Sentimental Analysis of Text Classification with NLP

Sai Nityamani Sahith Matsa

Objective

This project focuses on building and evaluating a sentiment analysis model using natural language processing (NLP) techniques. The goal is to classify text data into positive, neutral, or negative sentiments and reflect on the ethical challenges in NLP models.

Dataset Description

The dataset, "tripadvisor_hotel_reviews", contains 20,491 entries with Review and Rating. The Review column includes customer feedback as textual data, while the Rating column provides corresponding numerical ratings ranging from 1 to 5. A significant portion of the ratings is skewed towards 5 and 4, indicating a positive sentiment bias. There are no missing values in the dataset, ensuring its readiness for analysis.
The Review column is of type "String", and Rating is an "Integer", making this dataset ideal for sentiment analysis and text classification tasks.

Data Preprocessing

This stage prepares raw text for analysis by transforming it into a structured and clean format. It begins with tokenization, which splits sentences into individual words while converting text to lowercase and removing punctuation. Next, stop-word removal eliminates common words that do not add significant meaning, reducing noise in the dataset. Finally, lemmatization converts words to their base forms while retaining relevant grammatical features, ensuring the data is concise, meaningful, and ready for modeling and feature extraction.

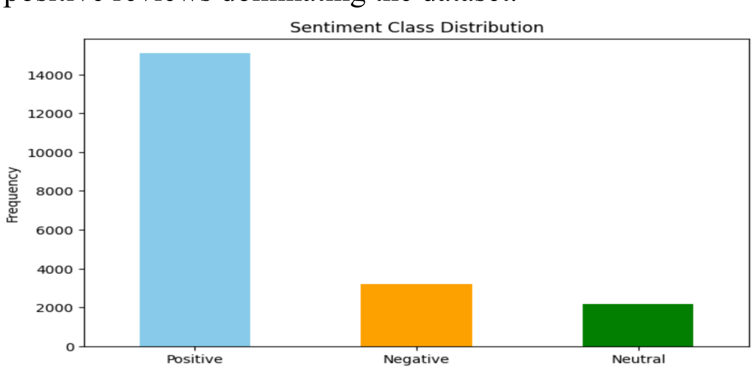
- **Tokenization**: Splits sentences into individual words using the sent_to_words function, converting text to lowercase and removing punctuation to simplify the data.
- **Stop-word Removal**: Filters out common words like "the," "is," and "and" that do not add significant meaning, ensuring only relevant terms are retained for analysis.
- **Lemmatization**: Converts words to their base forms (e.g., "running" to "run") while retaining important parts of speech such as nouns, verbs, adjectives, and adverbs to streamline the text for modelling.

Exploratory Data Analysis

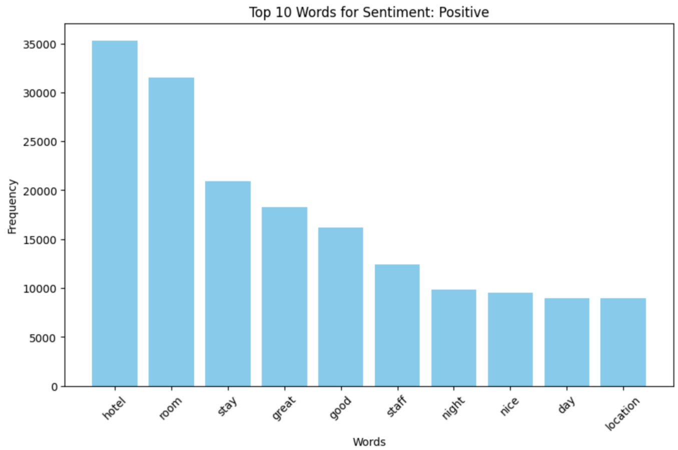
- Sentiment Class Distribution
- Top Words for Each Sentiment
- Word Clouds

Sentiment Class Distribution

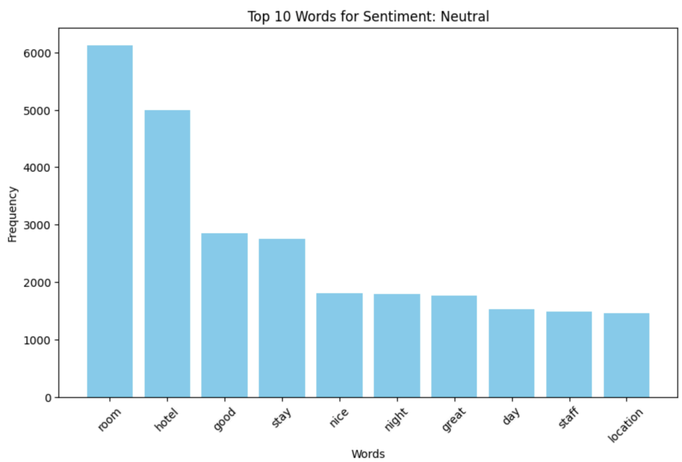
The bar chart shows the distribution of sentiment classes, highlighting a significant imbalance with positive reviews dominating the dataset.



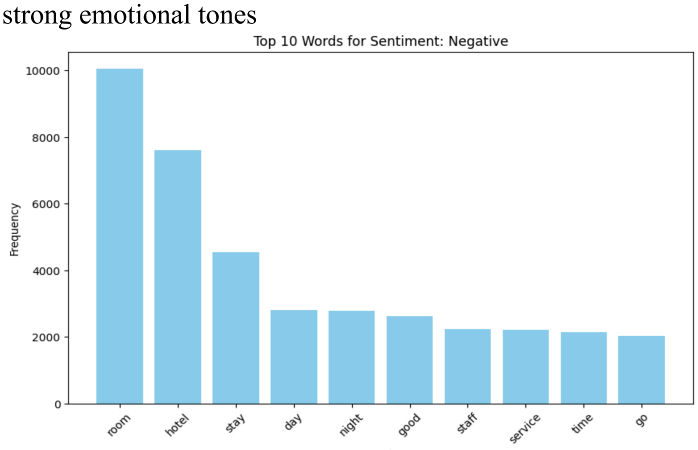
Top Words for Each Sentiment



Positive Sentiment: Common terms like "hotel," "room," "stay," and "great" emphasize customer satisfaction, highlighting enjoyable experiences and quality service



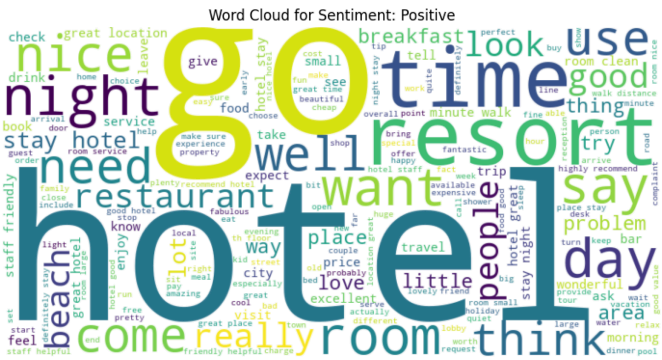
Neutral Sentiment: Frequent words such as "room," "hotel," and "good" indicate general observations without strong emotional tones



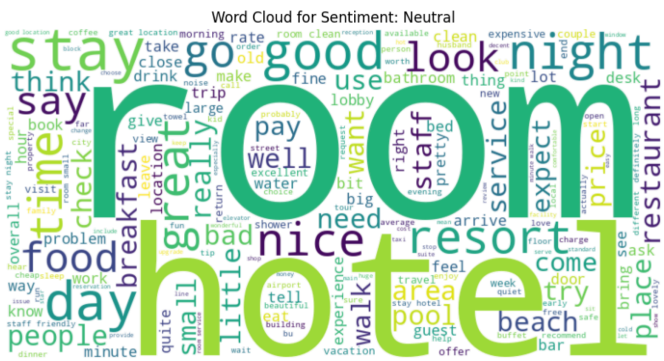
Negative Sentiment: Words like "room," "hotel," "stay," and "bad" dominate, reflecting dissatisfaction with specific aspects such as service or amenities

Word Clouds

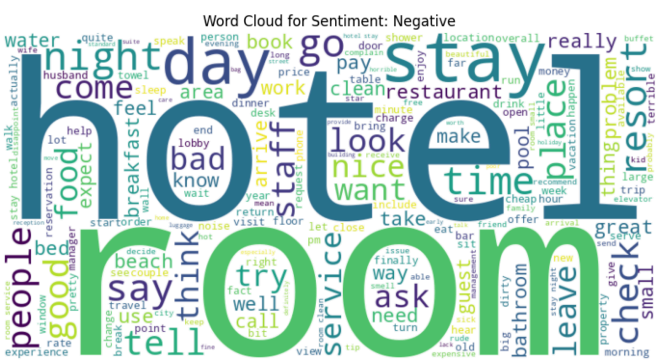
Word clouds visualize the top terms for each sentiment, offering a textual representation of key patterns. Positive reviews emphasize enjoyable stays and high-quality service, while negative reviews highlight specific issues like poor service or inadequate facilities



Positive Sentiment: Terms like “Hotel”, “go”, “time”, “resort”, “night”, “well”, “room” dominate.



Neutral Sentiment: Terms like “Hotel”, “room”, “night”, “stay”, “food”, “good”, dominate.



Negative Sentiment: Terms like “Hotel”, “room”, “day”, “stay”, “bad”, “time”, dominate.

The EDA highlights that certain words like "hotel," "room," and "stay" frequently appear across all sentiments, indicating their central role in customer experiences. Positive reviews often emphasize terms like "great," "good," and "staff," reflecting satisfaction with services, while negative reviews focus on issues using words like "bad," "service," and "dirty." Neutral reviews commonly use generic terms without strong emotional connotations. These word patterns provide critical insights into the linguistic cues driving sentiment, which can be leveraged to refine model performance and focus on differentiating nuanced expressions

Model Development and Evaluation

The model utilized Logistic Regression for sentiment classification. The text data was vectorized using TF-IDF with a maximum of 5000 features to reduce dimensionality and prevent overfitting. The data was split into 80% training and 20% testing subsets to ensure robust model training and evaluation.

Confusion Matrix

The confusion matrix highlights the model's performance across the sentiment categories:

- **Positive Sentiment:** The model achieved high accuracy, correctly classifying most positive reviews.
- **Neutral Sentiment:** Performance on neutral reviews was suboptimal, with many neutral reviews misclassified as positive or negative.
- **Negative Sentiment:** Negative reviews were classified with moderate accuracy, though some were misclassified as positive.

Classification Metrics

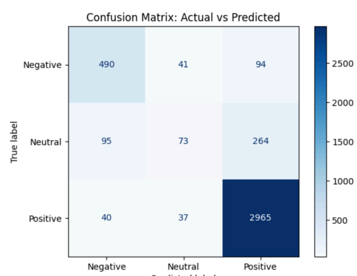
- **Accuracy:** The model achieved an overall accuracy of **86%**.
- **Precision:** High for positive (0.89) but lower for neutral (0.48).
- **Recall:** Excellent for positive (0.97) but poor for neutral (0.17).
- **F1-Score:** Indicates the model's balance between precision and recall; high for positive (0.93) but low for neutral (0.25).

Key Observations

- **Positive Sentiment:** The model performs very well with high precision (0.89) and recall (0.97), achieving an impressive F1-score of 0.93, making it highly reliable for identifying positive reviews.
- **Neutral Sentiment:** The model struggles significantly with neutral sentiment, showing low precision (0.48), recall (0.17), and F1-score (0.25), indicating challenges in distinguishing neutral reviews.
- **Negative Sentiment:** The model performs moderately for negative sentiment with precision and recall both at 0.78, achieving a balanced F1-score of 0.78, though there is room for improvement in reducing misclassification.

Accuracy: 0.86
Classification Report:

	precision	recall	f1-score
Negative	0.78	0.78	0.78
Neutral	0.48	0.17	0.25
Positive	0.89	0.97	0.93
accuracy			0.86
macro avg	0.72	0.64	0.66
weighted avg	0.83	0.86	0.84



Error Analysis

Error analysis highlights critical aspects of model misclassifications and their causes:

- **Index 2 (Neutral predicted as Positive):**
  The review includes positive phrases like "good location" and "great shopping," which led the model to classify it as Positive. This indicates an overemphasis on terms typically associated with positivity.
- **Index 29 (Negative predicted as Positive):**
  Despite containing strong negative cues like "horrible" and "dirty water," the review's length and descriptive nature, including neutral elements like "medical bill," confused the model. This shows the model's difficulty in prioritizing dominant sentiments in longer reviews.
- **Index 30 (Neutral predicted as Negative):**
  Negative terms like "terrible service" caused the model to misclassify, ignoring neutral and mildly positive elements like "curious fun experience." This suggests the model struggles to balance conflicting sentiment cues within a single review.

- **Index 46 (Neutral predicted as Positive):**
  Positive words such as "clean," "comfortable," and "quiet" influenced the model's prediction toward Positive. This reflects an overreliance on isolated positive words without considering the overall neutral tone.
- **Index 54 (Neutral predicted as Positive):**
  Strongly positive terms like "perfect location" and "spacious clean room" biased the model's prediction, ignoring the lack of strong enthusiasm in the overall sentiment. This demonstrates the challenge of identifying subtle neutrality.

These examples underline the model's tendency to overemphasize prominent sentiment-indicative words and its struggle to interpret complex or mixed sentiment contexts, pointing to areas for improvement like context-awareness and handling of lengthy reviews

Research Summary and Ethical Reflection

Summary:[Natural Language Processing for Sentiment Analysis: An Exploratory Analysis on Tweets**]**

The paper explores sentiment analysis on tweets using Natural Language Processing (NLP) techniques. Tweets, being short and informal, require specialized approaches different from traditional text sentiment analysis. The study emphasizes pre-processing to clean and normalize text for better machine interpretation, addressing challenges such as abbreviations, slang, and unstructured formats.

The proposed system comprises three main steps:
1. **Subjectivity Classification**: Distinguishing subjective tweets containing sentiment from objective ones.
2. **Semantic Association**: Associating sentiment lexicons to specific subjects within tweets using grammatical relationships.
3. **Polarity Classification**: Classifying sentiment polarity (positive, negative, or neutral) by evaluating associated sentiment lexicons using resources like SentiWordNet.

The system was tested using 1,513 manually labeled tweets about a Malaysian telecommunication service. Results were benchmarked against Alchemy API and machine learning algorithms (Naive Bayes, Decision Tree, and Support Vector Machine). The proposed system achieved higher accuracy (59.85%) than Alchemy API but was outperformed by Support Vector Machine with pre-processed tweets (64.95% accuracy).

Key findings indicate that pre-processing significantly improves sentiment classification performance, boosting accuracy, precision, and F-measure across tools. However, the system requires further refinement to handle the nuances of tweet-specific features like misspellings and comparative sentiments.

The study concludes with future directions to enhance system accuracy, particularly by improving pre-processing efficiency and incorporating larger datasets for training. This research underscores the importance of tailored NLP approaches for analyzing sentiment in informal, short-text domains like Twitter.

Ethical Reflection:

Sentiment analysis models often inherit biases present in the training data, which can lead to unfair or inaccurate predictions. For example, datasets might over-represent certain demographics, cultural viewpoints, or language styles, skewing the model toward these patterns. This can result in disproportionate misclassifications.

To mitigate these challenges, strategies include:

- **Data Diversity**: Ensure the training dataset is collected from diverse sources, covering a wide range of topics, regions, and demographics. This helps the model generalize better and reduces the risk of bias toward a specific group or context.
- **Pre-processing Checks**: During pre-processing, carefully examine and clean the dataset for any biased or irrelevant features. For example, removing identifiable demographic markers (e.g., names, locations) can help reduce unintended biases.