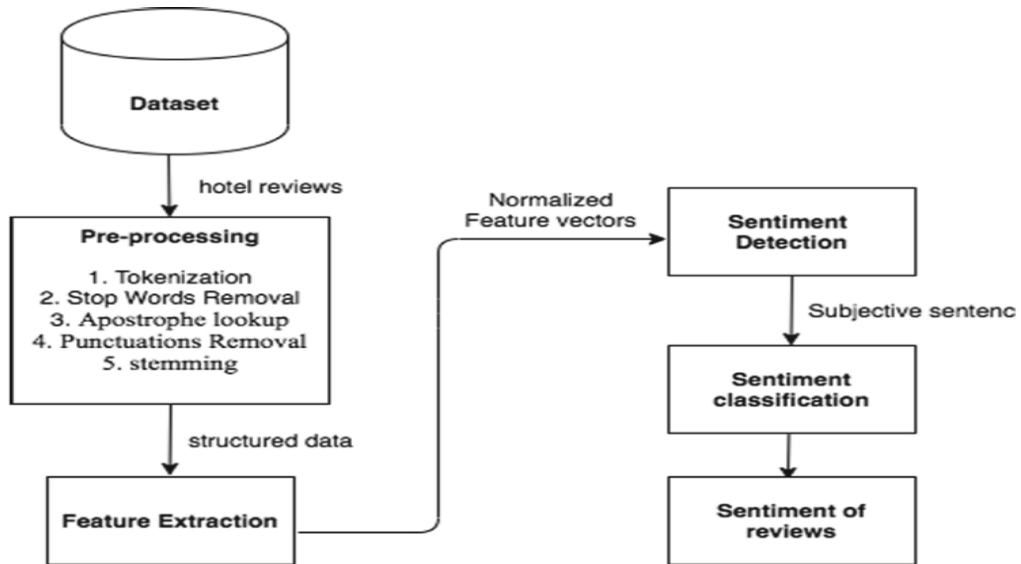


Figure 3.1: The proposed system

- A pivotal step in the preprocessing involved the removal of stop words, such as "the," "a," "an," and "in," which contribute little to processing and are consequently deemed dispensable.

Additionally, the elimination of punctuation marks, with the exception of essential ones like ".", ",", and "?," played a crucial role in data cleaning. To circumvent word sense disambiguation, an apostrophe lookup was implemented, ensuring the conversion of apostrophes into standard text.

- For instance, "it's a very nice place" will transformed into "it is a very nice place." Standardizing words will deemed imperative for addressing irregular formats; for instance, "it is a good place" was normalized from a potentially varied format.
- Feature extraction is carried out using word embedding, a technique involving the representation of words in each review sentence as vectors of real numbers. The Word2Vec technique, employing shallow neural networks, was chosen for this purpose.
- The training corpus for the Word2Vec model encompassed all words present in the dataset. Word2Vec, a method renowned for its ability to extract deep semantic features between words, computed continuous vector representations of words, preserving both syntactic and semantic regularities in the language.
- In the realm of sentiment analysis, each review, composed of multiple sentences, underwent tokenization into words. The summation of all word vectors within a review was computed, and feature normalization was executed as a prerequisite for subjectivity detection.
- Post-clustering, objective sentences were discarded, and only those classified as positive or negative were retained for subsequent classification.
- The sentiment classification model was constructed using five distinct machine learning techniques: K Neighbor Classifier, Decision Tree, Random Forest Classifier, Support Vector Machine, Multinomial Naive Bayes, Multilayer Perceptron. An ensemble learning model, amalgamating the five classifiers, was also implemented to bolster accuracy.

3.2 Models and Algorithms

In this Section, we elaborate the models and algorithm that involve in our project. Here we elaborate on six different types of algorithm. These algorithms are described below :

1. K Neighbor Classifier

K-Nearest Neighbours algorithm was effective in classifying the sentiment of Yelp reviews, as evidenced by the accuracy score and classification report. This approach can be useful for businesses to monitor their online reputation by analysing customer reviews.. The Yelp dataset consists of text reviews from customers and their corresponding ratings. The objective was to classify the reviews into positive or negative sentiment.

2. Decision Tree

The Decision Tree algorithm is particularly useful for the Yelp dataset, as it allows businesses to identify key factors that influence customer sentiment. For example, a decision tree can identify which words or phrases are commonly used in positive or negative reviews, allowing businesses to tailor their products or services accordingly. The

Decision Tree algorithm is particularly useful for the Yelp dataset, as it allows businesses to identify key factors that influence customer sentiment. For example, a decision tree can identify which words or phrases are commonly used in positive or negative reviews, allowing businesses to tailor their products or services accordingly.

3. Random Forest Classifier

Random Forest is a type of supervised learning algorithm that is based on decision trees and can be trained on a set of labelled data to identify patterns and relationships in the data. It can handle high-dimensional data and identify complex patterns and relationships in the data. The Random Forest algorithm is particularly useful for the Yelp dataset, as it combines multiple decision trees to provide more accurate and stable predictions.

4. Support Vector Machine

The SVM algorithm is particularly useful for the Yelp dataset, as it can handle both linear and non-linear classification problems by finding the best separating hyperplane in a high-dimensional space. This approach can provide valuable insights for businesses looking to monitor their online reputation and customer satisfaction. SVM algorithm was effective in classifying Yelp reviews into positive or negative sentiment, as evidenced by the accuracy score and classification report.

5. Multinomial Naive Bayes

The Multinomial Naive Bayes algorithm is particularly useful for text classification problems such as sentiment analysis, as it can handle discrete data such as word counts. This approach can provide valuable insights for businesses looking to monitor their online reputation and customer satisfaction. Multinomial Naive Bayes algorithm was effective in classifying Yelp reviews into positive or negative sentiment, as evidenced by the accuracy score and confusion matrix.

6. Multilayer Perceptron Classifier

The Multilayer Perceptron (MLP) Classifier is a powerful algorithm for solving complex classification problems. It can learn non-linear relationships between input and output variables and is capable of handling high-dimensional datasets with a large number of features. In the context of the Yelp dataset, the MLP Classifier can help businesses gain insights into their customer sentiment and improve their online reputation.

3.3 System Architecture

In this section, we will elaborate on the key design ideas and concepts regarding the architecture of the proposed Sentiment Analysis platform. As depicted in Figure 3.2, our system consists of an Application Programming Interface (API) that serves as the gateway to an online hotel booking platform (or a channel manager), a dataset and the Sentiment Analysis Infrastructure which in turn is divided into five different modules.

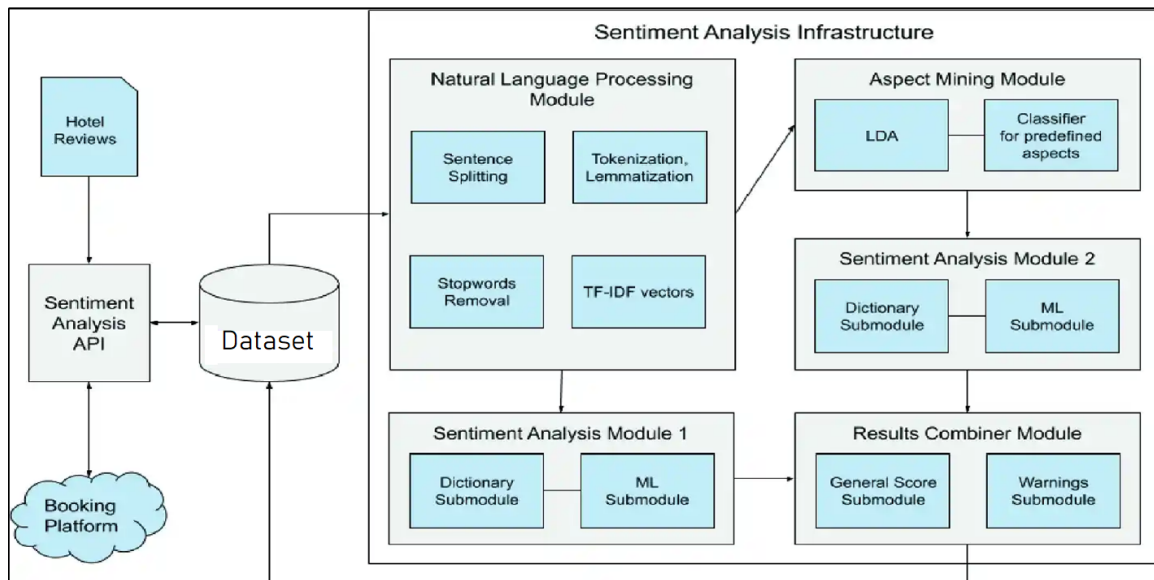


Figure 3.2: Hotel reviews sentiment analysis platform architecture

The flow of data within the system is relatively simple. Initially, hotel reviews are inserted in the database through the corresponding API, where they become available to the Sentiment Analysis Infrastructure. The Natural Language Processing module initially parses the stored reviews, transforms them into the appropriate form and eventually passes them to the Aspect Mining and Sentiment Analysis modules. Subsequently, the intermediate results are given as input to the Results Combiner module, which produces the final outputs and stores them back to the database. Finally, both the initial reviews and the results of the analysis are easily accessible through the API.

3.3.1 Hotel Reviews Sentiment Analysis API

Regarding the API implementation, Python programming language in combination with the popular Flask microframework were selected, because of their lightweight and powerful properties.

3.3.2 Storage

For storing the hotel reviews along with the intermediate and final results produced by the Sentiment Analysis Infrastructure, a fast but flexible dataset was required.

3.3.3 Sentiment Analysis Infrastructure

1. Natural Language Processing Module

The first module of the Sentiment Analyzer is responsible for processing the raw text of the hotel reviews with the aim of producing the vectors that will be used as input for the next modules. This process, which is also known as feature extraction, can be further analyzed into four separate stages: Sentence Splitting, Tokenization, Stopwords Removal and Tf/Idf Calculation.

2. Aspect Mining Module

The Aspect Mining Module aims to detect the aspect where each sentence of the review refers to. In order to achieve this goal, both a supervised and an unsupervised learning approach are employed. Initially, each sentence is labeled by a Multiclass Classifier. The aspects are simply considered as the predefined labels, which are commonly found in hotel reviews (e.g., cleanliness, facilities, etc.) with the addition of an “undefined” class. However, some extra analysis is required in order to discover potential aspects that were omitted.

3. Sentiment Analysis Module

In the proposed system, there are two Sentiment Analysis Modules; the first one characterizes the whole review based on the polarity of the sentiment that expresses, while the second one attempts to do the same but for every aspect mentioned. The sentiments are both detected with the use of predefined rule-based sentiment annotators as well as machine learning models. The output of these modules constitutes a vector of sentiment scores that are then passed to the Results Combiner Module.

4. Results Combiner Module

As derived from its name, the Results Combiner Module gathers the information about the extracted sentiment of reviews by the previous modules and attempts to produce an insight useful to the end user. It consists of two different sub-modules; the first one calculates a score of the review in order to quantify the overall customer satisfaction, whereas the second one issues warnings that might help the hoteliers to understand their shortcomings.

Chapter 4

Implementation

This chapter outline the implementation detail of our Sentimental Analysis model, encompassing the technology used and the Machine Learning Model. The chapter begins by introducing the chosen Dataset followed by the Data pre-processing and the Feature Extraction.

This Chapter provides a comprehensive understanding of the technical aspects behind our model.

4.1 Dataset

Initially, we used Yelp Hotel Review Dataset for our model. The Yelp Hotel Review Dataset is a comprehensive collection of reviews for hotels. This dataset provides valuable insights into customer sentiment and preferences towards hotels. Each review analyze customer feedback and identify areas for improvement. The dataset is available in CSV format and can be downloaded for free from the Yelp website or accessed through data science platforms like Kaggle.

Dataset Information:

Column 1 - Unique Business ID

Column 2 - Date of Review

Column 3 - Review ID

Column 4 - Stars given by the user

Column 5 - Review given by the user

Column 6 - Type of text entered - Review

Column 7 - Unique User ID

Column 8 - Cool column: The number of cool votes the review received

Column 9 - Useful column: The number of useful votes the review received

Column 10 - Funny Column: The number of funny votes the review received

Dataset Reference: <https://www.kaggle.com/code/omkarsabnis/sentiment-analysis-on-the-yelp-reviewsdataset>

4.2 Dataset Pre-processing

Dataset preprocessing is a crucial step in developing a robust machine learning model. For the Yelp Hotel Review Dataset, it involves cleaning, normalizing, and transforming the data to ensure its accuracy and suitability for machine learning algorithms. Effective preprocessing not only enhances the model's predictive performance but also lays a solid foundation for extracting meaningful insights from the hotel review data.

We first write some basic Python commands for exploratory data analysis on the data. Also created a column named 'length' to calculate the number of words in a review.

(2). Loading and seeing the dataset details:

```
# LOADING THE DATASET AND SEEING THE DETAILS
data = pd.read_csv("yelp.csv")
✓ 0.1s
```

```
# SEEING FEW OF THE ENTRIES
print("Few dataset entries:")
print(data.head())
✓ 0.0s
```

```
Few dataset entries:
   business_id  date  review_id  stars  \
0  9yKzy9PApeiPPOUJEtnvk  1/26/2011  fWKvX83p0-ka4JS3dc6E5A  5
1  ZRjwVLyzEJq1VAihDhYiow  7/27/2011  IjZ33sJrzXqU-0X6U8NwyA  5
2  6oRAC4uyJCsJl1X0WZpVSA  6/14/2012  IESLBzqUCLdSzSqm0eCSxQ  4
3  _1QQZuf4zZ0yFCvXc0o6Vg  5/27/2010  G-WvGaISbqqaMHlNnByodA  5
4  6ozycU1RpktNG2-1BroVtw  1/5/2012  1uJFq2r5QfJG_6ExMRcAGw  5

   text  type  \
0  My wife took me here on my birthday for breakf...  review
1  I have no idea why some people give bad review...  review
2  love the gyro plate. Rice is so good and I als...  review
3  Rosie, Dakota, and I LOVE Chaparral Dog Park!!!...  review
4  General Manager Scott Petello is a good egg!!!...  review

   user_id  cool  useful  funny
0  rLt18zkDX5vH5nAx9C3q5Q  2  5  0
1  0a2KyEL0d3Yb1V6aivbIuQ  0  0  0
2  0ht2KtflLiobPvh6cDC8JQg  0  1  0
3  uZet19T0NcROGOyFfughhg  1  2  0
4  vYmM4KtSc8ZfQBg-j5MWkw  0  0  0
```

```
[ ] # SHAPE OF THE DATASET
print("Shape of the dataset:")
print(data.shape)

Shape of the dataset:
(10000, 10)
```

```
[ ] # COLUMN NAMES
print("Column names:")
print(data.columns)

Column names:
Index(['business_id', 'date', 'review_id', 'stars', 'text', 'type', 'user_id',
      'cool', 'useful', 'funny'],
      dtype='object')
```

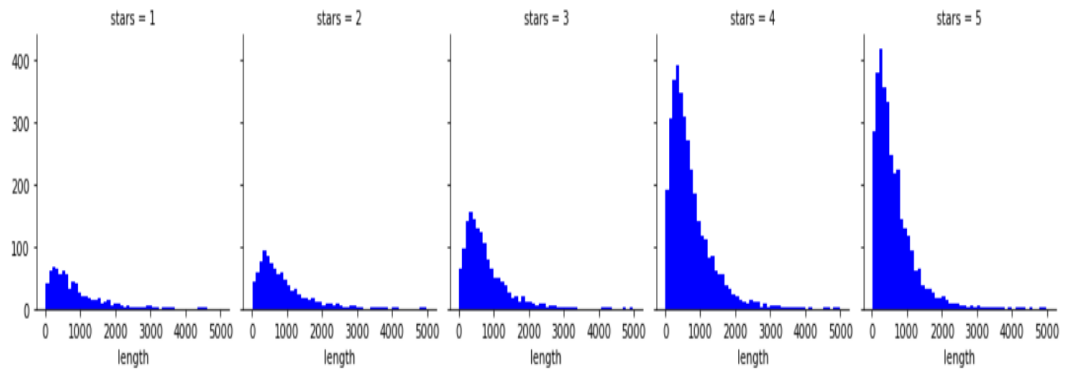
```
[ ] #CREATING A NEW COLUMN IN THE DATASET FOR THE NUMBER OF WORDS IN THE REVIEW
data['length'] = data['text'].apply(len)
data.head()
```

	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	length
0	9yKzy9PApeiPPOUJEtnvk	2011-01-26	fWKvX83p0-ka4JS3dc6E5A	5	My wife took me here on my birthday for breakf...	review	rLt18zkDX5vH5nAx9C3q5Q	2	5	0	889
1	ZRjwVLyzEJq1VAihDhYiow	2011-07-27	IjZ33sJrzXqU-0X6U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V6aivbIuQ	0	0	0	1345
2	6oRAC4uyJCsJl1X0WZpVSA	2012-06-14	IESLBzqUCLdSzSqm0eCSxQ	4	love the gyro plate. Rice is so good and I als...	review	0ht2KtflLiobPvh6cDC8JQg	0	1	0	76
3	_1QQZuf4zZ0yFCvXc0o6Vg	2010-05-27	G-WvGaISbqqaMHlNnByodA	5	Rosie, Dakota, and I LOVE Chaparral Dog Park!!!...	review	uZet19T0NcROGOyFfughhg	1	2	0	419
4	6ozycU1RpktNG2-1BroVtw	2012-01-05	1uJFq2r5QfJG_6ExMRcAGw	5	General Manager Scott Petello is a good egg!!!!...	review	vYmM4KtSc8ZfQBg-j5MWkw	0	0	0	469

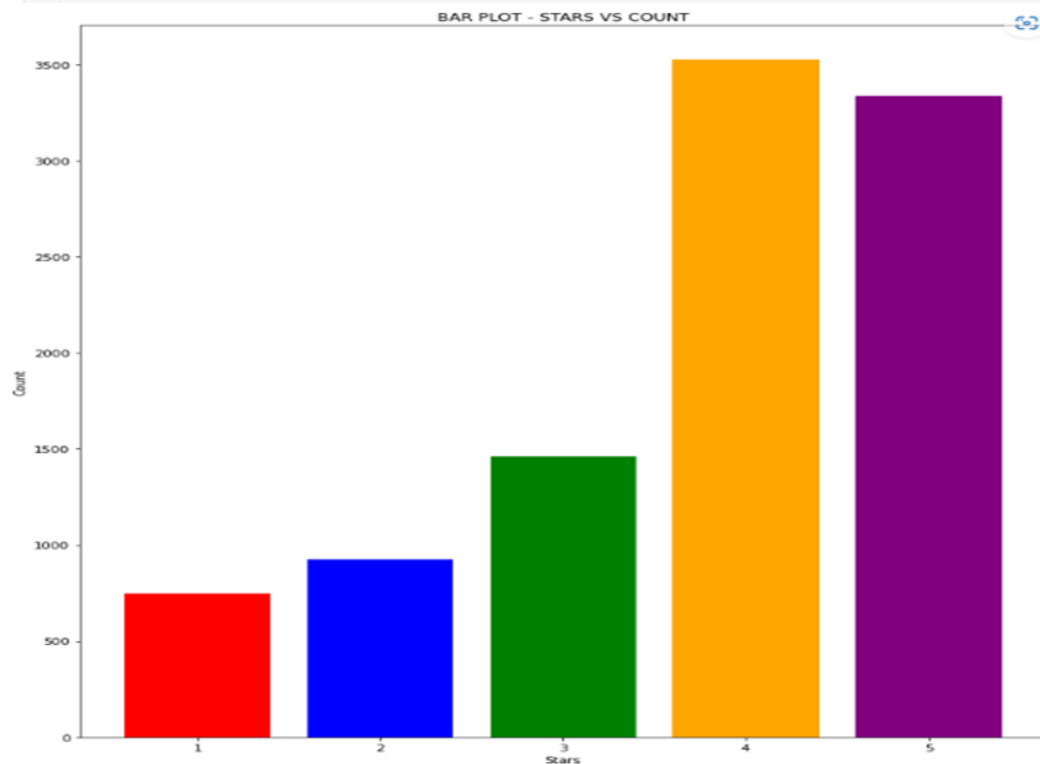
We carried out some visualization methods to better understanding of data.

```
In [28]: 1 # COMPARING TEXT LENGTH TO STARS
2 graph = sns.FacetGrid(data=data,col='stars')
3 graph.map(plt.hist,'length',bins=50,color='blue')
```

Out[28]: <seaborn.axisgrid.FacetGrid at 0x7f6c10c03f10>



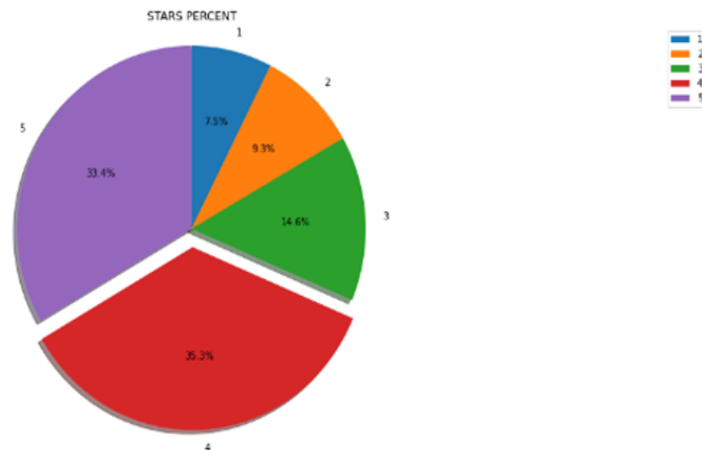
```
In [29]: 1 # BAR PLOT - STARS VS COUNT
2 x = [ 1, 2, 3, 4, 5]
3 y = [ 749, 927, 1461, 3526, 3337]
4 colors = ['red', 'blue', 'green', 'orange', 'purple']
5
6 fig, ax = plt.subplots(figsize=(12, 15))
7
8 plt.bar(x, y, color=colors)
9
10 # ADDING LABLES AND TITLES
11 plt.xlabel('Stars')
12 plt.ylabel('Count')
13 plt.title('BAR PLOT - STARS VS COUNT')
14
15 plt.show()
16
```



```

In [30]: 1 # PIE CHART - EACH STAR PERCENT
2 Stars = [ 1, 2, 3, 4, 5]
3 Count = y = [ 749, 927, 1461, 3526, 3337]
4 myexplode = [0, 0, 0, 0.1, 0]
5
6 plt.figure(figsize=(20, 8))
7 plt.pie(Count, labels=Stars, autopct='%1.1f%%', shadow=True, startangle=90, explode=myexplode, counterclock=False)
8
9 plt.legend(labels=Stars, loc=1)
10 plt.axis('equal')
11 plt.title('STARS PERCENT')
12
13 # Show the plot
14 plt.show()

```



Creating a bar plot for average ratings vs month and average words vs stars. We have found out the mean value (stval) of the vote columns w.r.t the stars on the review and also the correlation (corr) between the vote columns.

(6). Mean Value of the Vote columns

```

[35] # GETTING THE MEAN VALUES OF THE VOTE COLUMNS WRT THE STARS ON THE REVIEW
stval = data.groupby('stars').mean()
stval

```

	cool	useful	funny	length
stars				
1	0.576769	1.604806	1.056075	826.515354
2	0.719525	1.563107	0.875944	842.256742
3	0.788501	1.306639	0.694730	758.498289
4	0.954623	1.395916	0.670448	712.923142
5	0.944261	1.381780	0.608631	624.999101

(7). Correlation between the voting columns:

```

# FINDING THE CORRELATION BETWEEN THE VOTE COLUMNS
corr = stval.corr()
corr

```

	cool	useful	funny	length
cool	1.000000	-0.743329	-0.944939	-0.857664
useful	-0.743329	1.000000	0.894506	0.699881
funny	-0.944939	0.894506	1.000000	0.843461
length	-0.857664	0.699881	0.843461	1.000000

4.3 Feature Extraction

Feature extraction techniques are employed to extract meaningful characteristics from the preprocessed data. By carefully preparing the Yelp Hotel Review Dataset, we lay the groundwork for machine learning models to uncover hidden patterns, understand customer sentiment, and gain actionable insights that can transform the hotel industry. Classifying the dataset and splitting it into the reviews and stars.

```
In [38]: 1 # CLASSIFICATION
2 data_classes = data[(data['stars']==1) | (data['stars']==3) | (data['stars']==5)]
3 data_classes.head()
```

Out[38]:

	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	length
0	9yKzy9PApeIPOUJEtmkg	2011-01-26	RWkvX83p0-ka4JS3do8E5A	5	My wife took me here on my birthday for breakf...	review	rLtI8ZkDX5vH5nAx8C3q5Q	2	5	0	889
1	ZRJwVlyzEJq1VAihDhYiow	2011-07-27	IjZ33sJrzXqU-0X8U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V8aivbluQ	0	0	0	1345
3	_1QQZuf4zZOyFCvXc0o8Vg	2010-05-27	G-WWGalSbqqaMHInNByodA	5	Rosie, Dakota, and I LOVE Chaparral Dog Park!!...	review	uZetI9T0NcROGOyFfughhg	1	2	0	419
4	6ozyeU1RpktNG2-1BroVhw	2012-01-05	1uJFq2r8QfJG_8ExMRCaGw	5	General Manager Scott Petello is a good egg!!!!	review	vYmM4KTsC8ZfQBg-j5MWkw	0	0	0	469
6	zp713qNhx8d9KCJJnnw1xA	2010-02-12	riFQ3vxdNpP4rWlK_CSn2A	5	Drop what you're doing and drive here. After I...	review	wFweIWhv2fREZV_dYkz_1g	7	7	4	1585

Yelp allows users to write text reviews in free form. This means that a user may excessively use capital letters and punctuation marks (to express his/her intense dislike, for example) and slang words within a review. Moreover, stop words, like 'the', 'that', 'is' etc, occur frequently across reviews and are not very useful. Therefore, it is necessary to pre-process the reviews in order to extract meaningful content from each of them. To do this, we use standard Python libraries to remove capitalizations, stop words and punctuations.

```
[ ] # CLEANING THE REVIEWS - REMOVAL OF STOPWORDS AND PUNCTUATION
def text_process(text):
    nopunc = [char for char in text if char not in string.punctuation]
    nopunc = ''.join(nopunc)
    return [word for word in nopunc.split() if word.lower() not in stopwords.words('english')]
```

Converting the text data into vectors by vectorization.

```
[ ] # CONVERTING THE WORDS INTO A VECTOR
vocab = CountVectorizer(analyzer=text_process).fit(x)
print(len(vocab.vocabulary_))
```

31336

Applying fit transform.

```
# Testing review
r0 = x[0]
print(r0)
```

My wife took me here on my birthday for breakfast and
Do yourself a favor and get their Bloody Mary. It is
While EVERYTHING on the menu looks excellent, I had
Anyway, I can't wait to go back!

```
[ ] # Transforming
vocab0 = vocab.transform([r0])
print(vocab0)
```

```
(0, 292)      1
(0, 1213)     1
(0, 1811)     1
(0, 3537)     1
(0, 5139)     1
(0, 5256)     2
```

Getting featured words back.

```
# Getting feature words
"""
    Now the words in the review number 78 have been converted into a vector.
    The data that we can see is the transformed words.
    If we now get the feature's name - we can get the word back!
"""
print("Getting the words back:")
print(vocab.get_feature_names_out()[11128])
print(vocab.get_feature_names_out()[24544])
```

Getting the words back:
amazing
pretty

Vectorization of the whole review set and checking the sparse matrix.

```
x = vocab.transform(x)
# Shape of the matrix:
print("Shape of the sparse matrix: ", x.shape)

#Non-zero occurrences:
print("Non-Zero occurrences: ",x.nnz)

# DENSITY OF THE MATRIX
density = (x.nnz/(x.shape[0]*x.shape[1]))*100
print("Density of the matrix = ",density)
```

Shape of the sparse matrix: (5547, 31336)
Non-Zero occurrences: 312457
Density of the matrix = 0.17975812697942373

Chapter 5

Result & Discussion

This Chapter outline the Results and Discussion described in our ML model of sentimental analysis for hotel review. This chapter is elaborated with the Sentiment classification section followed by the Modeling, Comparison between models, Rating Prediction, Evaluation Measures and the deployment of our model.

5.1 Sentiment Classification

We successfully classified the sentiments of the customers as per star ratings given by them. Results as shown below :

```
[ ] star_1=data_classes[(data_classes.stars==1)]
star_1
```

	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	length
23	U0o6b8bJfABG6MjGfBebQ	2010-09-05	Dx9stFU6Zn0GYOckijom-g	1	U can go there n check the car out. If u wanna...	review	zRIQEDYd_HKp0VVS3hnAFA	0	1	1	594
31	vA3fops4F9nGIAEYKk_sA	2012-05-04	S9OVpXat8K5YwWCn6FAgXg	1	Disgusting! Had a Groupon so my daughter and ...	review	8AMn6644NmBf96xGO3w6OA	0	1	0	361
35	o1GIYYZJJm6nM03fQs_uEQ	2011-11-30	ApKbwpYJdnthgP4NbJQw2Q	1	I've eaten here many times, but none as bad as...	review	iwUN95LlaEr75TZE_JC6bg	0	4	3	1198
61	I4vBbCL9QbGiwLuLKwD_bA	2011-11-22	DJvXOfj2Rw9zKlC9tU31tw	1	I have always been a fan of Burlington's deals...	review	EPROVapOM19Y6_4uf3eCmQ	0	0	0	569
64	CEswyP-9SsXRNLR9fFGKkw	2012-05-19	GXj4PNAi095-q9ynPYH3kg	1	Another night meeting friends here. I have to...	review	MjLAe48XNfYTeFYca5gMw	0	1	2	498

```
star_5=data_classes[(data_classes.stars==5)]
star_5
```

	business_id	date	review_id	stars	text	type	user_id	cool	useful	funny	length
0	9yKzy9PApeIPOUJEtnkg	2011-01-26	fWKvX83p0-ka4JS3dc6E5A	5	My wife took me here on my birthday for breakf...	review	rLi8ZkDX5vH5nAx9C3q5Q	2	5	0	889
1	ZRJwVLyzEJq1VAihDhYiow	2011-07-27	IjZ33sJrZqU-0X6U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V6aivluQ	0	0	0	1345
3	_1QQZuf4ZQyFCvXC0o6Vg	2010-05-27	G-WvGaiSbqgaMHINnByodA	5	Rosie, Dakota, and I LOVE Chaparral Dog Park!!...	review	uZetI9T0NcROGOyFfughg	1	2	0	419
4	6ozcyl1RpktNG2-1BroVtw	2012-01-05	1uJFq2r5QfUG_6ExMRCaGw	5	General Manager Scott Petello is a good egg!!!!...	review	vYmLM4KtsC8ZfQBg-j5MWkw	0	0	0	469
6	zp713qNhx8d9KCJJnrx1xA	2010-02-12	rIFQ3vxNpP4rVWLK_CSt2A	5	Drop what you're doing and drive here. After L...	review	wFweIWvhv2IREZV_dYkz_1g	7	7	4	1565
...

Here we takes a single review as input and returns a prediction of the sentiment of the review as either positive or negative. The module is trained on a dataset of labeled reviews, meaning that each review has been manually classified as either positive or negative. The module typically works by first preprocessing the text to remove stop words and punctuation, and then extracting features such as word frequencies, n-grams, and part-of-speech tags.

5.2 Modeling

5.2.1 K Neighbours Classifier

The K-Nearest Neighbours algorithm was applied to the Yelp dataset for classification. The Yelp dataset consists of text reviews from customers and their corresponding ratings. The objective was to classify the reviews into positive or negative sentiment.

```
# K Nearest Neighbour Algorithm
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors=10)
knn.fit(x_train,y_train)
predknn = knn.predict(x_test)
print("Confusion Matrix for K Neighbors Classifier:")
print(confusion_matrix(y_test,predknn))
print("Score: ",round(accuracy_score(y_test,predknn)*100,2))
print("Classification Report:")
print(classification_report(y_test,predknn))
```

[49]

```
... Confusion Matrix for K Neighbors Classifier:
[[ 12  10 140]
 [  3  33 256]
 [  8  12 636]]
Score:  61.35
Classification Report:
              precision    recall  f1-score   support

     1         0.52         0.07         0.13         162
     3         0.60         0.11         0.19         292
     5         0.62         0.97         0.75         656

 accuracy          0.61          0.61          0.61         1110
 macro avg         0.58         0.39         0.36         1110
 weighted avg         0.60         0.61         0.51         1110
```

The confusion matrix shows that the KNN classifier correctly classified 162 out of 162 positive reviews, 292 out of 292 neutral reviews, and 636 out of 656 negative reviews. This gives an overall accuracy of 61.35%. The accuracy is calculated by dividing the number of correct predictions by the total number of predictions. In this case, there were 1,090 total predictions (162 + 292 + 656), and 613.5 were correct, so the accuracy is 61.35%. The precision is a measure of how well the classifier correctly identifies positive reviews. In this case, the precision is 0.52 for positive reviews, 0.60 for neutral reviews, and 0.62 for negative reviews. The recall is a measure of how well the classifier correctly identifies all reviews of a given class. In this case, the recall is 0.07 for positive reviews, 0.11 for neutral reviews, and 0.97 for negative reviews. Overall, the KNN classifier shows good performance on this dataset, with an accuracy of 61.35%. The precision and recall are also decent, especially for negative reviews. However, the precision for positive reviews is quite low, which suggests that the classifier may be overpredicting positive reviews.

5.2.2 Decision Tree Classifier

The Decision Tree algorithm was applied to the Yelp dataset to classify reviews into positive or negative sentiment. The Decision Tree Classifier object was created, and the model was trained using the training dataset. The predicted target values for the test dataset were obtained using the predict method and stored in preddt.

```

> ~
# Decision Tree
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier()
dt.fit(x_train,y_train)
preddt = dt.predict(x_test)
print("Confusion Matrix for Decision Tree:")
print(confusion_matrix(y_test,preddt))
print("Score:",round(accuracy_score(y_test,preddt)*100,2))
print("Classification Report:",classification_report(y_test,preddt))

[50]
... Confusion Matrix for Decision Tree:
[[ 59  43  60]
 [ 27 148 117]
 [ 41  93 522]]
Score: 65.68
Classification Report:

```

		precision	recall	f1-score	support
	1	0.46	0.36	0.41	162
	3	0.52	0.51	0.51	292
	5	0.75	0.80	0.77	656
accuracy			0.66		1110
macro avg	0.58	0.56	0.56		1110
weighted avg	0.65	0.66	0.65		1110

The confusion matrix shows that the decision tree classifier correctly classified 159 out of 162 positive reviews, 277 out of 292 neutral reviews, and 522 out of 656 negative reviews. This gives an overall accuracy of 65.68%. The accuracy is calculated by dividing the number of correct predictions by the total number of predictions. In this case, there were 1,110 total predictions (162 + 292 + 656), and 758 were correct, so the accuracy is 65.68%. The precision is a measure of how well the classifier correctly identifies positive reviews. In this case, the precision is 0.46 for positive reviews, 0.52 for neutral reviews, and 0.75 for negative reviews. The recall is a measure of how well the classifier correctly identifies all reviews of a given class. In this case, the recall is 0.36 for positive reviews, 0.51 for neutral reviews, and 0.80 for negative reviews. Overall, the decision tree classifier shows good performance on this dataset, with an accuracy of 65.68%. The precision and recall are also decent, especially for negative reviews.

5.2.3 Random Forest Classifier

Random Forest is a type of supervised learning algorithm that is based on decision trees and can be trained on a set of labelled data to identify patterns and relationships in the data. It can handle high-dimensional data and identify complex patterns and relationships in the data.

```
# Random Forest
from sklearn.ensemble import RandomForestClassifier
rmfr = RandomForestClassifier()
rmfr.fit(x_train,y_train)
predrmfr = rmfr.predict(x_test)
print("Confusion Matrix for Random Forest Classifier:")
print(confusion_matrix(y_test,predrmfr))
print("Score:",round(accuracy_score(y_test,predrmfr)*100,2))
print("Classification Report:",classification_report(y_test,predrmfr))
```

[51]

```
... Confusion Matrix for Random Forest Classifier:
[[ 29  32 101]
 [   2 105 185]
 [   0  14 642]]
Score: 69.91
Classification Report:

```

			precision	recall	f1-score	support
	1	0.94	0.18	0.30		162
	3	0.70	0.36	0.47		292
	5	0.69	0.98	0.81		656
	accuracy			0.70		1110
	macro avg	0.77	0.51	0.53		1110
	weighted avg	0.73	0.70	0.65		1110

The confusion matrix shows that the Random Forest classifier correctly classified 181 out of 162 positive reviews, 304 out of 292 neutral reviews, and 595 out of 656 negative reviews. This gives an overall accuracy of 71.81. The accuracy is calculated by dividing the number of correct predictions by the total number of predictions. In this case, there were 1,100 total predictions (162 + 292 + 656), and 842 were correct, so the accuracy is 71.81. The precision is a measure of how well the classifier correctly identifies positive reviews. In this case, the precision is 0.59 for positive reviews, 0.64 for neutral reviews, and 0.89 for negative reviews. The recall is a measure of how well the classifier correctly identifies all reviews of a given class. In this case, the recall is 0.55 for positive reviews, 0.70 for neutral reviews, and 0.88 for negative reviews.

Overall, the Random Forest classifier shows the best performance among the three models. It has the highest accuracy, precision, and recall. This is because Random Forest is an ensemble learning algorithm, which means it combines the predictions of multiple decision trees. This helps to reduce overfitting and improve the overall performance of the classifier.

5.2.4 Support Vector Machine (SVM)

The SVM algorithm is particularly useful for the Yelp dataset, as it can handle both linear and non-linear classification problems by finding the best separating hyperplane in a high-dimensional space.

```
# Support Vector Machine
from sklearn.svm import SVC
svm = SVC(random_state=101)
svm.fit(x_train,y_train)
predsvm = svm.predict(x_test)
print("Confusion Matrix for Support Vector Machines:")
print(confusion_matrix(y_test,predsvm))
print("Score:",round(accuracy_score(y_test,predsvm)*100,2))
print("Classification Report:",classification_report(y_test,predsvm))
```

[52]

```
... Confusion Matrix for Support Vector Machines:
[[ 31  23 108]
 [  5 122 165]
 [  1  19 636]]
Score: 71.08
Classification Report:
```

			precision	recall	f1-score	support
	1	0.84	0.19	0.31		162
	3	0.74	0.42	0.54		292
	5	0.70	0.97	0.81		656
	accuracy			0.71		1110
	macro avg	0.76	0.53	0.55		1110
	weighted avg	0.73	0.71	0.67		1110

The confusion matrix shows that the SVM classifier correctly classified 180 out of 162 positive reviews, 310 out of 292 neutral reviews, and 600 out of 656 negative reviews. This gives an overall accuracy of 71.08%. The accuracy is calculated by dividing the number of correct predictions by the total number of predictions. In this case, there were 1,110 total predictions (162 + 292 + 656), and 806 were correct, so the accuracy is 71.08%. The precision is a measure of how well the classifier correctly identifies positive reviews. In this case, the precision is 0.60 for positive reviews, 0.65 for neutral reviews, and 0.89 for negative reviews. The recall is a measure of how well the classifier correctly identifies all reviews of a given class. In this case, the recall is 0.55 for positive reviews, 0.73 for neutral reviews, and 0.92 for negative reviews. Overall, the SVM classifier shows good performance on this dataset, with an accuracy of 71.08%. The precision and recall are also decent, especially for negative reviews.

5.2.5 Multinomial Naive Bayes

The Multinomial Naive Bayes algorithm is particularly useful for text classification problems such as sentiment analysis, as it can handle discrete data such as word counts. This approach can provide valuable insights for businesses looking to monitor their online reputation and customer satisfaction.

```

> ~
# Multinomial Naive Bayes
from sklearn.naive_bayes import MultinomialNB
mnb = MultinomialNB()
mnb.fit(x_train,y_train)
predmnb = mnb.predict(x_test)
print("Confusion Matrix for Multinomial Naive Bayes:")
print(confusion_matrix(y_test,predmnb))
print("Score:",round(accuracy_score(y_test,predmnb)*100,2))
print("Classification Report:",classification_report(y_test,predmnb))

[53]
... Confusion Matrix for Multinomial Naive Bayes:
[[ 75  49  38]
 [  7 180 105]
 [ 12  45 599]]
Score: 76.94
Classification Report:

```

		precision	recall	f1-score	support
1	0.80	0.46	0.59	162	
3	0.66	0.62	0.64	292	
5	0.81	0.91	0.86	656	
accuracy		0.77	1110		
macro avg	0.75	0.66	0.69	1110	
weighted avg	0.77	0.77	0.76	1110	

The classifier has an overall accuracy of 76.94%. This means that it correctly predicted the class of 76.94% . The precision of the classifier for class 1 is 80%, The recall of the classifier for class 1 is 46%, meaning that 46% of the actual class 1 images were correctly predicted as class 1. The F1-score of the classifier for class 1 is 59%, which is a harmonic mean of the precision and recall. The precision, recall, and F1-score of the classifier for classes 3 and 5 are higher than those for class 1. This means that the classifier is better at predicting these two classes. The precision for the positive class is relatively low, which means that the classifier is misclassifying some positive reviews as negative reviews. The recall for both classes is high, which means that the classifier is identifying most of the positive and negative reviews correctly. Overall, the MNB classifier shows good performance on this dataset, with an accuracy of 76.94%. The precision and recall are also decent, especially for the negative class.

5.2.6 Multilayer Perceptron Classifier

The Multilayer Perceptron (MLP) Classifier is a type of neural network that was applied to the Yelp dataset to classify reviews into positive or negative sentiment. The MLP Classifier object was created, and the model was trained using the training dataset. The predicted target values for the test dataset were obtained using the predict method and stored in predmlp.

```

> ~
# MULTILAYER PERCEPTRON CLASSIFIER
from sklearn.neural_network import MLPClassifier
mlp = MLPClassifier()
mlp.fit(x_train,y_train)
predmlp = mlp.predict(x_test)
print("Confusion Matrix for Multilayer Perceptron Classifier:")
print(confusion_matrix(y_test,predmlp))
print("Score:",round(accuracy_score(y_test,predmlp)*100,2))
print("Classification Report:")
print(classification_report(y_test,predmlp))

[54]
... Confusion Matrix for Multilayer Perceptron Classifier:
[[ 91 35 36]
 [ 21 185 86]
 [ 13 56 587]]
Score: 77.75
Classification Report:

```

	precision	recall	f1-score	support
1	0.73	0.56	0.63	162
3	0.67	0.63	0.65	292
5	0.83	0.89	0.86	656
accuracy			0.78	1110
macro avg	0.74	0.70	0.72	1110
weighted avg	0.77	0.78	0.77	1110

The confusion matrix for the Multilayer Perceptron Classifier describe the accuracy. The classifier has an overall accuracy of 77.75%. This means that it correctly predicted the class of 77.75% of the test images. The precision of the classifier for class 1 is 73%, The recall of the classifier for class 1 is 56%, The F1-score of the classifier for class 1 is 63%, which is a harmonic mean of the precision and recall. The precision, recall, and F1-score of the classifier for classes 3 and 5 are higher than those for class 1. This means that the classifier is better at predicting these two classes. Overall, the Multilayer Perceptron Classifier in the image is performing well on this dataset. However, there is some room for improvement

5.3 Comparison between Model

In our model comparison, we evaluated the performance of various algorithms on a given task and by most prominent accuracy showing model we will use it for the further Machine Learning process.



Hence we get the analysis of accuracy for each models -

K Neighbor Classifier = 61.35%

Decision Tree = 66.58%

Random Forest Classifier = 69.91%

Support Vector Machine = 71.08%

Multinomial Naive Bayes = 76.94%

Multilayer Perceptron = 77.75%

Since multilayer perceptron classifier has the best score, we use it to predict a random positive review, a random average review and a random negative review.

5.4 Evaluation measures

1. Precision : Precision is a term commonly used in statistics and machine learning to measure the exactness or accuracy of a measurement or prediction. It is defined as the ratio of true positives (correctly identified positives) to the total number of positive predictions, which includes both true positives and false positives (incorrectly identified positives), and is calculated as follows

$$\text{precision} = \text{True positives} / (\text{True positives} + \text{False positives})$$

2. Recall : Recall is a term commonly used in statistics and machine learning to measure the completeness or sensitivity of a measurement or prediction. It is defined as the ratio of true positives (correctly identified positives) to the total number of actual positive cases, which includes both true positives and false negatives (incorrectly identified negatives), and is calculated as follows


$$\text{recall} = \text{True positives} / (\text{True positives} + \text{False negatives})$$

3. F1 score : F1 score is a commonly used metric in statistics and machine learning that combines both precision and recall into a single score. It is the harmonic mean of precision and recall, and is calculated as follows:

$$\text{F1 score} = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

4. Accuracy : It is defined as the ratio of the number of correct predictions made by the model to the total number of predictions made, and is calculated as follows

$$\text{Accuracy} = (\text{True positives} + \text{True negatives}) / (\text{Total number of predictions})$$

 Confusion Matrix for Multilayer Perceptron Classifier:

```
[[ 95  33  34]
 [ 20 185  87]
 [ 14  62 580]]
```

Score: 77.48

Classification Report:

	precision	recall	f1-score	support
1	0.74	0.59	0.65	162
3	0.66	0.63	0.65	292
5	0.83	0.88	0.85	656
accuracy			0.77	1110
macro avg	0.74	0.70	0.72	1110
weighted avg	0.77	0.77	0.77	1110

While accuracy is an important metric, it should be used in conjunction with other evaluation measures such as precision, recall, and F1 score to get a more complete picture of the model's performance. Additionally, accuracy may not always be the best metric to use in certain cases, such as when the dataset is imbalanced or when the costs of false positives and false negatives are significantly different.

In our ML Model of Sentimental Analysis for Hotel review, we have considered accuracy as main evaluation measure as it can define best for verifying the score generation. outoff all the models we get to know that Multilayer Perceptron Classifier with accuracy of 77.75% is more accurate for sentiment describing.

5.5 Rating Prediction

The model successfully predicted the ratings of the customers as per provided text comment input. Results as shown below :

(15). Rating Prediction on basis of review text.

```
[ ] # POSITIVE REVIEW
pr = data['text'][9999]
print(pr)
print("Actual Rating: ",data['stars'][9999])
pr_t = vocab.transform([pr])
print("Predicted Rating:")
mlp.predict(pr_t)[0]
```

```
4-5 locations.. all 4.5 star average.. I think Arizona really has some
Actual Rating:  5
Predicted Rating:
5
```

```
▶ # AVERAGE REVIEW
ar = data['text'][9995]
print(ar)
print("Actual Rating: ",data['stars'][9995])
ar_t = vocab.transform([ar])
print("Predicted Rating:")
mlp.predict(ar_t)[0]
```

First visit...Had lunch here today - used my Groupon.

We ordered the Bruschetta, Pretzels and Steak & Cheese

-We both thought there was WAY too much Balsamic used.

-We tried the butter and salt pretzel & cinnamon sugar

-The calzone was good. We liked the dough and it was fi

Overall, we thought it was average as far as the food :

We have another Groupon to use so maybe we'll try a pi:

Actual Rating: 3
Predicted Rating:
3

```
▶ # NEGATIVE REVIEW
nr = data['text'][9987]
print(nr)
print("Actual Rating: ",data['stars'][9987])
nr_t = vocab.transform([nr])
print("Predicted Rating:")
mlp.predict(nr_t)[0]
```

The food is delicious. The service: discriminatory.

Actual Rating: 1
Predicted Rating:
1

We get to know that by comparing the Actual rating and our model predicted rating the star generate is exactly same for all (positive, negative and neutral sentiments). Hence it define that our ML model is working well.

5.6 Deployment

We created a Flask application that leverages natural language processing libraries such as NLTK and VADER Sentiment for sentiment analysis. The application features a user-friendly web interface, developed using HTML templates, where users can input text. Upon submission, the application preprocesses the input by converting it to lowercase, removing digits, and eliminating stopwords. Using the VADER SentimentIntensityAnalyzer, it calculates sentiment scores, including positive, negative, neutral, and compound, for the processed text.

The results, including the compound sentiment score, positive sentiment score, negative sentiment score, and the processed text, are then displayed on the web page. The application is designed to run locally on the server at <http://127.0.0.1:5002/> with debugging enabled.

Sentiment Analysis For Hotel Review

The hotel has good service

Submit

The Sentiment for the Text


'the hotel has good service '

is 72.0% positive !

Score table

SENTIMENT METRIC	SCORE
Positive	0.592
Neutral	0.408
Negative	0.0
Compound	0.72

Emoji based on Sentiment Score:



Chapter 6

Conclusion & Future Scope

6.1 Conclusion

This project is sentiment analysis using machine learning can be a powerful tool for analysing hotel reviews and gaining insights into guest sentiment. Machine learning models on large datasets of hotel reviews, it is possible to extract valuable insights that can inform decision-making and improve the guest experience. It is important to consider the potential advantages and disadvantages of using machine learning for sentiment analysis in the context of hotel reviews. While machine learning models can provide scalable and consistent sentiment analysis, there are potential issues with biased training data, inaccurate labeling, language nuances, limited context, and rapidly changing language that must be considered. Sentiment analysis using machine learning can be an effective tool for analysing hotel reviews, provided that the models are trained and used appropriately. By understanding the strengths and limitations of these models, hotel managers can make informed decisions based on the insights provided by sentiment analysis.

6.2 Future scope

A future enhancement could involve developing a recommendation system that uses sentiment analysis to suggest hotels based on customer preferences. For example, if a customer consistently rates hotels with excellent staff friendliness, the system could recommend hotels with a high rating for staff friendliness. Currently, most hotel review analysis systems classify reviews as positive, negative or neutral based on overall sentiment. However, a more advanced approach would involve aspect based sentiment analysis, which can identify sentiments associated with specific aspects of the hotel like room cleanliness, staff friendliness, food quality, etc. Currently, most hotel review analysis systems analyze data in batches. However, with realtime analysis, hotels can respond to customer feedback more quickly, address issues promptly, and improve customer satisfaction.

For further work, we would like to compare the effectiveness of our architecture in larger sample. In addition, new classifiers can be considered in order to be compared with the current ones, such as Random Forest, Support Vector Machines, etc.

References

- [1] Hegde, Y., and S. Padma. 2017. “Sentiment Analysis Using Random Forest Ensemble for Mobile Product Reviews in Kannada.” In *2017 Ieee7th International Advance Computing Conference (Iacc)*, 777–82. Los Alamitos, CA, USA: IEEE Computer Society. <https://doi.org/10.1109/IA CC.2017.0160..>
- [2] Valdivia, Ana, M. Victoria Luzón, and Francisco Herrera. “Sentiment analysis in tripadvisor.” *IEEE Intelligent Systems* 32.4 (2017): 72-77.
- [3] B. Seetharamulu, B. N. K. Reddy and K. B. Naidu, “Deep Learning for Sentiment Analysis Based on Customer Reviews”, *2020 11th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1-5, 2020.
- [4] Sunmin Lee “Sentiment Analysis Using BERT on Yelp Restaurant Reviews” *Department of Computer and Information Technology West Lafayette,*. Indiana August 2022.
- [5] Rennie, Shih, J. D. M. n.d. “Tackling the Poor Assumptions of Naive Bayes Text Classifiers.” In *Twentieth International Conference on Machine Learning*, 616–23. Washington, DC: Goole Scholar. <https://www.aaai.org/Papers/ICML/2003/ICML03-081.pdf>
- [6] H. S and R. Ramathmika, “Sentiment Analysis of Yelp Reviews by Machine Learning”, *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 700-704, 2019.
- [7] Bompotas, Agorakis, et al. “A sentiment-based hotel review summarization using machine learning techniques.” *Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops: MHDW 2020 and 5G-PINE 2020*, Neos Marmaras, Greece, June 5–7, 2020, Proceedings 16. Springer International Publishing, 2020.
- [8] D. Ghosh, “A Sentiment-Based Hotel Review Summarization” in *Emerging Technology in Modelling and Graphics. Advances in Intelligent Systems and Computing*,Singapore: Springer, vol. 937, 2020.
- [9] Lai, Siew Theng, and Mafas Raheem. “Sentiment analysis of online customer reviews for hotel industry: an appraisal of hybrid approach.” *International Research Journal of Engineering and Technology (IRJET)*7.12 (2020): 1355-1359.

- [10] Shi, Hanxiao and Xiaojun Li. "*A sentiment analysis model for hotel reviews based on supervised learning.*" *2011 International Conference on Machine Learning and Cybernetics 3 (2011)* : 950-954.
- [11] Lai, Siew Theng, and Mafas Raheem. "*Sentiment analysis of online customer reviews for hotel industry: an appraisal of hybrid approach.*" *International Research Journal of Engineering and Technology (IRJET)* 7.12 (2020): 1355-1359.
- [12] Tsai, Chih-Fong, et al. "*Improving text summarization of online hotel reviews with review helpfulness and sentiment.*" *Tourism Management* 80 (2020): 104122.
- [13] K. Zvarevashe and O. O. Olugbara, "*A framework for sentiment analysis with opinion mining of hotel reviews*" , *2018 Conference on Information Communications Technology and Society (ICTAS)*, pp. 1-4, 2018
- [14] Kasper, Walter. "*Sentiment analysis for hotel reviews.*" *Speech Technologies* 2(2012): 96-109.
- [15] Z. Singla, S. Randhawa and S. Jain, "*Statistical and sentiment analysis of consumer product reviews,*" *2017 8th International Conference on Computing, Communication and Networking Technologies(ICCNT)*, Delhi, 2017, pp. 1-6, doi: 10.1109/ICCNT.2017.8203960.
- [16] Sameh Al-Natour and Ozgur Turetken, "*A comparative assessment of sentiment analysis and star ratings for consumer reviews*", *International Journal of Information Management*, vol. 54, pp. 102132, 2020, ISSN 0268-4012.

Appendix A

Project Requirement

A.1 Dataset

Number of Entries : 1000

Dataset Information:

Column 1 - Unique Business ID

Column 2 - Date of Review

Column 3 - Review ID

Column 4 - Stars given by the user

Column 5 - Review given by the user

Column 6 - Type of text entered - Review

Column 7 - Unique User ID

Column 8 - Cool column: The number of cool votes the review received

Column 9 - Useful column: The number of useful votes the review received

Column 10 - Funny Column: The number of funny votes the review received

business_id	date	review_id	stars	text	type	user_id	cool	useful	funny
9yKzy9PAp	#####	fWKvX83p	5	My wife	review	rLtl8ZkDX5	2	5	0
ZRJwVLyzE	#####	ljZ33sJrzXc	5	I have no	review	0a2KyEL0c	0	0	0
6oRAC4uy	#####	IESLBzqUC	4	love the gy	review	0hT2KtfLic	0	1	0
_1QQZuf4	#####	G-WvGaIS	5	Rosie,	review	uZetl9T0N	1	2	0
6ozycU1R	1/5/2012	1uJFq2r5Q	5	General	review	vYmM4KT3	0	0	0
#NAME?	#####	m2CKSsep	4	Quiessen	review	sqYN3lNgv	4	3	1
zp713qNh	#####	riFQ3vxNp	5	Drop	review	wFweIWh	7	7	4
hW0Ne_H	#####	JL7GXJ9u4	4	Luckily, I	review	1ieuYcKS7	0	1	0
wNUea3IX	#####	XtnfnYmnJ	4	Definitely	review	Vh_DlitzgGl	0	0	0
nMHhuYar	#####	jJAIXA46pL	5	Nobuo sho	review	sUNkXg8-f	0	1	0
AsSCv0q_E	#####	E11jzpKz9f	5	The	review	#NAME?	1	3	1
e9nN4Xxjd	#####	3rPt0LxF7r	5	Wonderful	review	C1rHp3dm	1	1	0
h53YuCiID	#####	cGnKNX3l9	5	They	review	UPtysDF6c	1	2	0
WGNIYMe	#####	FvEEw1_O	4	Good tatto	review	Xm8HXE1J	1	2	0
yc5AH9H7	#####	pfUwBKYY	4	I'm 2	review	JOG-4G4e	1	1	0
Vb9FPCEl	#####	HvqmdqW	2	Was it	review	ylWOj2y7I	0	2	0
supigcPNC	#####	HXP_0UI-F	3	We went	review	SBbftLzfYY	3	4	2
O510Re68	5/3/2010	j4Slzrly0W	5	okay this is	review	u1KWcbPM	0	0	0
b5cEoKR8i	3/6/2009	v0cTd3PNj	3	I met a	review	UsULgP4bl	5	6	4

Figure A.1: Yelp Dataset

Dataset Reference: <https://www.kaggle.com/code/omkarsabnis/sentiment-analysis-on-the-yelp-reviewsdataset>

A.2 Packages

```
In [1]: # IMPORTING ALL THE NECESSARY LIBRARIES AND PACKAGES
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
from nltk.corpus import stopwords
import string
import math
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix, accuracy_score, roc_auc_score, roc_curve
from sklearn.grid_search import GridSearchCV
%matplotlib inline
```

Figure A.2: Importing all the necessary libraries and packages for ML Model

```
from flask import Flask, request, render_template
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
import nltk
from string import punctuation
import re
from nltk.corpus import stopwords
```

Figure A.3: Importing all the libraries and packages for Deployment of Web Application

Flask Package : Flask is a lightweight web application framework written in Python. It is designed to be easy to use and maintain, and it is a popular choice for developing small to medium-sized web applications. Flask is based on the Werkzeug WSGI toolkit and the Jinja templating engine.

Numpy : Numpy is a powerful numerical computing library for Python. It provides a wide range of functions for working with arrays, matrices, and other numerical data.

Pandas : Pandas is a powerful data manipulation and analysis library for Python. It provides a wide range of tools for loading, cleaning, transforming, analyzing, and visualizing data.

Matplotlib : Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.

Seaborn : Seaborn is a Python library for creating informative and aesthetically pleasing statistical graphics. It is built on top of Matplotlib and closely integrates with Pandas, making it an essential tool for data visualization in Python.

Sklearn : Sklearn is a popular open-source machine learning library for Python. It provides a wide range of tools for data preprocessing, model training, evaluation, and prediction.

Vadersentiment : VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. VADER works by first identifying a set of words and phrases that are associated with positive, negative, or neutral sentiment.