Date May 6, 2020

BOSTON AIRBNB FEEDBACK ANALYSIS

IST664 group3



**Instructor:**

Mei Zhang

**Members:**

Renjie Zhu(ADS) Chaoying Lyu(ADS) Zeyi Luo(BA)

Mengdie Zhuang(IM) Ruiwei Zhang(ADS)

CONTENTS

[1. Abstract 3](#_Toc39612389)

[2. Methods 3](#_Toc39612390)

3. Preprocessing………………………………. 3

[4. Modeling 6](#_Toc39612391)

[4.1 K-means Clustering​ 6](#_Toc39612392)

4.[2 Sentiment Polarity for Reviews—Bigram 7](#_Toc39612393)

[4.2.1 Comment level 7](#_Toc39612394)

[4.2.2 Sentence level 8](#_Toc39612395)

[4.3 Latent Dirichlet allocation (LDA)​ 10](#_Toc39612396)

[5. Model summary 16](#_Toc39612397)

[6. Business question and conclusion 16](#_Toc39612398)

[7. Reference 17](#_Toc39612399)

# Abstract

Since 2008, guests and hosts have used Airbnb to expand on traveling possibilities and present more unique, personalized way of experiencing the world. The Airbnb has been trying to improve their service. In the past few years, applying Natural Language Process (NLP) to analyze users’ satisfaction becomes a trend for service improvement. Hence, we find a dataset of the guests’ comments on the Airbnb services.

This dataset describes the listing activity and metrics in Boston. ​We are focusing on the following Airbnb activity included in this Boston dataset. Based on the comments of the reviewers, we want to dig deeper on the general feedbacks of each Boston neighborhood and give specific suggestions on the Airbnb services.

The sentiment analysis will be done on both the paragraph and the sentence level. What users pay attention to is the target of the analysis.

# Methods

Our sentimental analysis will be applied on both sentence and comment level.

The k-means model will be applied to discover the similar patterns. This will classify the words in the comments into different categories.

The sentiment polarity of comments is also in our consideration. In this step, words will be judged whether they are positive or negative.

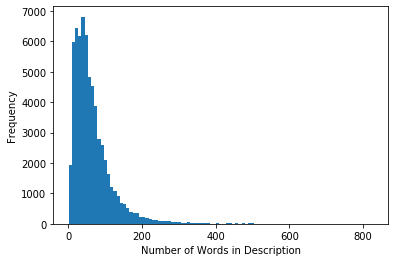
After this step, it could be seen clearly that what kinds of words are the guests’ focus. In addition, the Latent Dirichlet allocation (LDA) will be used for topic analysis. It can separate the topics that users pay most attention on. In this way, specific suggestions can be given to the Airbnb for future improvement.

# preprocessing

Before building a K-means model, we did some exploring and preprocessing to the data. Firstly, we explored the distribution of number of words in the customers review.

A screenshot of a cell phone

Description automatically generated



From the summary information of the distribution and the histogram, we could see that the distribution of number of words was a right skewed one but with a wide range of distribution (from 1 to 827). This could be beneficial to the k-means model, since the model using tf-idf score as features and we could get high variance tf-idf scores from the data set.

Lowercase all the words



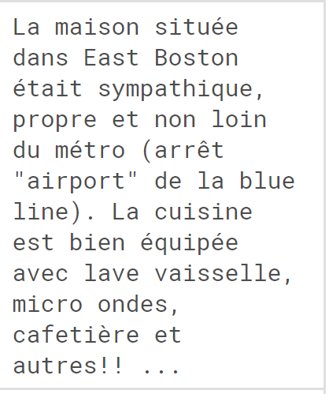
NA removement



Tokenization



Language detection



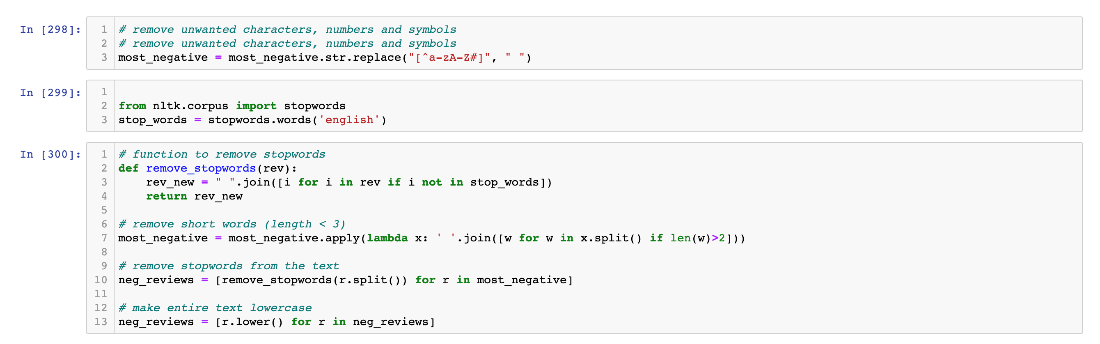
As for the preprocessing part, we removed the stop words and negation words, did the lemmatization and used the tf-idf vectorizer to transform the original data.

In order to improve the performance of our clustering algorithm, we chose to add some stop words to the original nltk stop words. Because the negation words only reflected the customers’ attitudes and we cared more about the topic words, so we removed them.

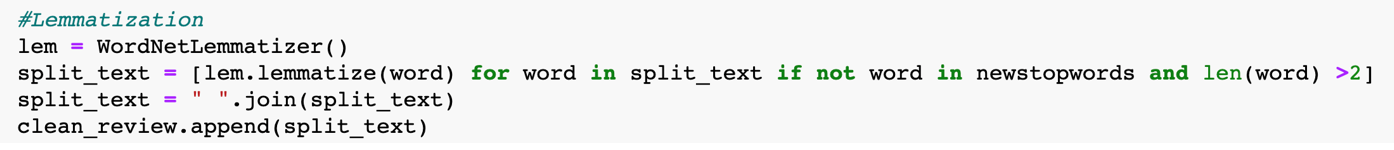
A screenshot of a cell phone

Description automatically generated

Remove all the numbers, the characters that are not alphabetical, the stop words, words whose lengths are less than three



Then we did the lemmatization to the words that not appeared in the stop words and have a length more than 2.



# Modeling

## 4.1 K-means Clustering​

Since our Airbnb data set does not have a target variable such as score or attitude. It would be suitable to use unsupervised learning to do some analysis.

K-means is a simple but powerful unsupervised learning algorithm (meaning there are no target labels) that allows us to identify similar groups or clusters of data points within data. Using this algorithm, we could cluster our customer reviews into some specific number of groups to figure out what words the customers care the most and we could gain some insights about the service quality in the Boston area.

After those preprocessing, we used the *TfidfVectorizer* class to do the vectorization on the cleaned reviews.

A screenshot of a cell phone

Description automatically generated

Then we built a 10-clusters K-means clustering model using the sklearn package in python. We set the max number of iterations as 200 to decrease the execution time and set the relative tolerance as 0.01 to improve the performance.

A screenshot of a cell phone

Description automatically generated

The clusters we got were as follows.

A close up of text on a white background

Description automatically generated

From the visualization we could see that the first several clusters were pretty clear about certain topics. For example, the cluster 0 included ‘canceled’, ‘day’, ‘host’ and some numbers, which could be parts of the time topic. The cluster 1 included ‘house’, ‘room’, ‘stay’, we could consider it as the topic mainly about the living environment. What is more, there were some words that showed in nearly every cluster such as ‘clean’ and ‘comfortable’. Those words could reflect what customers care the most and are valuable for us to dig deeper.

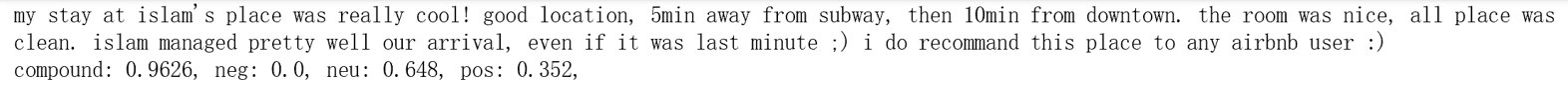
To summarize, we used K-means algorithm as our first unsupervised learning method. We built a 10 clusters K-means model and found that time, comfortability, location would be the most important features of service quality. And ‘clean’, ‘comfortable’ would be the evaluation factors that nearly every single customer would consider.

## 4.2 Sentiment Polarity for Reviews—Bigram

### **4.2.1 Comment level**



This package labels comments based on positive, negative, and neutral words in the comments, and then come up a compound score to identify the polarity.



These three charts illustrate the distribution of positive, negative, and neutral score of all comments. We can learn that most words are positive and neutral.

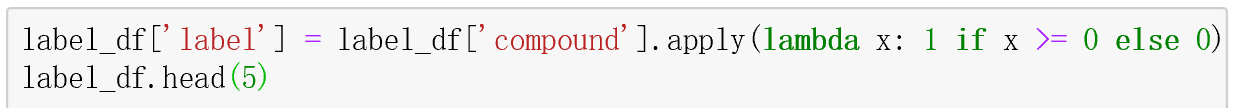
A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

To find out more informative negative comments, we label the comments 1 when its compound score above 0, others label to 0. Exact all the comments with label 0 to do bigram analysis.



1. Bigram

A close up of text on a white background

Description automatically generated

  ‘Impossible find ‘is the only one clear positive comment we find, and the other most frequent bigrams are more about positive comments.

1. bigram\_measures.raw\_freq

A close up of a newspaper

Description automatically generated

Almost all bigrams here are positive, which means the way we label comment based on compound score are not accurate and suitable to our analysis

1. bigram\_measures.pmi

A close up of a newspaper

Description automatically generated

‘stone throw’ is the negative comment, and here are 5 fields--

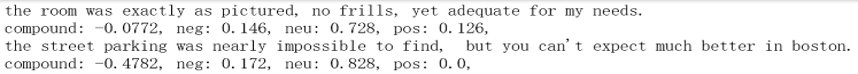
‘Ear plugs’​, ‘Commuter rail’​, ‘Newly renovated’​, ‘Closet hangers​’—that we can pay more attention to improve the service quality.

### **4.2.2 Sentence level**

We extract comments that were labeled 0 to do some sentence level analysis.

A screenshot of a cell phone

Description automatically generated

****

Similar with score distribution in comment level, most words are positive and neutral.

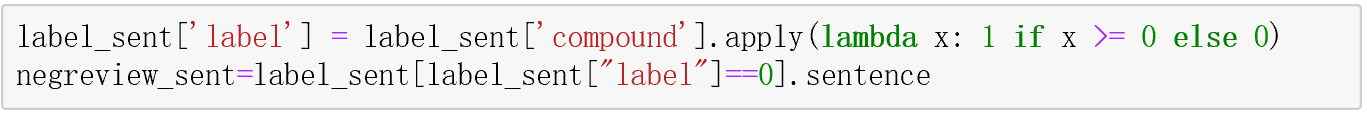
A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

To find out more informative negative comments, we label the sentence 1 when its compound score above 0, others label to 0. Exact all the sentence with label 0 to do bigram analysis.



1. Bigram

A close up of text on a white background

Description automatically generated‘impossible find’, ‘instruction problems’ are the two negative comments we find.

1. A close up of a newspaper

   Description automatically generatedbigram\_measures.raw\_freq

 ‘dirty dishes’ –clear negative comment

‘Air conditioning’​, ‘Bus stop’​, ‘Street parking’​, ‘Apartment clean’​, ‘Check time​’, ‘Grocery store’—where we can pay more attention to improve service quality

1. bigram\_measures.pmi

A close up of a newspaper

Description automatically generated

​‘stone throw’—clear negative comment

‘Ear plugs’​, ‘Commuter rail’​, ‘Newly renovated’​, ‘Closet hangers’​—where we can pay more attention to improve service quality

## 4.3 Latent Dirichlet allocation (LDA)​

Topic Modeling is a process to automatically identify topics present in a text object and to derive hidden patterns exhibited by a text corpus. Topic Models are very useful for multiple purposes, including:

• Document clustering

• Organizing large blocks of textual data

• Information retrieval from unstructured text

• Feature selection

A good topic model, when trained on some text about the stock market, should result in topics like “bid”, “trading”, “dividend”, “exchange”, etc. The below image illustrates how a typical topic model works:

A close up of a map

Description automatically generated

In our project, we have extreme Airbnb reviews. Our aim here is to extract a certain number of groups of important words from the reviews. This review can help us find the topic which they really care about. So we used Latent Dirichlet Allocation as our topic modeling technique.

# 4.3.1 prepare

What we interested are attitude of comments. To transform it into review sentiments, I used the Vader sentiment model based on NLTK.

A screenshot of a cell phone

Description automatically generated

First, we used SentimentIntensityAnalyzer to calculates compound sentiment polarity of the comments.



Then, we analyze the distrition of the comments attitude. Then

#We extracted the highest score (> 0.995) and the lowest score (<-0.8). respectively as extremely positive dataset and extrmely negative dataset.

A screenshot of a cell phone

Description automatically generated

Then we get two datasets. We used WordCloud and Folium to visualize data according word frequency and geographic distribution. In addition, we preprocessed the comment data for LDA model.



### 4.3.2 Negative Topic

We will start by creating the term dictionary of our negative corpus. Then we will convert the list of reviews (neg\_reviews\_2) into a Document Term Matrix using the dictionary prepared above.

**A screenshot of a cell phone

Description automatically generated**

Then we print result and visualization it.

**A screenshot of a cell phone

Description automatically generated** The Topic 0, Topic 1, and Topic 5 has terms like ‘bathroom, ‘clean’, ‘dirty’, indicating that the topic is very much related to bathroom. Similarly, Topic 3 and Topic 6 seems to be about the location of the apartment as it has terms like ‘place, and ‘location’.

4.3.3 Positive Topic

Besides, we do same thing with our very positive reviews.

A screenshot of a social media post

Description automatically generated

A screenshot of a cell phone

Description automatically generated

The Topic 0, topic 2, and Topic 4 mentioned ‘neighborhood’, ‘location’ which is relevant with ‘location’ of the apartment. In addition, the Topic 0, Topic 1, Topic 3, Topic 4, Topic 5, and topic 6 mention about ‘comfortable’ and ‘clean’ of ‘living environment’ Topic.

# Model summary

We used three model to analyze Airbnb reviews. These model have advantages and disadvantages respectively.

First model is K-means. We used K-means model to classify words into different groups, getting general word frequency. However, this model does not tag for reviews and it has no clear classification based on sentiment.

The second model we applied is Vader sentiment model. Now we focus on negative comments and sentences after tagging sentiment polarity and then calculate bigram frequency. But in comments level, the results show that there are lots of positive words misclassified into negative comments. In sentence level, Vader sentiment model extracts noun words without clear sentiment polarity.

The third one is LDA(Latent Dirichlet Allocation) model. After tagging positive and negative words, we use LDA to extract topics and key words from each review. The disadvantage of this model is that if the data in negative topic and positive topic are not very adequate, the accuracy of topic extraction will be low.

# Business question and conclusion

After summarizing the results got from all three models, we can make a conclusion for our business question: what are the major factors that affect customers’ satisfaction.

First, cleanliness. Customers really care about if the room, bathroom, the sheet on the bed is clean or not. It is very important factor for them to rating a house or apartment.

Second, facility, like bed is comfortable, the kitchen is well equipped, etc.

Third, location. The house near to public transportation and grocery store are always very popular. Because customers can easy travel to other place for their trip and they can buy necessary supplies nearby. ​

The last one is about service. Lots of comments mentioned about great host , feel at home. So we think the service provide by host, or the attitude of host may be a significant factor. A nice host will leave great impression to customers.​

# reference

1. IST 664, Project description documentation

2. IST 664, Project Information powerpoint file

3. Kaggles dataset：<https://www.kaggle.com/airbnb/boston>

4. Many times of review of class recording video

5. An NLP Approach to Mining Online Reviews using Topic Modeling, <https://www.analyticsvidhya.com/blog/2018/10/mining-online-reviews-topic-modeling-lda/>, PRATEEK JOSHI

6. Airbnb\_project, <https://github.com/Dima806/Airbnb_project>, Dima806