University of Melbourne

MAST30034 Applied Data Science

Airbnb Rating Analysis

Shiwen Wei(957082), Xiangcheng Su (928241)
Ze Pang (955698), Zitao Jiang (987857)

October 2020

Abstract

With the rapid spread of information today, viewing reviews has become the norm for online purchases. According to the research of Georg L, Daniel K and Kenan K, 85.57% of consumers read reviews often or very often before they purchase online. Reviews will be more valuable to refer especially when people cannot physically experience the product. In this project, we use several approaches **(Doc2Vec, Naive Bayes, Logistic Regression, Neural Network, Decision Tree)** to build a Natural Language Processing(NLP) prediction model for an Airbnb recommendation system based on datasets from Kaggle. Neural Network provides relatively better results and the highest accuracy of 0.9747.

1 Introduction

Airbnb is an open platform providing inexpensive hotels with different kinds of properties for travelers to choose. There are over 150 million users worldwide on Airbnb in 2020, so it is worth exploring housing information and evaluations to construct a suitable recommendation system to bring better user experience. This report will introduce the chosen datasets, explore applied methodology, and evaluate key findings.

2 Dataset

2.1 Dataset Selection

Datasets are chosen from Kaggle and generated by Murali(2020):

- **Airbnb ratings new.csv (1048575, 35):** A multi-variate dataset of Airbnb Listings. It contains owner identification, specific geographic location, room type, and review scores
- **Airbnb-reviews.csv (8348173, 6):** It includes comments of each order.

2.2 Processed Datasets

After pre-processing (removing missing values, outliers, merging different datasets together), processed datasets are shown as follows:

- **airbnb_ratings_processed.csv:** UID "listing_id" with entire Airbnb information such as Response Rate, Location, number of rooms, prices and ratings in different aspects (About 250,000 unique Airbnb after preprocessing)

- **airbnb_reviews_en_doc2vec50.feather:** listing_id, reviewer_id, reviewer_name and 50-dimensional sentence vector from doc2vec
(About 7 million reviews in total, 28 reviews per Airbnb)
- **airbnb_reviews_en_join_doc2vec50.feather:** listing_id, 50-dimensional sentence vector after combinning all reviews from the same Airbnb
(About 350,000 reviews in total)
- **df_final.feather:** 50-dimensional sentence vector, Average Score
(Same shape and order as airbnb_review_en_join_doc2vec50.feather, the final dataset)

2.3 Exploratory Data Analysis

Exploratory data analysis is a necessary step to analyze data sets and examine characteristics of features. This section will explore datasets using visualization.

The first analyzed attribute is Maximum nights, it represents the longest stay in the listing. To determine the overall distribution of this continuous attribute, all nights will be cut into 7 bins: (0, 10], (10, 20], (20, 50], (50, 100], (100, 500], (500, 1130] and greater than 1130 nights.

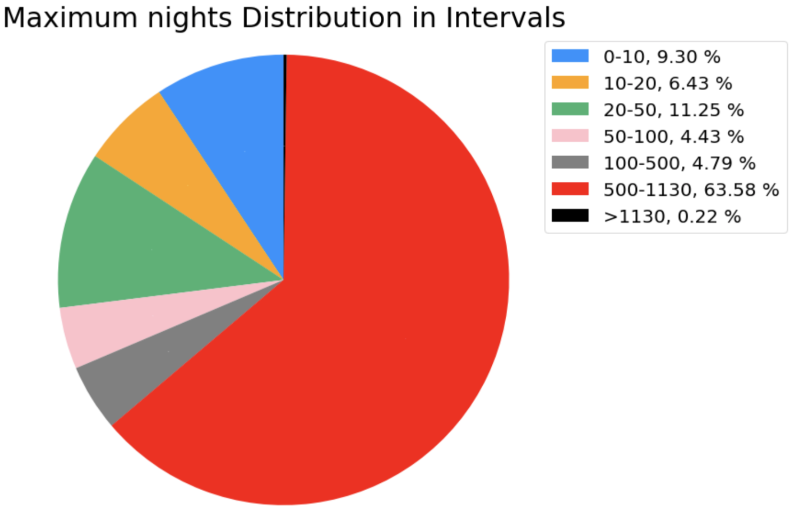


Figure 1: Pie Chart of Maximum nights Distribution

In Figure 1, it can be seen that the majority of the lease term are within 500-1130 nights, users prefer long-term stable accommodation. It seems that they are settled in somewhere through the Airbnb platform. Thus Airbnb can recommend more listings that are available for long-term lease to satisfy more users' requirements.

High score value represents the high level of tenants' satisfaction, it is one of the most important factors to determine the quality and rating of listings. So the distribution of high rating listing will be explored.

Figure 2 shows the number of listing at different scores. The range of the score number is between 0 and 10, but it seems that there is no listing that has a score of 1. Moreover, score 9 and 10 account for majority proportions, it means that listings posted on Airbnb have generally high quality and satisfy users' demands. The platform should promote these high rating listings, and it is also good proof that Airbnb's house registration selection is generally perfect.
Review comments can also determine the rating of listings, it is important to evaluate the number of reviews. Review number is split into 6 groups: (0, 5], (5, 10], (10, 20], (20, 50], (50, 100], (100, 800] to plot a bar chart and examine the distribution.

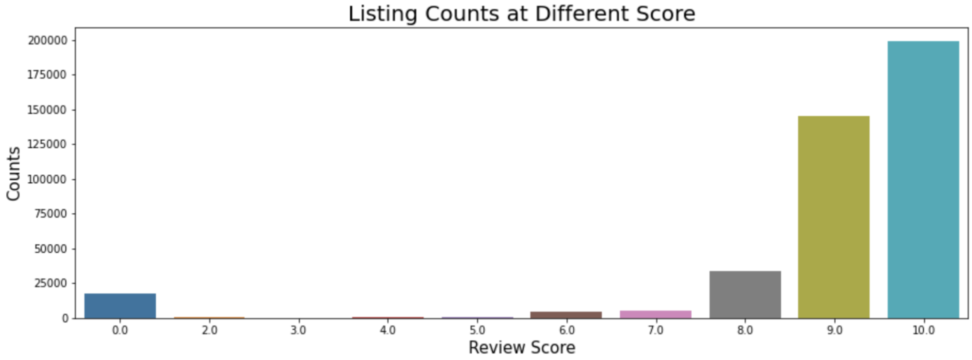


Figure 2: Number of Listing at Different Scores

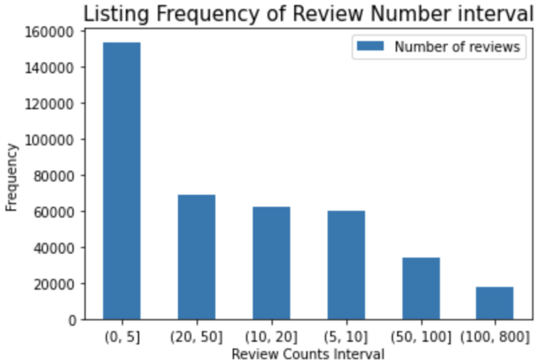


Figure 3: Number of Listings in Review Number Interval

Figure 3 provides a number of listings within different review counts intervals. Many listings have no more than 50 reviews, it seems that some tenants are not willing to share their living experiences. Airbnb can introduce some activities to encourage users to write comments.

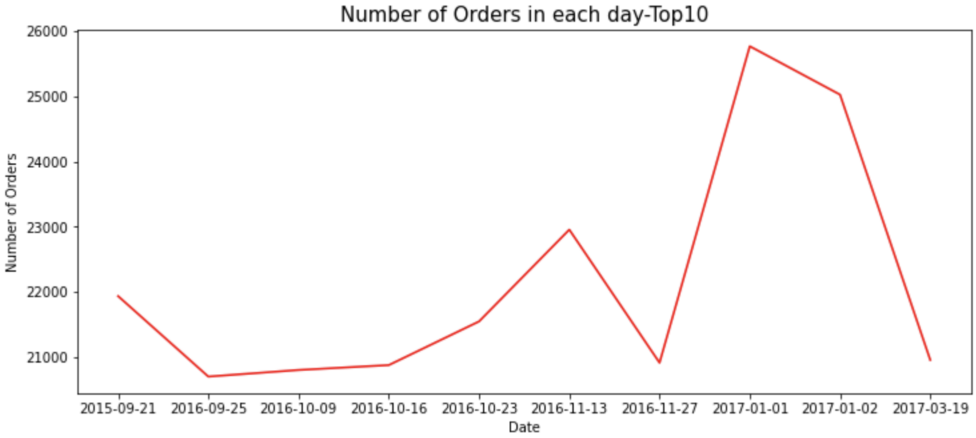


Figure 4: Date of Top 10 Orders

Figure 4 shows the line chart of date with the highest orders, only 10 days with the highest orders displayed to better visualize the trend. It is worth to mention that there are most orders on the first day in 2017. Generally, there will be more orders in the first and fourth quarters.

3 Methods

3.1 Sentiment Analysis

Sentiment analysis model learns the input text and returns polarity and subjectivity values of text. In this project, the polarity of reviews becomes an indicator that we can use to judge the tendency of the reviews in addition to the response. The boundary value between positive reviews and negative reviews is obtained by the average value of reviews with negative polarity.

3.2 Grid Search

Grid Search is a technique to generate combinations of hyper-parameter and then output the optimal results. It will be used to select the best hyper-parameter for models.

3.3 SMOTE Algorithm

SMOTE (synthetic minority oversampling technique) is an oversampling method to reduce the impacts of imbalance distributed data.

In the rating data set, the scores are generally high, which will lead to bias towards the majority class. Assume that the **Majority class A** contains the data with a rating higher than 7, while the **Minority class B** contains data with a rating lower than 7.

For each instance in **Minority class B**, the k-Nearest-Neighbours will be calculated, the **k** is a hyper-parameter of SMOTE Algorithm.

By randomly selecting **N**(Hyper-parameter: Ratio of SMOTE N%. Generally will be 100%, 200%, 300% and other whole hundred numbers, corresponding to once, twice, thrice the size of minority class B) data from k-nearest-neighbors as the new set $\mathbf{B_1}$
For each instance $\mathbf{x_i}$ (i = 1, 2, ... N/100) in set $\mathbf{B_1}$, new samples' will be generated based on following formula

$$\mathbf{x' = x + rand(0, 1) + |x - x_i|}$$

3.4 Doc2Vec

Doc2Vec is a method for numeric representation of a document. For this research, all the model required a numerical input. They can not handle text inputs. Therefore it's necessary to transfer texts into vectors. Doc2Vec is an advanced version of Word2Vec. Like Word2Vec, it captures the relationship between different words.

3.5 BASELINE MODELS: 0R & Naive Bayes

0R is a popularity-based-model, which selects the label with the highest frequency as the classification result. After being processed by Doc2Vec, reviews will be vectorized into a 50-dimensional vector. **Naive Bayes Classifier with Gaussian Model** is a probability-based-model that dealing with numeric features.

3.6 Logistic Regression

Logistic Regression is a statistical model for binary classification, which is suitable for this topic. It can train data sets efficiently and less inclined to over-fitting. However, it can only predict discrete attributes, so labels should be discretized before fitting the model.

3.7 Decision Tree

Decision Tree is one of the most popular supervised machine learning strategies. The algorithm for decision tree makes it relatively faster to build as well as faster to train than other learning methods. Also, the decision tree handles both continuous and categorical variables which are nice in this case. It's a white box model other than the neural network which means it's easy to interpret and understand.

3.8 Neural Network

The **Neural Network** is a model that relies on the complexity of the system and achieves the purpose of processing information by adjusting the relationship between a large number of internal nodes. Basically, a neural network is composed of an input layer, some hidden layers and an output layer. When dealing with a binary classification problem with vectorized text in this project, a neural network with less than 3 hidden layers will be sufficient.

4 Experiments and Results

4.1 Text Vectorization

For the review dataset, many comments are not written in English. Considering the huge amount of data and translation work, we decided to drop all the non-English comments. Since we don't have a direct rating for each comment, so all the comment for each house has to be concatenated for the merging purpose.

In order to vectorize the English comments and easier to apply the model later, we decide to use Doc2Vec for this problem. Doc2Vec is an unsupervised machine learning technique in order to find the similarity and relation between each word by vectorizing them using the following steps. The first step is considering all the comments in the review file as a corpus and tokenizing them into lists of words. Then build a model and use all the corpus to train the model and finally infer the document embeddings using the same data.

4.2 Balance Data Distribution

Before using the **SMOTE Algorithm**, the response "Average score" will be discretized into 0 and 1 based on the sentiment score, representing whether Airbnb is worth choosing. The SMOTE algorithm randomly generates new samples based on existing minority samples. By weighing the ratio in the original data set and the generalization ability of the model, we think that the ratio of positive and negative samples of 2:1 will be appropriate.

4.3 BASELINE MODELS

Accuracy of the **0R model** will be controlled by the "ratio" of the SMOTE algorithm, which is about 60%.

After confirming features have an approximately normal distribution(Assumption of **Gaussian Naive Bayes Model**), the accuracy of unbalanced data is about **58.69%** by fitting the model, while the accuracy of the data balanced by the SMOTE algorithm can reach **69.74%**.

4.4 Logistic Regression

4.4.1 Tuning

To improve the performance of the model, Grid Search will be applied to select the most suitable model. This report will mainly focus on two hyper-parameters of Logistic Regression: C value and multi_class. C represents regularization strength, larger values indicate weaker regularization, and multi_class means types of multi classification method. The chosen test values for C are 0.01,0.1,0.5,5,10,20,50,100, test values for multi_class are 'multinomial' and 'ovr'. Grid Search spends around 37 seconds to select the best model.

Figure 5 shows the visualization of hyper-parameter selection result. It can clarify the optimal hyper-parameters: C=0.5, multi_class = 'multinomial'. And the relatively higher accuracy is around 0.812575, there is no significant difference between hyper-parameters, the general accuracy is around **81.3%**.

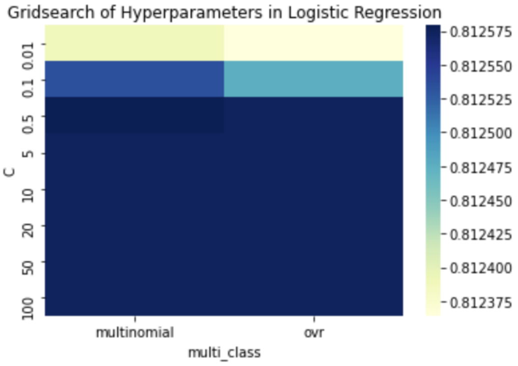
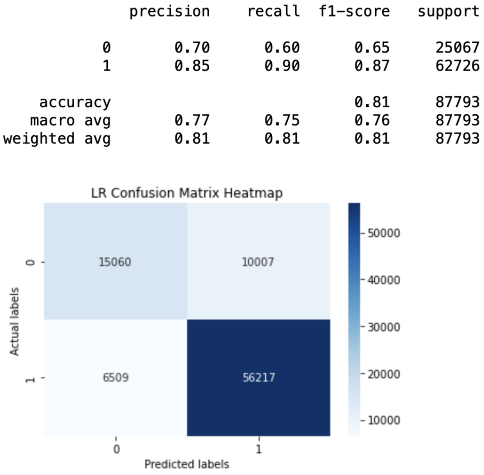


Figure 5: GridSearch Heatmap of Logistic Regression



```
              precision    recall  f1-score   support

           0       0.70      0.60      0.65     25067
           1       0.85      0.90      0.87     62726

    accuracy                           0.81     87793
   macro avg       0.77      0.75      0.76     87793
weighted avg       0.81      0.81      0.81     87793
```

Figure 6: Confusion matrix of Logistic Regression

4.4.2 Evaluation

Figure 6 contains both classification report and confusion matrix heatmap of Logistic Regression after fitting the model. They indicate the prediction performance. Prediction label 1 has relatively better performance, higher precision and recall, which means label 1 can be more easily to be detected and predicted. There is a large gap between labels 0 and 1, the possible reason is the biased datasets. Positive comments take the majority proportions of all reviews, which means there are no enough negative comments in training sets, two labels are distributed unevenly. Therefore, the overall performance of the Logistic Regression model is acceptable, and extra improvements can be applied to generate a more accurate model.

4.5 Decision Tree

4.5.1 Tuning

Since the target variable has been discretized into discrete value, therefore the exact method for this problem is using a Classification Tree. The hyperparameter for the classification tree is the maximum depth and the method to find the optimum depth is using exhaustive search. Figure 7 shows both training and testing accuracy for different depth, with minimum accuracy when depth equals 1 and surges when depth increasing. When depth is greater than 10, the training accuracy and testing accuracy start showing a different increase trend and both almost reach their maximum between 20 and 30. After that, both trends are flattened.

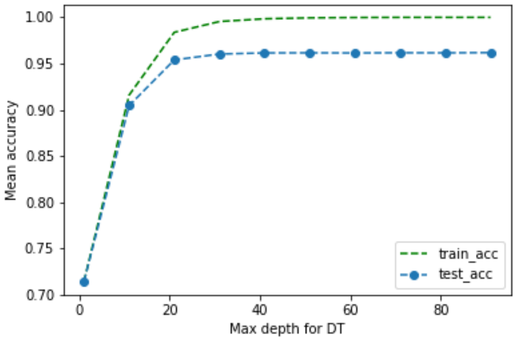


Figure 7: Training&Validation Mean Accuracy

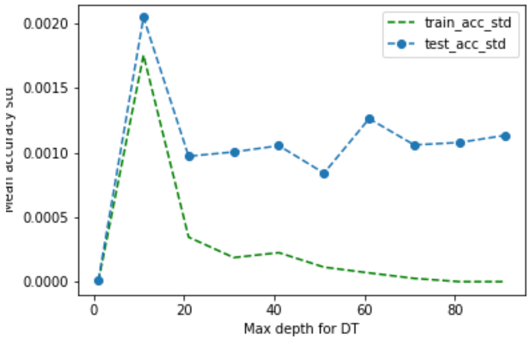


Figure 8: Training&Validation Mean Accuracy Std

4.5.2 Evaluation

The line plot in Figure 8 indicates the standard deviation of mean accuracy after using cross-validation for each depth. From the graph, Both train and test standard deviation have peaked around 10 and dropped significantly after that. This means the depth of around 20 makes the model consistent and accurate when combined with the previous graph. Similarly, both trends become steady after 20 although the line for test standard deviation has fluctuated around 60. Therefore, the optimum max depth for this model is around 20 with relatively high accuracy and consistency. The mean test accuracy for the classification tree using this as the parameter is around **0.95**, which is quite high compared with logistic regression.

4.6 Neural Network

4.6.1 Tuning

A Multi Layer Perceptron with
Input layer: Input_dimension = 50, relu activation, output_dimension = 101
Hidden layer: Input_dimension = 101, relu activation, output_dimension = 101
Output layer: Input_dimension = 101, sigmoid activation, output_dimension = 1 is built for this binary classification problem.
The layer dimension is based on the Kolmogorov theorem mentioned in Robert H's research when the number of unit in the input layer is **n**, then the number of the input unit of the hidden layer is **2n + 1** and the sigmoid activation is used to map the output into a number between 0 - 1.
To avoid the over-fitting issue, a validation set is separated from the training set, it will be used to compare with the accuracy and loss of the training set in each epoch.

4.6.2 Evaluation

The fluctuation that appears around the tenth epoch represents the tendency of the model to overfit. To retain the generalization ability of the model, the training process will be terminated at this epoch.

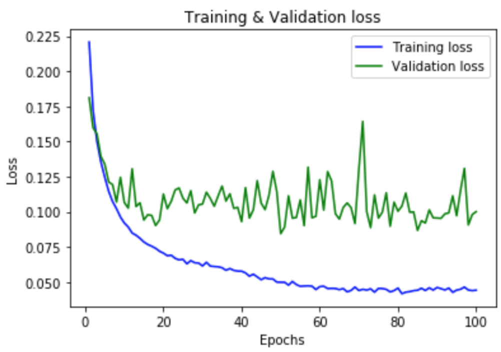


Figure 9: Training & Validation Loss

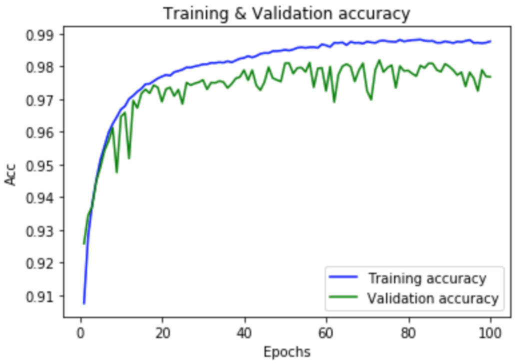


Figure 10: Training & Validation Accuracy

5 Result & Discussion

Model	Accuracy (Test set)	Precision	Recall	R1	Hyper-Para
Baseline - 0R	0.60	-	-	-	-
Baseline - GNB	0.6974	-	-	-	-
Logistic Regression	0.8119	0.85	0.90	0.87	C=0.5 multi_class=multinomial
Decision Tree	0.9473	0.98	0.94	0.96	Depth: 20
Neural Network	0.9747	0.99	0.98	0.98	Epoch: 12

Figure 11: Result Table of Prediction Models

Neural Network turns out to be the best model in predicting the goodness of Airbnb based on vectorized text features. Compared with the other two models, the accuracy of the neural network model is about 17% higher than Logistic Regression and about 3% higher than Decision Tree, and precision and recall are also higher than expected. The performance is surprisingly good in multiple test sets. However, this project only explored models with limited hyper-parameters, more tests can be applied to generate more comprehensive models. Moreover, the labels predicted by the Logistic Regression model are distributed unevenly, it may due to the unbalanced number of comment polarity. This dataset lacks negative reviews, the data itself should be refined. It could also be that Airbnb chooses their listings very carefully, so positive comments are much more than negative comments.

In conclusion, this report has already explored Airbnb Rating Datasets, applied Natural Language Processing and evaluated multiple prediction models. The overall results are better than expectations, but there is no doubt that these models still have the potential for improvement when they are applied as the realistic recommendation system. Furthermore, extra researches will be focused on the methodology of processing diversified inputs, including pictures of listings using CNN(Convolutional Neural Network), pre-processed location information, and audio recognition. Moreover, since the uniqueness of Users' ID, a system can be built to collect and learn a specific user's living preferences and then recommend other Airbnbs that similar to user preferences. It is necessary for Airbnb to expand the user community.

6 Reference

Murali S. (2020, May). Airbnb Ratings Dataset, Version 11. Retrieved September 17, 2020 from https://www.kaggle.com/samyukthamurali/airbnb-ratings-dataset

Georg L, Daniel K and Kenan K. (2013). Importance of Online Product Reviews from a Consumer Perspective. Retrieved October 20, 2020 from http://www.hrpub.org/download/201307/aeb.2013.010101.pdf

Nitesh V, Kevin W, Lawrence O and Philip K. (2002, June). SMOTE: Synthetic Minority Over-sampling Technique. Retrieved October 21, 2020 from https://arxiv.org/pdf/1106.1813.pdf

Robert H, Kolmogorov's Mapping Neural Network Existence Theorem Retrieved October 22, 2020 from https://cs.uwaterloo.ca/~y328yu/classics/Hecht-Nielsen.pdf