## Hotel Reviews Analysis

## 1. Introduction

Reviews and ratings have revolutionized traditional word-of-mouth marketing into a viral form of feedback that can influence consumers’ opinions. When we consider reviews and ratings, both business owners and consumers can benefit. On one hand, reviews help consumers voice their requirement and act as a form of recommendation, a high-value social proof.  On the other hand, reviews allow business owners to identify potential areas of improvement, measure overall customer satisfaction, and create micro-marketing campaigns.

With the constantly growing volume and variety of review data, effective analysis becomes of paramount importance. Our project is based on reviews and rating analysis with a case study of hotel reviews for hotels in the US. We seek to provide insights to both the customers and the hotel owners. We used data visualization and text mining techniques such as word clouds, NRC Sentiment Analysis, and Vader Analysis to explore the patterns and features from the dataset. Next, through feature engineering, we augmented roughly 28,000+ features from the raw review texts, which were later used as inputs to the predictive models. Finally, we built, tuned, and compared the performances of various machine learning models such as Logistic Regression, Decision Tree, Random Forest, and Bagging to categorize hotels based on the review texts.

## 2. Data Cleaning and Preprocessing

#### 2.1 Data Cleaning

As we had three different datasets from Kaggle, we first concatenated the datasets, keeping only the common columns. Doing this left us with information on the hotels, such as name, address, categories, coordinates, city, and country, and information on the reviews, such as date, numeric rating, reviews text and titles, as well as information on the reviewers, such as username, user’s city, and user’s province. As our analysis will focus mostly on the ratings and reviews, we dropped the observations with missing ratings or missing review texts. Additionally, we converted the review titles whose value was missing into an empty string and concatenate all review titles to review texts to create a review column.

Taking a deeper look at the review column, we noticed that some reviews were not meaningful. These appear to be a placeholder for the ratings with no review texts, and as such, we removed these reviews. In addition, we noticed that some reviews were not legible. These either contain non-ASCII characters or foreign-language texts. We employed Google Translator to convert these reviews into English texts. The illustrations of what we did are provided in Figure 1.

#### 2.2. Data Processing

Diving deeper into reviews rating, we noted that the obvious rating scale is from 1 to 5. However, we also noticed that there were some observations with a rating of 0 or rating greater than 5, with a maximum of 10, as shown in Figure 2. There were not many instances like these, with 0.61% and 0.65% of total observations having ratings of 0 and ratings greater than 5, respectively. Instead of immediately removing these observations, we first determined whether there exists another rating scale of 0-to-10 in our dataset. If a hotel employed a 0-to-10 rating scale, a rating of 5 would not be considered a good rating, but rather a neutral rating. By simply removing ratings greater than 5, we would mistakenly label the not-so-good reviews as good reviews, thereby impacting our model performance. Upon further investigation, we found that there are 253 hotels with ratings of 0 and 5 hotels with ratings greater than 5. Among all observations, there are 9.6% and 1.4% of total observations associated with these 253 hotels and 5 hotels, respectively. This meant that there were hotels with a 0-to-10 rating scale. As such, we rescaled these ratings from a 0-to-10 scale to a 1-to-5 scale. Figure 3 shows the rating distribution after all ratings were scaled to the 1-to-5 scale.

We also investigated the hotels’ locations and plotted the coordinates of the hotels. As shown in Figure 4, there are hotels that are located outside of the US. In fact, 14.7% of total reviews are for hotels outside of the US. As we wanted to focus our analysis on hotels in the US, we excluded these reviews for the purpose of this analysis. The result of this is shown in Figure 5a. From the coordinates of the hotels, we also extracted the hotels’ state using reverse geocoding and created a heatmap of the reviews, as shown in Figure 5b. It can be seen from both Figures 5a and 5b that the distribution of hotels’ locations is reasonable given the geographical location of the US.

## 3. Exploratory Data Analysis

#### 3.1 Rating EDA

The first question we wanted to answer was “What are the best and worst hotels in the country?” As we have about 3,000 hotels with around 55,000 reviews, we grouped the reviews by hotel name and plotted the average ratings by the hotel in Figure 6. We can see from Figure 6 that there are a lot of hotels with an average rating of 1 or 5, the two extremes. We also noticed that these hotels with an average rating of 1 or 5 only have one review each. As such, the “average” ratings are in fact calculated from a single review. This aligns well with what we would expect out of common behaviors from travelers: one would leave a review if he or she is particularly dissatisfied or satisfied with the service he or she received, explaining why the many hotels with an extreme average rating only have one review. To account for this, in Figure 7, we plotted average ratings by the hotel, including only hotels with more than 5 reviews. Not surprisingly, the number of hotels with an average rating of 1 or 5 decreased significantly. In contrast, most hotels had an average rating ranging from 3.75 to 4.5. However, as we focus more on reviews and ratings at the individual level, rather than aggregated by hotels, at this stage, we do not remove the hotels with less than 5 reviews.

As aforementioned, we noted that there were more good reviews than bad reviews. Figure 8, in which we plotted the hotel reviews by rating categories, confirms this, showing that the “Very good” reviews, reviews whose ratings round to 5, are the most common ones, followed by the “Moderately good” reviews, those whose ratings round to 4. Broken down by month, we can see the obvious seasonality in traveling, with a spike in the summer months, especially in July, as can be seen in Figure 9. Further broken down by month and rating category, we see the trend once again that travelers tend to leave more extreme reviews, with the number of “Very bad” and “Very good” reviews increasing in the summer months as well. This makes sense, considering that people tend to travel more in the summer thanks to school breaks, and that weather is generally warmer and nicer in the summer, leading to more hotel usage. Knowing this could help hotels owners focus their resources on the more popular season to increase their profits.

#### 3.2 Review EDA

Other than seasonality, to increase their profits, hotel owners may also be interested in knowing what their customers care about the most in their properties. After removing the common words like “hotels,” “stays,” and “room,” as well as the short words like “I” and “u,” we generated word clouds from all reviews. Looking at the word cloud in Figure 10, we can see that many reviews mention “clean” and “staff,” along with “friendly” and “comfortable,” implying that travelers’ tend to pay more attention to the cleanliness of the property, the friendliness of the staff, and the comfort of their overall experience. We were able to verify this by looking closely at the word clouds for reviews with ratings of 5 in comparison to reviews with ratings of 1. As shown in Figure 11, we can see that the good reviews mentioned a lot of "clean" and "staff." There are also a lot of mentions of "friendly," "helpful," "comfortable," "convenient," "location," which all hint that the good hotels are the ones that are clean and comfortable, with friendly and helpful staff, and at a convenient location. On the other hand, the bad reviews are the ones about hotels that are "dirty," "booked," or even "smelled." With this information, hotel owners could tailor their hotels’ features and focus on keeping their hotels clean and training their staff to gain a better reputation.

Taking the word clouds one step further, we considered reviews for the best and worst hotels, as indicated by the average rating by hotels discussed above. To ensure we have long enough texts for each hotel, we considered the top three and bottom three hotels with more than 5 reviews. Figure 12 tells a similar story as far as customers’ attention. However, we can see that the Litchfield Inn, while being one of the worst hotels, had many good words, such as “great,” “location,” “beach,” and “oceanfront.” This emphasizes a drawback of word clouds, being that they only show words and not the context of the words. Thus, we wanted to examanine reviews summarization to better understand what the customers care about.

As we see in the word clouds, review ratings may not relate to the actual review context. A low rating may be given to a hotel with positive reviews. This problem may be collocated with the problem of advertising, where the hotel will put up their best description and imaging for their properties while the reality is otherwise. We decided to do review summarization to enable customers to get opinions from peers who have had first-hand experience.  We first put together all hotel reviews for each hotel. Then using gensim.summarization, we built a function “get\_summarization” which if given a hotel name and number of word\_count will summarize reviews from the hotel as illustrated in Figure 13. This way the feedback is more accurate.

We looked to understand the focus of the reviews to get a sense of what people are talking about. One way could be to read through each review, which is impossible given the number of reviews. Another way would be to use review summarization as shown above, but with increasing data, it becomes difficult to keep track of what you are looking for. We decided to do topic modeling on the reviews using Latent Dirichlet Allocation (LDA). We extracted the review text and title data and use LDA to classify the reviews into topics. With several trials for two to eight topics, we picked three topics as the optimal number of topics. We visualized these topics using word cloud Figure 14 and pyLDAvis tool to draw powerful insights from the results as illustrated in Figure 15.  We see the first topic focused on service where great importance is given to “staff,” “time,” “service,” and “appreciation”; this reflects on customer service provided by the hotels. Another topic is centered around “room,” “bed,” “bathroom,” and “water,” all of which speak to the property itself. Finally, we see a focus on “location,” “area,” “convenient, “place,” and “comfortable,” which relate to the frequent travelers such as professionals and bring an emphasis on location and experience. These results can be used to provide actionable insights for hotel owners.

#### 3.3 Vader Analysis

Vader sentiment analyzer was applied to the review texts and titles in this dataset. Vader is a rule- and lexicon-based framework for sentiment analysis, with support for intensity estimation. Each review and review title can be classified as having an overall negative, neutral, or positive sentiment polarity based on its compound sentiment score. This produces 4 Vader sentiment score features. We perform the analysis in 3 columns, text, title, and the text and title combined. We obtained 3 compound scores, named review polarity, title polarity, and review and title polarity.

We grouped the dataset by hotel name, counted the number of rating observations, and averaged the compound scores across different observations. We categorized the hotels as either Terrible, Bad, Neutral, Good, or Great based on the following threshold values.

|  |  |
| --- | --- |
| Terrible | review polarity < -0.4 |
| Bad | -0.4 <= review polarity < -0.1 |
| Neutral | -0.1 <= review polarity < 0.1 |
| Good | 0.1 <= review polarity < 0.5 |
| Great | review polarity >= 0.5 |

We plotted the distribution of hotel review categories, where it is heavily skewed towards Great and Good. Therefore, we performed sentiment analysis on a selected part of our dataset, where we focused on the hotels in the category of Good and with more than 10 reviews. We defined popularity as a hotel’s average rating multiplied by the total number of reviews and mapped the 10 most popular hotels.

## 4. Feature Engineering

To extract more features about the reviews and integrate those features into our predictive models, we applied text mining techniques on the translated, preprocessed review texts. Reviews convey important information about the hotel location, customer service quality, room service quality, and other particular reasons why a certain hotel is rated as *Great* or *Terrible*.

From the texts, we were able to generate two kinds of features: meta-features and text features. Meta features are related to only the structure of the review texts. Intuitively, a long review with a rich vocabulary provides more details and insights than a short review with a small vocabulary.

Text features are concerned about the content of the reviews. Through Vader Analysis, we calculated statistics on how positive, negative, and neutral a review is and combined these results to give a compound sentiment (higher = more positive) for the review. In addition, emotion analysis attributes 8 emotional scores based on NRC data. Finally, the Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer converted a collection of raw review texts to a matrix of TF-IDF features -- the overall document weightage of every single word in the review text corpus.

|  |  |
| --- | --- |
| **Meta Features (structure of the text)** | **Text Features (the content of text)** |
| * Sentence length (characters & words) * Word length * Percentage of unique words * Stopword count | * Vader Analysis: positivity/negativity * NRC Emotions: Scores for 8 emotions * TF-IDF: TF-IDF Vectorizer |

Through the transformations mentioned above, we were able to augment the data set with 28,000+ additional features related to the review text itself.

## 5. Model Development and Outcomes Analysis

During the model development process, four different learning problems were explored. Namely, regression, five-class classification, three-class classification, and binary classification. For each approach, four classification algorithms were evaluated: decision tree, random forest, bagging classifier, and logistic regression. Model features were developed based on term frequency inverse document frequency (TF-IDF). Furthermore, the data for each set of models was randomly partitioned into a 80:20 train-test split.

As previously discussed, there is a severe class imbalance with respect to the ratings whereby a large majority of the population corresponds to ratings greater than or equal to 4. To treat this, the data set was augmented by oversampling the minority rating classes using an approach referred to as Synthetic Minority Oversampling Technique (SMOTE). This technique does just what the name implies, oversamples the minority classes by generating new observations from the existing ones. SMOTE was run on the training set for each set of models. Model performance before and after data augmentation using SMOTE was evaluated. In each case, model performance declined on the oversampled dataset, though only marginally. In the following subsections, we summarize and opine on the performance of each set of models. Please note, only the performance after data augmentation was conducted is discussed; please refer to the attached Python notebook for a more granular overview of model development.

#### 5.1 Vader Analysis Considerations for Model Development

#### 5.2 Regression

Several regression techniques were developed on a combination of both review text and review title text, where the dependent variable was defined as the continuous rating in the raw dataset.

In all, six regression models were developed, with the support vector machine model exhibiting the strongest performance, as measured by R2 and mean-square error (MSE). However, with a R2 of 0.572 and an MSE of 0.704, the SVM model is still considered ...weak performer, so we wanted to explore classification to see if we could do any better.

#### 5.3 Classification

#### 5.3.1 Five-Class Classification

An additional column, entitled ‘five\_class\_target’, was added to the working data frame which grouped the ratings into one of five categories. In particular, the ratings from the raw dataset were rounded to the nearest integer. Four different classification models were subsequently trained on the TF-IDF features with the new five\_class\_target column set as the dependent variable.

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| Decision Tree | 0.40 |
| Random Forest | 0.52 |
| Bagging Classifier | 0.45 |
| Logistic Regression | 0.51 |

Based on these results, the random forest model exhibits the strongest performance, as measured by accuracy. However, an accuracy score of 0.52 isn’t generally considered strong performance.

#### 5.3.2 Three-Class Classification

For three-class classification, the ratings were partitioned into three groups: Bad, Neutral, and Good which were encoded by 0, 1, and 2, respectively. The accuracy of each model is displayed in the table below.

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| Decision Tree | 0.68 |
| Random Forest | 0.75 |
| Bagging Classifier | 0.74 |
| Logistic Regression | 0.78 |

Model performance is much stronger for three classes, which was expected. In this case, logistic regression exhibits the strongest performance while performance of the other classifiers is also strong. Despite the increase in performance, we were interested if we could do any better, so we generalized the reviews even further, into two classes.

#### 5.3.3 Binary Classification

Four binary classifiers were trained on review text such that ratings greater than or equal to 3 were considered good and everything else was considered bad (or denoted by a zero). Binary classification performance was much stronger, which is no surprise as there is a large degree in term frequency between ratings above 3 and those below 3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Decision Tree** | **Random Forest** | **Bagging Classifier** | **Logistic Regression** |
| Precision | 0.82 | 0.83 | 0.84 | 0.86 |
| Recall/TPR | 0.82 | 0.94 | 0.88 | 0.92 |
| F1 Score | 0.82 | 0.88 | 0.86 | 0.89 |
| Accuracy | 0.75 | 0.83 | 0.80 | 0.85 |
| AUROC | 0.72 | 0.88 | 0.83 | 0.90 |

Again, logistic regression outperforms the decision tree, random forest, and bagging classifiers, as measured by accuracy, precision, recall, and the area under the ROC curve.

Because of the apparent disparity between the compound polarity scores between the review text and the corresponding title text, binary classifiers were also trained on the title text. As evidenced by the results in the table below, the classifiers trained on the title text features perform at a similar level to that of classifiers trained on the review text, with logistic regression again exhibiting the strongest performance.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Decision Tree** | **Random Forest** | **Bagging Classifier** | **Logistic Regression** |
| Precision | 0.83 | 0.83 | 0.83 | 0.80 |
| Recall/TPR | 0.83 | 0.88 | 0.86 | 0.93 |
| F1 Score | 0.83 | 0.85 | 0.84 | 0.86 |
| Accuracy | 0.77 | 0.80 | 0.79 | 0.80 |
| AUROC | 0.76 | 0.83 | 0.81 | 0.84 |

## 6. Conclusion and Future Improvements

#### 6.1. Future Development

1. Further, develop review summarization for hotel recommendations taking into account similarity between reviewers. The goal is to build an app or extend functionalities of available apps to give user recommendations and real experience in a click.
2. Optimize speed of some data processing functions
3. Use additional variables for example the Positive & Negative Review Attributes offered by google as illustrated in Figure 17. Instead of using only ratings and comments/reviews, these additional attributes will improve the performance and accuracy of the insights and recommendation
4. Explore other techniques such as In Neural Networks, Convolutional Neural Networks, LSTMs, and more recently Transformers for sentiment analysis and topic detection.