

Analysis Hotel Reviews in The U.S.

Cristina Giraldo

Renzo Castagnino

Siqi jiang

George Washington University

Table of Contents

Chapter 1: Introduction	3
Chapter 2: Data	4
Summary of dataset.....	4
Smart Question.....	5
Chapter 3: Methods.....	5
Preprocessing	5
Data Mining / Text Mining	7
Chapter 4: Results	13
Chapter 5: Summary and Conclusions.....	15
References.....	17

Chapter 1: Introduction

The human being is dominated by feelings and these feelings are what generally lead everyone to make decisions and create opinions about people, things, experiences, and services. Therefore, when people want to plan a vacation or a trip, it is normal that we are interested in knowing the place, but it is also part of our interest to know where we are going to sleep. The best way to know about the lodging is knowing how good the reviews of the place are. However, commentaries sometimes mismatch ratings leading to bad decisions. The main purpose of this project is to predict the rate of the review according to the reviews made by the customers. In order to do this, data mining techniques will be applied to generate a model that will help us to predict the rating according to the reviews.

Chapter 2: Data

Summary of dataset

The data for this report is a collection of hotel reviews in different cities located in The U.S. the information was collected from 2016 to 2018 sourced from data.world. The dataset lists ten thousand observations and includes information on date added, date updated, address, categories, primary categories, city, country, keys, latitude, longitude, name, postal code, province, reviews date, reviews date seen, reviews rating, reviews source URLs, reviews text, reviews title, reviews user city, reviews user province, reviews username, source URLs, websites and location. Categories refers to the type of hotel, primary categories indicate accommodation or arts and recreation. Location, longitude, postal code and city are the location of the hotel.

Smart Question

Does the customers' reviews can be useful to predict the rating of the hotels? We can answer that question by looking and analyzing at the date set provided by data.world. The purpose of this analysis is to assess the different reviews and provide suggestions to have better services for future customers.

Chapter 3: Methods

The analysis conducted in this report is based on the comments written by travelers according to their experiences in the hotels. It is worth to remember that the dataset has a sample between the year 2016 and 2018 with observations of ten thousand records of different hotels in cities of the United States of America.

To respond to the SMART questions some analysis was carried out on the dataset which included the application of statistical methods for preprocessing and data mining. In this case, we coped with data cleaning to remove noise, data transformation, feature selection to reduce dimensionality and remove redundant columns. For data mining we made use of statistical methods (Naïve Bayes, support vector machine and lasso regression) to predict the review rating. For the analysis and preprocessing we made use of PyCharm and Plotly Dash.

Preprocessing

It is an important step for data mining which basically consist and delete unnecessary information, standardize data, reduce features (if it is necessary) and cope with missing values. This important step is done to avoid overfitting our models and to have better and accurate

results as much as possible. In this project we used some techniques to clean our data as much as possible. These techniques are mentioned below.

Cleaning Data

For this process we implemented feature selection in which we dropped irrelevant features from our dataset: (*'date updated', 'address', 'categories', 'keys', 'date added', 'reviews date seen', 'reviews source URLs', 'websites', 'location', 'reviews username'.*) These features or columns will not be use in our data mining analysis.

A next step in the cleaning process was also removing noise by converting all the words in the column “reviews text” to lowercase, in ordering to avoids having copies of the same works. We also made sure that all the content had a string format. In addition, we removed the punctuation (including that we found the sign “@”), since it doesn’t add any extra information while analyzing the text data.

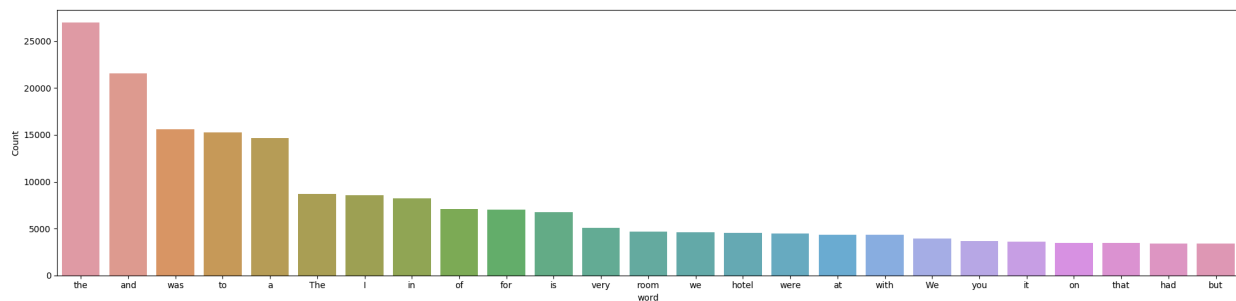
Missing Values.

In this step, we used the python command `df.isnull().values.any()` to identify which values were missing. For text mining we made use of instruction `df["reviews_text"].isna().sum()` to verify if the column “reviews text” had any missing values. When missing values were acknowledged they were removed using the following command `df['reviews_text'] = df['reviews_text'].dropna().reset_index(drop=True).`

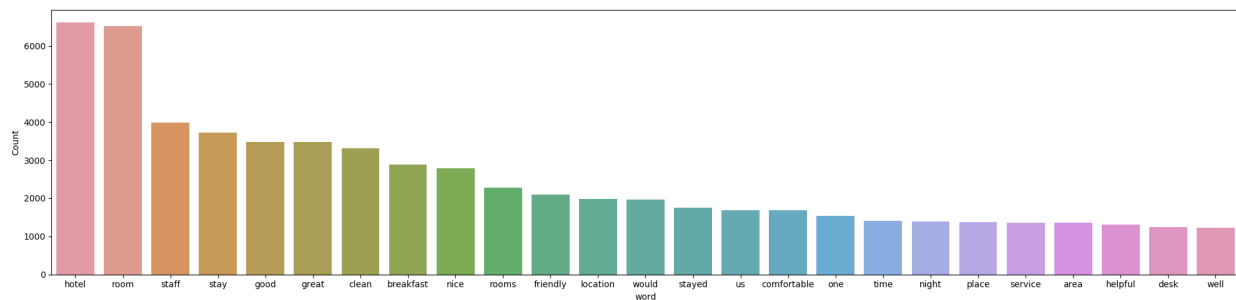
Data Mining / Text Mining

Data mining is about dealing with problems provided by information systems such as databases. It is also a technology for exploring data and find undiscovered patterns (García, Luengo, & Herrera, 2015, pp. 1). According to the previous definition data mining is about to extracting valid data patterns and do predictions.

The first pre-processing step is to remove the stop words, which commonly occurring words should be excluded in our text data. Here, we utilized the list of stop words from the predefined nltk library to identify and remove the stop words. A graph with the most frequent words is shown below:



These words were removed from the instances to avoid noise. After the process, we can see a different result:



By checking the uncommon words list, we realized most of the rare words come from the typing error, special sign and meaningless words. Those can create the noise. Our second step is to replace the rare words with a more general form, so that we can have higher counts. The similar process, we also checked the most commonly occurring words. However, we realized some of the words may be valuable to determine the positive or negative attitude. So, we decide to leave them in the text dataset.

Due to the nature of the online review data, there is no auto spelling-check and people trend to write the online comments without reviewing them. We have seen those online reviews with a plenty of spelling mistake. In this regard, spelling correction is necessary step for us to reduce the duplication meaning of words. For example, “Hotel” and “Hotal” will be treat as different words even if we can tell this is actually cause by the misspelling. The textblob library is used to identify and correct the spelling errors.

The fourth and fifth step is tokenization and lemmatization. Tokenization is the method consist in breaking out sentences in “reviews text” and separating it by words and punctuation. According to Stanford NLP online instruction, the lemmatization refers to “doing things properly with the use of vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word.” This is an example of this process:

Before the lemmatization and tokenization:

- experience rancho valencia absolutely perfect beginning end felt special happy stayed would come back heart beat

Tokenization:

- ['experience', 'rancho', 'valencia', 'absolutely', 'perfect', 'beginning', 'end', 'felt', 'special', 'happy', 'stayed', 'would', 'come', 'back', 'heart', 'beat']

Lemmatization:

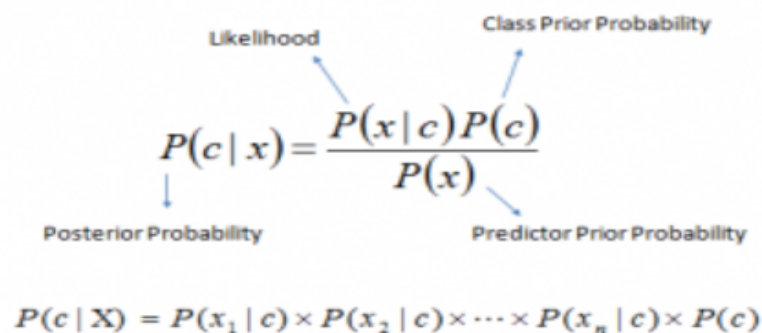
- experi rancho valencia absolut perfect begin end felt special happi stay would come back heart beat

Experimental Setup

We implemented three methods to predict the rating of the review by using the column “reviews_text”. The methods that we implemented were naïve Bayes, Support Vector Machine with RBF kernel and Linear Vector Machine with Lasso penalty.

Naïve Bayes

This algorithm is based in probability. It is considered reliable and useful to work with big datasets; specially with categorical variables. For this reason, it has been used for a long time in classification problems such as text classification, spam filtering and sentiment analysis. Given that we decided to try this algorithm to verify how precise is the prediction according to the rate review.



The diagram illustrates Bayes' theorem with the following components and labels:

- Posterior Probability:** Labeled as $P(c | x)$ with a blue arrow pointing to the numerator of the main equation.
- Likelihood:** Labeled as $P(x | c)$ with a blue arrow pointing to the first term in the numerator.
- Class Prior Probability:** Labeled as $P(c)$ with a blue arrow pointing to the second term in the numerator.
- Predictor Prior Probability:** Labeled as $P(x)$ with a blue arrow pointing to the denominator.

$$P(c | x) = \frac{P(x | c)P(c)}{P(x)}$$

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

Figure 1. Bayes theorem

$$P(\text{rating}|\text{review}) = \frac{P(\text{rating}|\text{review}) \times P(\text{rating})}{P(\text{review})}$$

Figure 2. Bayes theorem - Our case

Since, we want to know what are the probabilities of getting a review with one, two, three, four or five categories. We can convert the formula in figure 2 in the following equation:

$$P(\text{rating}|\text{review}) \times P(\text{rating } 1)$$

$$P(\text{rating}|\text{review}) \times P(\text{rating } 2)$$

$$P(\text{rating}|\text{review}) \times P(\text{rating } 3)$$

$$P(\text{rating}|\text{review}) \times P(\text{rating } 4)$$

$$P(\text{rating}|\text{review}) \times P(\text{rating } 5)$$

Figure 3. Bayes theorem modified to our case

This algorithm basically tries to find the mean of a word according to the class. In our case the classes that were used for this model were define as 1,2,3,4 and 5; being 5 the best rating for the review. Since we have 5 classifiers a multinomial naive Bayes was implemented in our project and we did use of this variables as a label to do our prediction.

Support Vector Machine with RBF kernel

support vector machine is an algorithm that determines the best decision boundary between vectors that belong to the given group and vectors that do not belong to it. Here, we choose to use the SVM RBF Kernel for the purpose of comparing the result with the other two methods. The RBF kernel is defined as below. The $\|\mathbf{x} - \mathbf{x}'\|^2$ recognized as the squared Euclidean distance between the two feature vectors.

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right)$$

Figure 2. RBF kernel

Linear Vector Machine with Lasso Regression

With this model, we are trying to improve the results obtain by the support vector machine, and have a better accuracy in the predictive model. The basic (linear) maximum margin program is defined by the following optimization problem:

$$\begin{aligned} \min_{\mathbf{w}, b, \xi_i} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T \mathbf{x}_i - b) \geq 1 - \xi_i \quad i = 1, \dots, m \\ & \xi_i \geq 0 \end{aligned}$$

This program finds a hyperplane between the groups of data points and uses that to categorize new data. In this model, we are making use of two additional models: Chi-squared and lasso regression for feature tuning. LASSO (Tibshirani, 1996) solves the following optimization problem where t is a tuning parameter:

$$\min \|X\beta - b\|_2^2 \quad (1)$$

$$s.t. \sum_{j=1}^d |\beta_j| \leq t \quad (2)$$

LASSO will set some coefficients $\beta_j = 0$, Shrinks the parameters (and variance) and introduces bias to decrease mean squared error.

In the case of the Chi-squared, we are using it to select the best features out of the bag of words to train our model. In our case, we are using $k=10,000$ to select the best 10,000 features, by giving weighting to each of the features.

On the other side, we are also making use of lasso to shrink the parameters (variance) and decrease the mean squared error. We also need to mention that our test size for this model is 25%, the vectorization of n-grams was from 1-3.

It is also valid to mention that for this model, since we are trying to predict the rating of the hotel based on the reviews, we had to standardize the rating values. In the data set, rating could be decimal numbers (such as 1.2, 1.5, 4.8, etc). For this model, we round those numbers to improve the accuracy of the model; which means that we are only going to consider integers numbers from 1 to 5.

Chapter 4: Results

By using first method naive Bayes we wanted to find out the probabilities that a review is proper defined between categories 1 to 5. The results obtained by this algorithm showed an accuracy of 48% which means that the precision for our model it is not very accurate. Below it can be seen a sample of the prediction:

- 5 => pleasant surprise cottages neat clean perfect family toasty warm even temps dipped windchill night full sized oven stove fridge plenty space fit whatever food snacks need bring shower
- 4 => returned week long stay Americana want fancy don't stay complaints breakfast morning fine us bagel juice roll muffin coffee piece fruit good rooms clean yes older motel
- 4 => bad beds creaky thin walls hear everything room next door hallway housekeeping isn't consistent laundry delivery wrong room good location
- 4 => really nice smallish resort right lake geared towards families smaller children allow pets quite dogs upon checking gave us treats dog nice room clean resort nice little beach lots things kids
- 5 => bedroom suite right husband daughter enough space spread stay candlewood whenever visiting family nj like kitchen full size refrigerator microwave dishwasher two burner stove like free laundry place go
- 4 => bad bed small two people blanket thin staff ok friendly though good location new England style building room cute

- 4 => smooth early check given upgraded room room gorgeous enjoyed jacuzzi casino little smokey buffet nice although find piece plastic greens manager apologize would rated higher awakened
- 5 => want thank wonderful service received best western plus encino visiting mother cottage hospital front desk clerks housekeeping cordial courteous best western

According to the sample showed above, it can be seen that indeed the model is not predicting very well and may be convenient to do have other approaches to improve the model.

Using the second method the Support Vector Machine with RBF kernel, we finally got the accuracy of 46.2%. It is the lowest outcome, comparing with the other two method. We can say that it doesn't give us any better performance by nonlinearly mapping samples into a higher dimensional space.

Finally, from the results with the third method (Linear Vector Machine with lasso penalty). The model has an accuracy of 50%. Considering that the dataset has only 10,000 rows, we would expect that the model will improve its accuracy. To achieve this result, as there is no specific formula for all the cases, we needed to adjust few parameters. In the first case of the test size, we tried with values from 20% to 35%; which end up being 25% that size of the test that will give the higher accuracy.

In the case of the n-grams, we started using 1-1, 1-2 and 1-3 N-grams. We realize that for this model, 1-3 was the best value for the model. For the K-value for the chi-squared, we also had to define which number was improving the result, and we can conclude that k=8000 was giving the best result.

Below is an example of the results obtained by the model. As we can see, the model is classifying the reviews to each rating. The reviews are ordered by the number of rating, and it's being extracted as a result of the algorithm.

1: dump rude staff disgusting room people refused need repair breakfast front room immediately worst pest

2: service staffs micro reset bad cleanliness everything old needs major made difficult go dinner falling apart complained

3: sparse unacceptable disorganized incompetent pillow case end night show age lot desired today better places

4: ac okay door bad room great hotel bad pool however good hotel bad valet nice price

5: great experience lovely thank cant wait love hotel amazing excellent exceptional loved hotel one best

The accuracy score of this model is 50%.

Chapter 5: Summary and Conclusions

Our project used text mining techniques to draw meaning out of the written online reviews. Unlike normal data mining, most of the text mining data is unstructured with a content that can be valuable. However, it requires to implement several steps of preprocessing to extract the meaningful information.

In our project, we use different preprocessing methods, such as removing common words, stop words, rare words, lemmatization, stemming, and spelling correction. At the end, we reduced our vocabulary cluster so that features produced for the classification model would be more accurate.

According to our result from comparing the accuracy rate among the three models, we can tell that the linear vector machine had a better performance. Following are some explanations about the reasons to lead this outcome. First, the text has a lot of features and the linear kernel is good when lots of features are used. It doesn't really help to increase the dimensional space like using the RBF Kernel. Second, analyzing the text related information means that it takes more time. Training with linear kernel is faster. Third, comparing with the other kernels, the linear need less parameters to optimize when train the data.

We want also to discuss the improvements that have been made for the better outcome. In case of the linear vector machine with lasso penalty, the model could predict a 50%. If we consider that the data it is only 10,000 instances, we would expect that the model increase the percentage of prediction with more data. We can also conclude that the model is better with 1-3 N-grams, compared to 1-1 or 1-2. We have also improved this model by using chi-squared to improve the feature selection.

Last but not least, by re-evaluating the observations from the sample dataset, we found out most of the comments are subjective that people may use their own "rating system" to give their own hotel rating. Also, people are more likely to give the 4+ or 5 score even they have slightly positive hotel experience. As result, we can see some limitation by nature on predicting using the "1 to 5" rating system. To get a better outcome, we went back to the preprocessing stage and did the test to use different scales on our hotel rating column. As we can see in our comparison by taking SVM Kernel as an example, if we rescale to the negative (for ratings from 0 to 3), Moderate (for ratings from 3 to 4) and positive (for ratings from 4 to 5), our accuracy rate will significantly increase from 46.20% to 74.36%.

Reference

García, S., Luengo, J., & Herrera, F. (2015). Data preprocessing in data mining. Cham: Springer.

<https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html>

A. N. (n.d.). *Support Vector Machines*. Lecture

<http://cs229.stanford.edu/notes/cs229-notes3.pdf>

Jain, S. (2019, March 11). Ultimate guide to deal with Text Data (using Python) - for Data Scientists and Engineers. Retrieved from <https://www.analyticsvidhya.com/blog/2018/02/the-different-methods-deal-text-data-predictive-python/>

R. (n.d.). *An Idiot's guide to Support vector machines (SVMs)*. Lecture.

Text Analytics for Beginners using NLTK. (n.d.). Retrieved from

<https://www.datacamp.com/community/tutorials/text-analytics-beginners-nltk>

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. J.R.S.S.B. 58, 267-288

A. (2016, February 24). *Topics in sparse Support Vector Machines*. University of Arizona.

<https://www.math.arizona.edu/~wammonj/talks/topics-sparse-svms.pdf>

J., S., T., & R. (n.d.). *1-norm Support Vector Machines*. Stanford University

<https://papers.nips.cc/paper/2450-1-norm-support-vector-machines.pdf>