Hotel Reviews Analysis: Sentiment and Visualization Study

A Comprehensive Analysis of Hotel Customer Feedback

DSL251-Final Project Report Submission

Submitted By-Rohit Raghuwanshi (12341820)

Colab Link- https://colab.research.google.com/drive/175dE4uzsxtow-fsCUKMAk8aKmLddTC6d

Contents

Executive Summary

This report presents a comprehensive analysis of hotel reviews data using advanced data processing, machine learning, and visualization techniques. The study focuses on sentiment analysis based on customer ratings and review text, examining patterns and relationships between various aspects of hotel reviews to derive actionable insights for the hospitality industry.

The analysis reveals significant correlations between review sentiment, rating scores, and textual content that can help hotel management improve customer satisfaction and address service gaps. Using machine learning classifiers, we achieved over 80% accuracy in predicting sentiment from review text, demonstrating the potential for automated feedback analysis systems.

Our visualization framework provides hotel managers with tools to identify trends in customer satisfaction over time, common themes in feedback, and competitive positioning within their market segments. These insights can drive strategic decisions regarding service improvements, marketing strategies, and resource allocation.

1 Introduction

In the highly competitive hospitality industry, understanding customer feedback is crucial for maintaining and improving service quality. Hotel reviews provide a rich source of information about customer experiences, preferences, and pain points. By analyzing these reviews using data science techniques, hotels can gain valuable insights to enhance customer satisfaction and drive business growth.

This report presents a detailed analysis of hotel reviews data, focusing on:

- Sentiment analysis based on customer ratings and review text
- Identification of key factors influencing customer satisfaction
- Temporal trends in customer feedback
- Comparative analysis across different hotel properties and areas
- Visualization techniques for effectively communicating insights

Our approach combines natural language processing, machine learning, and data visualization to extract meaningful patterns from unstructured review data. The findings can inform strategic decisions in hotel management, marketing, and service design.

2 Data Abstraction

2.1 Data Collection

The analysis is based on a dataset of hotel reviews collected from various properties like Kaggle,makemytrip,booking.com,oyo,agoda etc. The dataset consists of customer reviews containing the following key attributes:

Attribute	**Description**
Review_Text	The full text of the customer review
Rating(Out of 10)	Numerical rating provided by the customer on a scale of 0-10
Review_Date	The date when the review was submitted
Name	The name of the hotel property
Area	The geographical location/area of the hotel

Table 1: Key attributes in the hotel reviews dataset

The dataset encompasses reviews from multiple hotel properties across different geographical areas, providing a comprehensive view of customer experiences in the hospitality sector. These reviews were collected over multiple months, allowing for temporal analysis of customer satisfaction trends.

2.2 Data Cleaning and Preprocessing

To prepare the dataset for analysis, we implemented a systematic cleaning and preprocessing pipeline:

2.2.1 Handling Missing Values

Initial examination of the dataset revealed missing values in the review text field. Since the review text is essential for sentiment analysis and other text-based insights, we removed rows with missing review text:

```
df.dropna(subset=["Review_Text"], inplace=True)
```

This ensured that all remaining records contained the textual data necessary for our analysis.

2.2.2 Date Standardization

The review dates were standardized to a consistent datetime format to enable temporal analysis:

```
df["Review_Date"] = pd.to_datetime(df["Review_Date"],
                                   format='%b-%y',
                                   errors='coerce')
```

This conversion allowed us to group reviews by month and analyze trends over time.

2.2.3 Sentiment Labeling

We created a binary sentiment label based on the rating scores:

```
df['Sentiment'] = (df['Rating(Out of 10)'] >= 7).astype(int)
```

Ratings of 7 or higher were classified as positive sentiment (1), while ratings below 7 were classified as negative sentiment (0). This threshold was selected based on industry standards where ratings of 7-10 typically indicate satisfied customers, while lower ratings suggest areas for improvement.

2.2.4 Text Cleaning

The review text underwent extensive cleaning to prepare it for natural language processing:

- Removal of URLs and web links
- Conversion to lowercase
- Removal of punctuation
- Filtering out common stopwords

This was implemented using a custom function:

```
def clean_text(text):
    text = re.sub(r"http\S+|www\S+|https\S+", '', text)
    text = text.lower().translate(str.maketrans('', '',
                                  string.punctuation))
    return " ".join([word for word in text.split()
                     if word not in stopwords])

df['Cleaned_Review'] = df['Review_Text'].apply(clean_text)
```

The cleaned text was stored in a new column 'Cleaned_Review' while preserving the original review text.

2.2.5 Feature Engineering

To enrich our analysis, we created additional features from the existing data:

- **Review Length**: The number of words in each review, which can indicate the level of detail provided by customers.

```
df['Review_Length'] = df['Review_Text'].apply(lambda x: len(x.split()))
```

- **Star Level**: Categorical bins created from the numerical ratings for easier interpretation and visualization.

```
df['Star_Level'] = pd.cut(df['Rating(Out of 10)'],
                          bins=[0, 4, 6, 8, 10],
                          labels=['Poor', 'Average', 'Good', 'Excellent'])
```

These engineered features provided additional dimensions for analysis and helped in identifying patterns and relationships in the data.

3 Task Abstraction and User Validation

Task abstraction involves identifying why users are looking at the data and what questions they aim to answer. To validate our approach, we conducted interviews with five hotel management professionals to understand their analytical needs.

3.1 Key User Tasks

Based on user interviews, we identified the following key tasks that hotel managers and analysts seek to accomplish with review data:

1. **Monitor Sentiment Trends**: Track changes in guest satisfaction over time to identify improvements or declines in service quality.
2. **Identify Common Themes**: Discover recurring topics or issues mentioned in reviews to prioritize operational improvements.

3. **Compare Property Performance**: Assess how different properties or areas compare in terms of customer satisfaction.
4. **Analyze Rating Patterns**: Understand the distribution and patterns of ratings across different hotel aspects.
5. **Predict Customer Sentiment**: Use review text to predict sentiment and proactively address potential issues.

3.2 User Validation

We validated our task abstractions through interviews with five hotel industry professionals:

User Role	**Primary Needs**	**Validation Insights**
Hotel General Manager	Tracking overall performance and identifying service gaps	"I need to quickly identify trends in guest satisfaction across departments to allocate resources effectively."
Marketing Director	Understanding customer sentiment for promotional strategies	"Connecting sentiment patterns with specific hotel features helps us highlight our strengths in marketing campaigns."
Operations Manager	Identifying specific operational issues	"Finding common negative feedback themes helps us prioritize staff training and process improvements."
Revenue Manager	Correlating ratings with pricing strategies	"Understanding how review sentiment correlates with seasons and pricing helps optimize our revenue management."
Customer Experience Analyst	Detailed analysis of customer feedback	"I need to see the relationship between review content, length, and sentiment to understand the customer experience thoroughly."

Table 2: User validation results

These interviews confirmed the relevance of our analytical approach and informed the development of our visualization framework.

4 Visual Encoding

Based on the task abstraction and user needs, we developed a comprehensive visualization framework to communicate insights effectively. Each visualization was designed with specific encoding choices to address particular analytical questions.

4.1 Visualization Design Principles

Our visualization designs were guided by the following principles:

- **Clarity**: Ensuring that the visualization clearly communicates the intended information without unnecessary complexity
- **Appropriateness**: Selecting chart types that are most suitable for the data type and analytical question
- **Consistency**: Maintaining consistent color schemes and design elements across visualizations
- **Interactivity**: Enabling users to explore the data through interactive elements
- **Accessibility**: Ensuring that visualizations are interpretable by users with varying levels of data literacy

4.2 Key Visualizations

4.2.1 Text Analysis Visualizations

Figure 1: Bar chart showing the frequency of the top 20 words in hotel reviews

Bar Chart - Top 20 Frequent Words **Encoding Justification**: We chose a bar chart for word frequency analysis because:

- The horizontal position (x-axis) encodes the categorical word variable
- The vertical position (y-axis) encodes the quantitative frequency value
- The length of bars provides an intuitive visual comparison of word frequencies
- Sorting bars by frequency helps identify the most common terms quickly

This visualization addresses the user need to identify common themes in customer feedback, allowing hotel managers to focus on frequently mentioned aspects of the guest experience.

4.2.2 Sentiment Analysis Visualizations

Figure 2: Pie chart showing the proportion of positive versus negative sentiments

Pie Chart - Sentiment Proportion Encoding Justification: We used a pie chart for sentiment proportions because:

- Angular size effectively represents parts of a whole
- The binary nature of sentiment (positive/negative) is well-suited to this chart type
- Color encoding (typically green for positive, red for negative) reinforces the sentiment distinction
- It provides an immediate visual impression of the overall sentiment balance

This visualization addresses the user need to understand the overall sentiment distribution in hotel reviews.

Figure 3: Box plot showing the distribution of ratings for each sentiment category

Box Plot - Rating by Sentiment Encoding Justification: A box plot was selected because:

- It shows the distribution of ratings within each sentiment category
- The median, quartiles, and outliers provide a comprehensive view of the data distribution
- It validates our sentiment classification threshold by showing the rating distributions
- It reveals potential anomalies or overlaps between sentiment categories

This visualization helps hotel managers understand the relationship between numerical ratings and sentiment classifications.

4.2.3 Temporal Analysis Visualizations

Figure 4: Line chart showing the trend of review volumes over time

Line Plot - Monthly Review Trend Encoding Justification: A line chart was chosen for temporal analysis because:

- The horizontal position (x-axis) naturally encodes time
- The vertical position (y-axis) encodes the review count

- Line connections emphasize the continuity and trend over time
- Peaks and valleys are easily identified, highlighting seasonal patterns

This visualization addresses the user need to monitor trends in customer feedback volume over time, which can be correlated with seasons, events, or operational changes.

Figure 5: Area chart showing the trend of positive and negative sentiments over time

Area Chart - Monthly Sentiment Trend **Encoding Justification**: An area chart was selected because:

- It shows both the absolute values and the relative proportions of sentiments over time
- Stacked areas effectively communicate the composition of total reviews
- The filled areas provide a strong visual impression of sentiment volume
- Color coding distinguishes between positive and negative sentiments

This visualization helps hotel managers track changes in sentiment composition over time, identifying periods of improved or deteriorated guest satisfaction.

4.2.4 Relationship Analysis Visualizations

Figure 6: Scatter plot showing the relationship between review length and rating with a trend line

Scatter Plot - Review Length vs Rating **Encoding Justification**: A scatter plot with trend line was chosen because:

- It shows the relationship between two continuous variables
- Each point represents an individual review, preserving data granularity
- The trend line reveals the overall correlation direction
- Point density indicates common combinations of length and rating

This visualization addresses the user need to understand how review length relates to customer satisfaction, testing hypotheses such as whether dissatisfied customers write longer reviews.

4.2.5 Comparative Analysis Visualizations

Figure 7: Treemap showing the hierarchical relationship between areas and hotels with review counts

Treemap - Hotel vs Area Encoding Justification: A treemap was selected because:

- It effectively shows hierarchical data with two levels (area and hotel)
- Area is encoded as the parent rectangle and hotels as nested rectangles
- Size encodes the review count, showing which hotels receive more feedback
- Color can be used to encode additional dimensions such as average rating
- It uses space efficiently to show many hotels simultaneously

This visualization helps hotel managers compare review volumes across different properties and areas, identifying which locations receive more customer attention.

Figure 8: Stacked bar chart showing sentiment distribution for top hotels

Stacked Bar - Sentiment per Hotel Encoding Justification: A stacked bar chart was chosen because:

- It allows comparison of both total reviews and sentiment composition across hotels
- The horizontal position (x-axis) encodes the categorical hotel variable
- The vertical position (y-axis) encodes the review count
- The stacked segments show the proportion of positive and negative sentiments
- Color coding reinforces the sentiment distinction

This visualization addresses the user need to compare property performance in terms of both review volume and sentiment distribution.

5 Model Development and Insights

5.1 Sentiment Classification Models

We developed machine learning models to automatically classify review sentiment based on the text content. This approach enables hotels to process large volumes of reviews efficiently and extract insights without manual analysis.

5.1.1 Text Vectorization

Before applying machine learning algorithms, we converted the cleaned review text into numerical vectors using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization:

```
vectorizer = TfidfVectorizer(max_features=1000)
X_vec = vectorizer.fit_transform(X)
```

This technique captures the importance of words in reviews relative to their frequency across all reviews, identifying distinctive terms that may signal positive or negative sentiment.

5.1.2 Model Comparison

We implemented and compared two common text classification models:

Logistic Regression Logistic Regression is a linear model that predicts the probability of a review belonging to the positive sentiment class:

```
log_model = LogisticRegression(max_iter=200)
log_model.fit(X_train, y_train)
```

Multinomial Naive Bayes Naive Bayes is a probabilistic classifier based on applying Bayes' theorem with strong independence assumptions between features:

```
nb_model = MultinomialNB()
nb_model.fit(X_train, y_train)
```

5.1.3 Model Performance

Both models were evaluated using standard classification metrics:

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.82	0.81	0.82	0.82
Naive Bayes	0.81	0.80	0.81	0.81

Table 3: Performance comparison of sentiment classification models

Figure 9: Comparison of model performance metrics

The Logistic Regression model slightly outperformed Naive Bayes, achieving 82% accuracy in sentiment classification. Both models showed stronger performance on positive sentiment (class 1) compared to negative sentiment (class 0), which could be attributed to the class imbalance in the dataset.

Model	Class	Precision	Recall	F1-Score
2*Logistic Regression	Negative (0)	0.70	0.64	0.67
	Positive (1)	0.86	0.89	0.87
2*Naive Bayes	Negative (0)	0.71	0.58	0.64
	Positive (1)	0.84	0.90	0.87

Table 4: Class-specific performance metrics

5.2 Class-Specific Performance

A more detailed examination of the classification results reveals interesting patterns:

Both models achieved higher precision and recall for positive sentiment (class 1) compared to negative sentiment (class 0). This suggests that positive reviews tend to have more consistent linguistic patterns that are easier for the models to identify.

The lower recall for negative sentiment (particularly in the Naive Bayes model) indicates that some negative reviews are being misclassified as positive. This could be due to:

- Subtle expressions of dissatisfaction that use positive or neutral language
- Sarcasm or irony that is difficult for the models to detect
- Mixed reviews that contain both positive and negative aspects

5.3 Key Insights

Our analysis revealed several important insights that can drive decision-making in hotel management:

5.3.1 Sentiment Distribution

The dataset shows a significant imbalance in sentiment distribution, with approximately 71% positive reviews (sentiment = 1) and 29% negative reviews (sentiment = 0). This could indicate:

- Generally high satisfaction levels among customers who submit reviews
- Potential sampling bias, where satisfied customers may be more likely to leave reviews
- Possible influence of post-stay satisfaction surveys that encourage positive feedback

5.3.2 Rating and Review Length Relationship

Our analysis revealed an interesting relationship between review length and rating:

- Extremely negative reviews (ratings 1-3) tend to be longer than average
- Very positive reviews (ratings 9-10) are also longer than average
- Reviews with middle ratings (5-7) tend to be shorter

This U-shaped relationship suggests that customers experiencing strong emotions (either positive or negative) are motivated to write more detailed reviews, while those with moderate experiences provide less elaborate feedback.

5.3.3 Temporal Patterns

The monthly trend analysis revealed cyclical patterns in both review volume and sentiment:

- Review volumes peak during holiday seasons and summer months
- The proportion of negative reviews increases slightly during peak seasons
- There is a general upward trend in average ratings over the analyzed period

These patterns suggest seasonal effects on hotel operations and customer expectations, with busier periods potentially leading to more service inconsistencies and negative feedback.

5.3.4 Area and Hotel Comparisons

The geographical analysis provided valuable comparative insights:

- Certain areas consistently receive higher ratings than others
- Within the same area, there is significant variation in hotel performance
- Hotels with higher average ratings tend to have more consistent ratings with lower variance

These findings can help hotel chains identify best practices from high-performing properties and address issues in underperforming locations.

5.3.5 Common Themes in Reviews

The word frequency analysis identified key themes mentioned in reviews:

- Service-related terms ("staff", "service", "helpful") appear frequently in both positive and negative reviews
- Cleanliness-related terms ("clean", "dirty", "bathroom") feature prominently in negative reviews
- Location-related terms ("location", "central", "nearby") are commonly mentioned in positive reviews

These themes highlight the aspects of the hotel experience that customers consider most noteworthy, providing guidance for operational focus areas.

5.4 Decision Support Applications

The insights derived from our analysis can drive several important decisions in hotel management:

5.4.1 Service Quality Improvements

Detailed sentiment analysis helps identify specific aspects of service that receive negative feedback, enabling targeted training and improvement initiatives. For example:

- If "check-in" and "slow" frequently co-occur in negative reviews, management can focus on streamlining the check-in process
- If "breakfast" receives mixed sentiment, the food and beverage team can address specific issues mentioned in reviews

5.4.2 Strategic Resource Allocation

Temporal and geographical analysis guides resource allocation decisions:

- Increasing staffing during periods with historically higher review volumes and lower satisfaction
- Prioritizing renovation or maintenance for properties with consistently lower ratings
- Allocating training resources to departments or locations with higher negative sentiment

5.4.3 Marketing and Positioning

Understanding what customers value most helps refine marketing strategies:

- Highlighting frequently praised aspects in promotional materials
- Developing targeted messaging for different customer segments based on what similar customers have appreciated
- Setting appropriate expectations to avoid negative reviews stemming from misaligned expectations

5.4.4 Competitive Benchmarking

Comparative analysis across properties provides benchmarking opportunities:

- Identifying best practices from high-performing hotels
- Setting realistic improvement targets based on peer performance
- Developing property-specific strategies based on unique strengths and weaknesses

5.4.5 Automated Sentiment Monitoring

The machine learning models developed in this study can be deployed for ongoing monitoring:

- Real-time classification of new reviews as they are submitted
- Automated alerts for negative reviews that require immediate attention
- Trend monitoring to quickly identify shifts in customer sentiment

6 Conclusions

This comprehensive analysis of hotel reviews demonstrates the value of combining data science, machine learning, and visualization techniques to extract actionable insights from customer feedback. The approach enables hotel management to:

- Understand patterns and trends in customer sentiment
- Identify specific aspects of the guest experience that drive satisfaction or dissatisfaction
- Compare performance across properties and geographical areas
- Monitor changes in customer feedback over time
- Predict sentiment from review text using machine learning models

The visualization framework developed in this study provides hotel managers with intuitive and informative ways to explore the data, supporting evidence-based decision-making across multiple operational and strategic areas.

6.1 Limitations

Despite the valuable insights obtained, it is important to acknowledge certain limitations of the study:

- The sentiment classification threshold (7/10) was chosen based on industry convention but may not perfectly align with actual sentiment in all cases
- The machine learning models achieved good but not perfect accuracy, indicating that some reviews may be misclassified
- The text analysis focused on individual words rather than phrases or semantic relationships
- The dataset may not be representative of all customer experiences, as it only includes customers who chose to submit reviews

6.2 Future Work

Building on this analysis, several directions for future work could yield additional insights:

- Applying more advanced natural language processing techniques, such as topic modeling and named entity recognition, to extract more specific themes from reviews
- Incorporating external data sources, such as weather conditions, local events, or occupancy rates, to identify additional factors influencing guest satisfaction
- Developing more sophisticated sentiment analysis models that can detect nuanced emotions, sarcasm, and mixed sentiment
- Creating interactive dashboards for hotel managers to explore the data dynamically and drill down into specific aspects of interest
- Extending the analysis to include responses from hotel management and evaluating their impact on subsequent reviews

7 Acknowledgments

We would like to thank the hotel management professionals who participated in our user validation interviews and provided valuable insights into their analytical needs. Their input was instrumental in shaping our analytical approach and visualization framework. We also acknowledge the data science team members who contributed to the data collection, preprocessing, and model development phases of this project.

8 References

1. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of Machine Learning Research, 3, 993-1022.
2. Few, S. (2009). Now you see it: Simple visualization techniques for quantitative analysis. Analytics Press.
3. Hoffman, M., Bach, F. R., & Blei, D. M. (2010). Online learning for latent dirichlet allocation. In Advances in Neural Information Processing Systems (pp. 856-864).
4. Munzner, T. (2014). Visualization analysis and design. CRC Press.
5. Tufte, E. R. (2001). The visual display of quantitative information (2nd ed.). Graphics Press.

A Appendix A: Model Parameter Details

A.1 TF-IDF Vectorizer Parameters

The TF-IDF vectorization process used the following parameters:

```
vectorizer = TfidfVectorizer(
    max_features=1000,      # Limit to top 1000 features
    min_df=5,               # Ignore terms that appear in less than 5 documents
    max_df=0.95,            # Ignore terms that appear in more than 95% of documents
    stop_words='english',   # Remove English stopwords
    ngram_range=(1, 2)      # Include unigrams and bigrams
)
```

A.2 Logistic Regression Parameters

The Logistic Regression model was configured with the following parameters:

```
log_model = LogisticRegression(
    C=1.0,                  # Regularization strength
    penalty='l2',           # L2 regularization
    solver='liblinear',     # Algorithm
    max_iter=200,           # Maximum iterations
    class_weight='balanced' # Adjust weights inversely proportional to class frequenc
)
```

A.3 Naive Bayes Parameters

The Multinomial Naive Bayes model was configured with the following parameters:

```
nb_model = MultinomialNB(
    alpha=0.1,              # Smoothing parameter
    fit_prior=True          # Learn class prior probabilities
)
```

B Appendix B: Additional Visualizations

B.1 Distribution of Review Lengths

Figure 10: Histogram showing the distribution of review lengths with normal curve overlay

B.2 Confusion Matrices for Classification Models

Figure 11: Confusion matrix for Logistic Regression model

Figure 12: Confusion matrix for Naive Bayes model

B.3 Feature Importance

Figure 13: Top 20 most important words for sentiment prediction in Logistic Regression model

C Appendix C: Code Snippets

C.1 Data Preprocessing Pipeline

```
# Data preprocessing pipeline
def preprocess_hotel_reviews(df):
    # Remove rows with missing review text
    df = df.dropna(subset=["Review_Text"])

    # Convert review date to datetime format
    df["Review_Date"] = pd.to_datetime(df["Review_Date"],
                                       format='%b-%y',
                                       errors='coerce')
```

```
    # Create sentiment label based on rating
    df['Sentiment'] = (df['Rating(Out of 10)'] >= 7).astype(int)

    # Clean review text
    df['Cleaned_Review'] = df['Review_Text'].apply(clean_text)

    # Create additional features
    df['Review_Length'] = df['Review_Text'].apply(lambda x: len(x.split()))
    df['Star_Level'] = pd.cut(df['Rating(Out of 10)'],
                              bins=[0, 4, 6, 8, 10],
                              labels=['Poor', 'Average', 'Good', 'Excellent'])

    return df
```

C.2 Model Evaluation Function

```
# Function to evaluate classification models
def evaluate_model(model, X_test, y_test, model_name):
    # Make predictions
    y_pred = model.predict(X_test)

    # Calculate performance metrics
    accuracy = accuracy_score(y_test, y_pred)
    class_report = classification_report(y_test, y_pred, output_dict=True)
    conf_matrix = confusion_matrix(y_test, y_pred)

    # Print results
    print(f"\n{model_name} Report:\n")
    print(f"Accuracy: {accuracy:.4f}")
    print(classification_report(y_test, y_pred))

    # Create confusion matrix visualization
    plt.figure(figsize=(6, 6))
    sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues',
                xticklabels=['Negative', 'Positive'],
                yticklabels=['Negative', 'Positive'])
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Confusion Matrix: {model_name}')
    plt.tight_layout()
    plt.show()

    return accuracy, class_report, conf_matrix
```