

Hotel Reviews Sentiment Analysis Using Natural Language Processing

1. Introduction

In today's digital era, online travel and hotel booking platforms such as Booking.com, TripAdvisor, and Expedia play a crucial role in influencing customer decisions. Travelers frequently share their experiences, opinions, and satisfaction levels through hotel reviews. While these reviews provide valuable insights for both potential customers and hotel management, the rapidly growing volume of textual feedback makes manual analysis inefficient and impractical.

Hotel reviews are typically unstructured and written in natural language, often containing mixed emotions, personal opinions, and varying writing styles. Extracting meaningful insights from such data is a challenging task. Sentiment Analysis, a subfield of Natural Language Processing (NLP), enables automated identification and classification of emotions expressed in text, making it highly suitable for analyzing customer feedback at scale.

With the increasing importance of customer satisfaction in the hospitality industry, automated sentiment analysis systems using Machine Learning (ML) and NLP techniques have become essential. These systems help hotels understand customer perceptions, monitor service quality, and make data-driven decisions.

This project focuses on building an intelligent system that automatically classifies hotel reviews into **Positive, Neutral, and Negative sentiments** using NLP techniques and machine learning algorithms.

The motivation for this project arises from the increasing reliance of travelers on online hotel reviews when making booking decisions. With thousands of reviews generated daily, hotel management faces challenges in manually analyzing customer feedback. Automated sentiment analysis provides an efficient solution to understand customer satisfaction levels, identify service gaps, and improve overall guest experience. This project aims to bridge the gap between raw textual feedback and actionable business insights.

2. Problem Definition

The main problem addressed in this project is:

How can we accurately classify hotel reviews into positive, negative, or neutral sentiments using machine learning and NLP techniques?

Hotel reviews often contain subjective opinions, mixed feedback, sarcasm, and informal language, making sentiment classification challenging.

Challenges:

- Unstructured and noisy text data
- Presence of mixed sentiments in a single review
- Class imbalance among sentiment categories
- Large vocabulary and high dimensionality
- Variations in writing style and length of reviews

3. Objectives of the Project

The primary objectives of this project are:

1. To study and understand customer sentiments expressed in hotel reviews
2. To preprocess hotel review text using NLP techniques
3. To extract meaningful textual features from reviews
4. To build machine learning models for sentiment classification
5. To evaluate and compare model performance
6. To identify important words and patterns influencing customer sentiment

3.1 Scope of the Project

The scope of this project is limited to the **sentiment classification of text-based hotel reviews** into **Positive, Neutral, and Negative** categories using Natural Language Processing techniques. The project focuses on analyzing the **content of customer reviews** rather than additional metadata such as reviewer demographics or hotel facilities.

The system is designed to work with structured datasets and can be extended to real-world applications such as:

- Customer feedback analysis systems
- Online reputation management tools
- Service quality monitoring platforms

Although the current implementation handles **English-language reviews only**, the methodology can be adapted for multilingual datasets with appropriate preprocessing techniques.

4. Literature Review

Sentiment analysis has been widely researched in the fields of text mining and NLP. Early approaches relied on rule-based sentiment lexicons, which were limited in scalability and adaptability.

Later, machine learning algorithms such as **Naive Bayes, Logistic Regression, and Support Vector Machines (SVM)** became popular due to their effectiveness in text classification tasks. Pang and Lee (2008) demonstrated the effectiveness of ML techniques for opinion mining.

Recent research focuses on deep learning and transformer-based models such as **BERT and RoBERTa**, which capture contextual meaning more effectively. While these models provide superior performance, they require higher computational resources. In this project, both traditional ML models and a pretrained RoBERTa model are explored for comparative analysis.

5. Dataset Description

Dataset Source

The dataset used in this project is the **Hotel Reviews Dataset** obtained from Kaggle.

URL:

<https://www.kaggle.com/datasets/jiashenliu/515k-hotel-reviews-data-in-europe>

Dataset Details

- Total reviews: Over 500,000
- Attributes include:
 - Positive Review
 - Negative Review
 - Reviewer Score
 - Review Date
 - Location details
 -

Why This Dataset Was Chosen?

This dataset was chosen due to its large size, real-world nature, and availability of both textual reviews and numerical ratings. The combination of positive and negative reviews provides a balanced view of customer experiences, making it ideal for sentiment analysis research.

Target Variable

Sentiment labels are derived from reviewer scores:

- **Positive:** Score ≥ 7
- **Negative:** Score ≤ 4
- **Neutral:** Score between 5 and 6

5.1 Dataset Exploration and Class Distribution

Initial exploration reveals that **positive reviews dominate the dataset**, followed by neutral and negative reviews. This class imbalance can influence model performance and must be considered during evaluation.

Understanding sentiment distribution helps in selecting suitable evaluation metrics such as precision, recall, and F1-score instead of relying only on accuracy.

Data Quality Discussion

Data quality checks revealed missing values in certain location attributes, which were handled using statistical imputation techniques. Duplicate reviews were removed to ensure unbiased model training.

6. Methodology

The project follows a structured machine learning pipeline:

1. Data Collection
2. Data Pre-processing
3. Feature Engineering
4. Exploratory Data Analysis (EDA)
5. Model Building
6. Model Evaluation

Workflow Explanation

Each stage of the methodology is designed to incrementally transform raw data into meaningful insights. Preprocessing improves data quality, feature extraction converts text into numerical form, and model training enables automated sentiment classification.

6.1 System Architecture

The system architecture consists of the following components:

1. **Input Layer** – Raw hotel review text
2. **Pre-processing Module** – Text cleaning and normalization
3. **Feature Extraction Module** – TF-IDF vectorization
4. **Classifier Module** – ML and DL models
5. **Output Layer** – Sentiment prediction

This modular design allows easy replacement or enhancement of individual components.

Architecture Advantages

One of the major challenges during preprocessing was handling long reviews containing mixed sentiments. Careful normalization and stemming were required to preserve sentiment-related words while removing noise.

7. Data Pre-processing

Raw text cannot be directly used by machine learning models. Therefore, multiple preprocessing steps are applied.

Pre-processing Steps:

- Converting text to lowercase
- Removing punctuation and special characters
- Removing stop words
- Tokenization
- Stemming using Snowball Stemmer

These steps reduce noise and improve learning efficiency.

7.1 Importance of Text Pre-processing

Text preprocessing improves model performance by removing irrelevant information. Without preprocessing, models may learn misleading patterns caused by noise.

For example, “Good” and “good” would be treated differently without case normalization, increasing dimensionality unnecessarily.

7.2 Text Normalization

All reviews are converted to lowercase to ensure uniformity.

Code Reference:

```
df["review"] = df["review"].str.lower()
```

7.3 Punctuation Removal

Punctuation symbols do not add semantic meaning and are removed from the text.

Code Reference:

```
def remove_punctuation(text):
    return text.translate(str.maketrans('', '', punc_to_remove))
```

7.4 Stop Word Removal

Common words such as “is”, “the”, and “and” are removed to focus on meaningful terms.

Code Reference:

```
def remove_stopwords(text):
    stop_words = set(stopwords.words('english'))
    word_tokens = word_tokenize(text)
    filtered_text = [word for word in word_tokens if word not in
stop_words]
    return " ".join(filtered_text)
```

7.5 Frequent and Rare Word Handling

Highly frequent and extremely rare words are analyzed to reduce noise and improve generalization.

7.6 Stemming

Snowball Stemmer is used to reduce words to their root form, improving feature consistency.

Code Reference:

```
stemmer = SnowballStemmer("english")

def preprocess_text(text):
    text = text.lower()
    text = re.sub(r'[^a-z\s]', ' ', text)
    words = text.split()
    words = [stemmer.stem(word) for word in words]
```

```
return " ".join(words)
```

8. Feature Extraction

8.1 TF-IDF Vectorization

TF-IDF converts text into numerical vectors by assigning higher importance to discriminative words while reducing the impact of commonly occurring terms.

📌 Code Reference:

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

8.2 Feature Selection and Dimensionality Reduction

TF-IDF inherently reduces dimensionality by down-weighting common words, making it suitable for large text datasets.

9. Exploratory Data Analysis (EDA)

EDA is performed to gain insights into the dataset.

Key Observations:

- Positive reviews are most frequent
- Negative reviews tend to be longer
- Certain words strongly correlate with sentiment
- Review length varies by sentiment category

9.1 Visualization Techniques Used

- Bar charts for sentiment distribution
- Box plots for review length analysis
- Frequency plots for common words

10. Model Building

Models Used:

1. Logistic Regression
2. Multinomial Naive Bayes

3. Linear Support Vector Machine (SVM)
4. Pretrained RoBERTa Model

10.1 Logistic Regression

A linear classifier suitable for high-dimensional sparse data.

10.2 Multinomial Naive Bayes

A probabilistic model particularly effective for text classification.

10.3 Linear Support Vector Machine

Maximizes the margin between sentiment classes and performs well with TF-IDF features.

10.4 Pretrained RoBERTa Model

A transformer-based deep learning model that captures contextual meaning effectively.

11. Model Evaluation

Evaluation Metrics:

- Accuracy
- Precision
- Recall
- F1-score
- Classification Report
-

11.1 Importance of Evaluation Metrics

Accuracy alone may not be sufficient for sentiment classification, especially when class imbalance exists. Precision measures the correctness of positive predictions, while recall evaluates the model's ability to identify all relevant instances. The F1-score provides a balance between precision and recall, making it a reliable metric for sentiment analysis tasks. Therefore, multiple evaluation metrics are used to ensure robust performance assessment.

11.2 Confusion Matrix Analysis

The confusion matrix helps analyze correct and incorrect sentiment predictions across classes.

12. Results and Discussion

Results indicate that **Linear SVM performs best among traditional ML models**, while the **RoBERTa model provides superior contextual understanding**. TF-IDF features significantly improve classification accuracy.

12.1 Performance Comparison

Traditional ML models offer efficiency and interpretability, while transformer models achieve higher accuracy at the cost of computation.

13. Applications of the Project

- Hotel reputation management
- Customer feedback analysis
- Service quality monitoring
- Travel recommendation systems

14. Limitations

- Limited to English reviews
- Neutral sentiment is harder to classify
- Pretrained model requires high computation

15. Future Scope

- Multilingual sentiment analysis
- Aspect-based sentiment analysis
- Real-time deployment
- Integration with hotel management systems

16. Software and Tools Used

- Python
- Jupyter Notebook
- Pandas, NumPy
- NLTK
- Scikit-learn
- Matplotlib
- Hugging Face Transformers

16.1 Streamlit Application

A Streamlit-based web interface was developed to demonstrate real-time sentiment prediction. The application allows users to input hotel reviews and instantly receive sentiment classification results. This interface enhances usability and demonstrates how the developed model can be deployed as an interactive application.

17. Key Contributions of the Project

- End-to-end sentiment analysis pipeline
- Comparison of ML and DL models
- Real-world hotel review analysis
- Practical business insights

18. Conclusion

This project successfully demonstrates the application of NLP and machine learning techniques for hotel review sentiment analysis. By combining effective preprocessing, TF-IDF feature extraction, and multiple classification models, the system accurately identifies customer sentiment and provides valuable insights for the hospitality industry.

19. References

1. Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis.
2. Devlin, J. et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers.
3. Liu, Y. et al. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach.

Student Details

Name: Sangamithra R

Email: hi.sangamithra@gmail.com

Phone: +91 7806959947

GitHub: <https://github.com/Sangamithra11>

Milestones

Milestone	Description
Define a problem	Identify sentiment analysis task
Business understanding	Understand customer feedback importance
Get the data	Collect hotel reviews dataset
Data preprocessing	Clean and normalize text
Feature creation	Apply TF-IDF
EDA	Analyze sentiment patterns
Model creation	Train ML & DL models
Model evaluation	Compare performance
Report writing	Document results
Project submission	Final delivery