HACKATHON

3.2 SENTIMENT ANALYSIS

Problem Statement:

Building a sentiment analysis model to analyze customer reviews and feedback for a hospitality company. The model should be able to classify reviews as positive, negative or neutral.

Data Exploration:

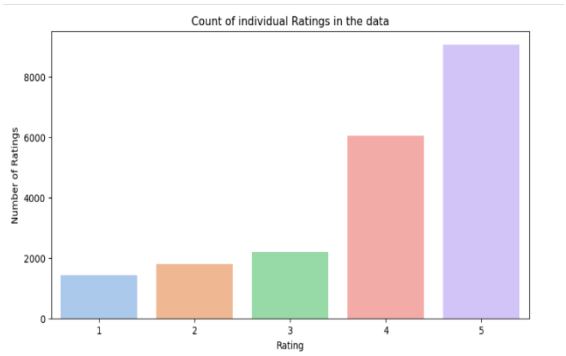
The dataset comprises customer reviews from a hospitality company. The target feature in the data is Review Column.

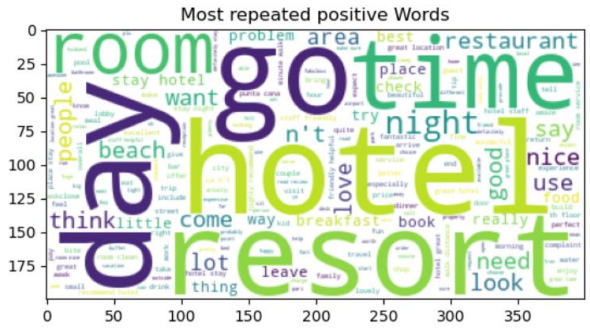
Data Cleaning :

- Checked data for missing values, duplicates and null entries.
- Ensuring data quality by addressing anomalies and inconsistencies

Exploratory Data Analysis:

EDA helps understand the distribution of data, such as the proportion of positive, negative, and neutral reviews.





Vizualised the distribution of the characteristics of the sentiments of each type with the help of seaborn and matplotlib.

Data Preprocessing :

Text preprocessing involves removal of stopwords, special characters, url's, etc..

```
: def lemmatization(x):
    list_stp = stopwords.words("english")
    wl = WordNetLemmatizer()
    def lemmatize_text(text):
        words = word_tokenize(text)
        lemmatized_words = [wl.lemmatize(word, pos="v") for word in words if word not in list_stp]
        return " ".join(lemmatized_words)
    return x.apply(lemmatize_text)
```

Model Selection :

- Deployed Bernoulli's Naïve Bayes and Multinomial Naïve Bayes models using CountVectorizer for Machine Learning models to predict the sentiment of the reviews
- Deployed a Deep Learning model using LSTM for sentiment analysis prediction

Model Evaluation :

- Evaluated the performance of the model using metrics like accuracy_score, classification report
- Calculated the performance of the model on both the training and test datasets.

Results:

- Achieved an accuracy score of 67% using the LSTM model
- Achieved an accuracy score of 69.5% using Naïve Bayes Models

Conclusion :

The sentiment analysis model successfully classifies customer reviews into positive, negative, or neutral categories. This can provide valuable insights into customer satisfaction and areas for improvement.

Future Improvements :

- As the data is imbalanced the model might be more inclined towards only onle class and result in decrease inperformance.
- The accuracy can be increased by performing hyperparameter tuning like GridSearchCV .