Dissect Airbnb Star Rating
by Analyzing Hidden Topics from Airbnb Comments

Evelyn Li

Introduction:

As an active Airbnb user, I appreciate the star rating of each listing which has provided me a general understanding of how people feel about this particular listing. As an experienced Airbnb user, I know that by looking at the star rating is not enough to evaluate the details of the home. To gain a full understanding of what people like/dislike about the particular listing, reading the comments under each listing is a step you cannot miss. Comments are not only crucial to guests, but they also contain valuable insights and suggestions for hosts regarding future improvements. However, it depends on the amount of time the listing has existed; the number of comments can be overwhelming to read through one by one. With this understanding in mind, in this project, I focus on using comments to provide insights that can be quickly understood by people without reading through all comments. This project is broken down into two main parts. First is to use unsupervised learning model to extract subtopics that people tend to talk about in comments from the most and least well-performing listings. While we are trying to make the content of the comments a bit more straightforward, this also allows us to understand if there is a difference between good listing reviews and bad listing reviews. Second, we build a model using all the words in the comments to predict the overall rating for each listing. Through understanding these topics, we then can apply these topics to each listing and give both guests and hosts insight into how well the listing is doing to others in San Francisco.

Data:

The data used for this project is from Inside Airbnb, an "independent, non-commercial set of tools and data that allows you to explore how Airbnb is being used in cities around the world." In inside Airbnb, datasets are collected on a monthly basis. For this project, I used data collected for all San Francisco listings on March. Data has two parts; one is the listing file which contains about 4000 listings in San Francisco with all information including hosts' description, amenities offered at the house, and overall review score rating. The other one is the review file which contains all the reviews for each listing in San Francisco. Overall, there were about 300,000 reviews in total. With all the comments and the overall review score rating for each listing, we can build a model to understand the relationship between what people talk about in the comments and the rating for each listing.

Exploratory Data Analysis with LDA:

Part 1, review rating analysis:
To understand what are some of the critical hidden topics in reviews for listings with high and low ratings, the first approach is to separate the comments based on the score. One interesting thing to note is that the majority of the Airbnb ratings has a range from 80 points to 100 points. Very rarely, there is a rating below 80. In the San Francisco dataset, out of three hundred thousand reviews, only 600 reviews are the ones associated with a score of below 80 points. With such observation, separating comments based on 0 to 100 rating scale would not be feasible because we do not have enough data to extract any subtopics. The approach I adopted is to use comments from listing with lower than 80 points of rating score. This way, we then can make sure there is enough information for the model to search upon and extract the subtopics. In comparison to the poor performing ratings, on the other hand, I used comments from listings with 100 points of rating score to understand what are the different topics customers tend to talk about when they express their satisfaction.

Latent Subtopics with Latent Dirichlet Allocation:

Latent Dirichlet Allocation (LDA) is a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in which each item of a group is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document. [1]
In the context of this project, all comment collected from listings after count vectorization is the collection of the text corpora. Each comment is treated as a bag of words of specific size and then is assigned to a topic via the Dirichlet Distribution. With stop words removal, bigram specification, 15 subtopics is extracted from the comments.

As an example, the breakdown of the top 4 topics and the words distribution for comments from listings with 100 points several topics is shown in the table below.

Topic 1 (11.2%)	Topic 2 (9.9%)	Topic 3 (9.5%)	Topic 4(9.3%)
definitely stay	Great location	Walking distance	Quick respond
Highly recommend	Like home	Public transportation	Clean comfortable
Look forward	Great view	Shop restaurants	Great place
Host responsive	Feel like	Short walk	Great host

Table I

The same process is done to comments from listings with less than 80 points, and the top 4 topics distribution is shown in the table below.

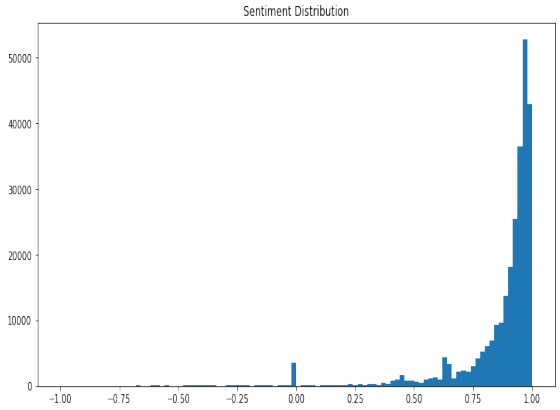
Topics 1 (10.6%)	Topics 2 (9.9%)	Topics 3 (8.9%)	Topics 4 (6.2%)
Great host	Arrival automated	Good location	Homeless people
Location great	Automated posting	Location close	Surrounded homeless
Good experience	Host canceled	Location really	Drug addicts
Good price	Canceled reservation	Room got	Homeless drug

Table II

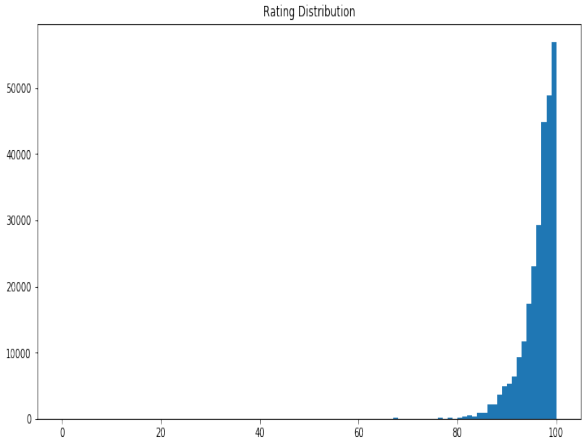
Part II, sentiment analysis:

From looking at the subtopics from the LDA model, words with positive sentiment seem to be a big part of the comments. To dig a little deeper into the feelings behind all the comments, I used a sentiment analysis package from both NLTK and Google Language API to analyze the sentiment for each observation. Due to the inefficiently extended amount of time Google Language API takes, although the result from Google Language API's sentiment analysis seems more accurate, we focus on discussing the result of the sentiment analysis using NLTK package. The NLTK sentiment analysis gives you four parameters to measure the sentiment of the comment. They are, positive, neutral, negative, and compound. Positive, neutral, and negative all range from 0 to 1, where 0 means the least relevant and 1 means the most relevant. While compound range from -1 to 1 where -1 represents absolute negative sentiment, and 1 represents perfect positive sentiment.

After measuring all the sentiment of comments, the next step is to understand what the distribution of the sentiment looks like. After plotting a histogram of the sentiment based on compound score, we see the following distribution:



As you can see, most sentiment are positive and concentrated around 0.8 to 1 which mean that most comments on the listing are expressing guest's satisfaction instead of dissatisfaction. If you take the rating distribution into account, this distribution of the sentiment is following the same shape of the rating scores where more of the ratings are also concentrated above 80 points.



Knowing that most comment's sentiments are positive, it is still crucial for us to understand what kind of words people tend to use to express their satisfaction and joy towards to stay. Therefore, another LDA was performed based on sentiment compound score where positive sentiments are all comments with a compound score higher than 0.5; negative sentiment is comments with a compound score less than -0.3. The topics extracted from analyzing comments based on sentiments gave us a good understanding of what are the keywords people tend to use to express their good/bad sentiment. This finding will be significant for the next step, which is to perform hidden topic analysis for each listing.

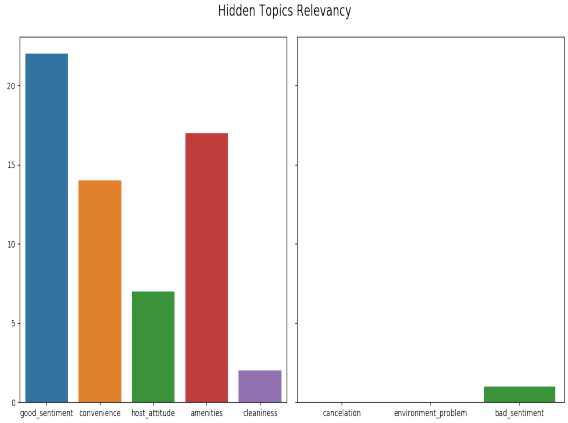
Hidden Topic Analysis:

Based on the unsupervised learning model LDA, we now have a good understanding of what are some of the hidden topics that guests tend to mention when they leave a comment. From both the rating LDA analysis and sentiment analysis, eight key topics will significantly influence the performance of a particular host. These eight topics are the following:

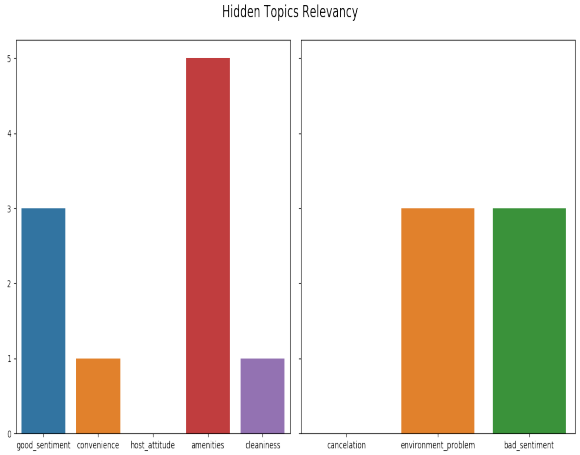
1. Good sentiment
2. Convenience
3. Host a positive attitude
4. Amenities mentioned
5. Cancelation
6. Environment issue
7. Bad sentiment

Each of the topics above contains keywords/phrases that are extracted from the LDA analysis, which will be used to measure individual host performance based on the relevancy of each topic mentioned in their comments.

For example, take the host with id number 27025. This host has a review rating of 100, which is among one of the highest scores. If we take a look at this host's hidden topic relevancy, we can see the distribution as follows:



Let's compare the result of a host who has a lower rating score of 76, this host's hidden topic relevancy looks drastically different from the host with good rating score:



For a listing with a good rating, the hidden topics relate to good sentiments are way more relevant comparing to a poor performing listing. Since this metric is wholly based on comments, both host and guests, and use this metric to evaluate the house. For this particular example, the host can go to the comments with bad sentiments in it and understand why this specific guest had a bad experience and act on it. Also, cleanliness is only

mentioned once for all 30 comments; perhaps this will lead to host understanding that they need to improve on the cleanliness of their property.

Neural Network Prediction Model:

Based on the previous finding of words in hidden topics, we can see that the comments in listings with higher ratings tend to focus on specific issues compared to the comments in listings with lower scores. In this case, we can understand the rating of a particular listing represent the aggregate sentiment of how people feel towards their experience. With this finding, I am interested in building a model where we use words purely from the comments to predict the rating for each listing. The hypothesis is that, if we see a listing with a high rank, and the comments people tend to leave for good rating listings also tend to have a particular topic within them, we might be able to build a model that understands the relationship between the comments and the rating.

The intuition behind this model:

Considering each word in the document as a feature of the target value, we have a corpus that represents different ratings, which will potentially allow us to find the relationship between words and the score.

Similar to sentiment classification analysis, in this model, we have a review score that represents how people feel towards the listings. The score is range from 0 to 100 and is provided by each guest who also had left a comment about this place. Although each listing is dependent on each other, they still share a lot of common topics as we had discovered above.

Result:

The current neural network model has no prediction power. The model successfully picked up the average of all ratings and has a mean absolute error of 5 points and a negative R square, which means that the model is worse than using the mean for prediction. Although the result of the model is not worth interpreting, we can try to understand why the model is performing poorly.

First, most of the reviews are positive and between the range of 80 to 100, and this score is simply someone hitting the screen, and there is no context behind it. From the guest's point of views, a high star rating doesn't mean much in terms of evaluating the quality of the property. Second, the model is only good at picking up reviews with the high rating because the entire dataset is filled with listings with high scores. There are not enough low rating comments for the model to learn and understand what the topics people tend to talk about when giving a bad grade to a listing are. Third, the rating score for this dataset is an average score of all the rating guests had given in the past. We do not have a clear understanding of how each guest has provided their rating when they leave the comments. Unlike a score which you can average over, comments are words and can't be average over. Therefore, using all the comments to predict one averaged number is merely hard to do.

Besides, the negative R square gives us some additional insight into interpreting the relationship between the rating and comments. The key takeaway is that there is no relationship between comments and rating. There is not a consistent pattern of words that lead to a good rating score. This result has proven to us the absolute independent between the rating score and the comments from a particular listing.

The Alternative Model:
sentiment score instead of a star score

I believe the biggest reason why the model has failed to predict the rating of the listing is that there is not enough connection between comments and the rating score itself. The failure of the model also raised another question, how useful is it for both the guest and the host to look at the rating and draw any meaningful conclusion knowing that the score is a "polite" score people give and the comments can say otherwise.
To fill the gap between what the rating represents and what people are people saying about their

experience, I purposed using sentiment analysis score to represent the actual rating for each listing.

There are many sophisticated packages and tools build by professional data scientists to detect the sentiment of each comment. After applying sentiment analysis to each review, we can then average all sentiment scores and transform it into a sentiment star rating. For sentiments that are on the negative side, the system can put a flag on that comment for host and guests to review later.

Future Work:

Based on the result from LDA analysis, there seem to be several underlying topics that guests tend to discuss in the comment they leave for the host. Although both regression models and the neural network have failed to capture this relationship that I believe is there, it does not mean the research should stop here. For future analysis, there are methods I wish to attempt to find this relationship between comments and rating. First, I would try out word embedding. As we all know, language makes sense because words are not independent of each other. How words connect will significantly affect the meaning of the sentence. Therefore, representing these relationships will be a crucial next step in capturing the relationship of positive comments and good rating, vice versa. Tools I consider using are word2vec, embedding layer within the neural network, and so on.

Secondly, extract the subtopics from all comments. If using all the words from all comments doesn't work, next thing we should try to do is to extract out the sentences that are related to the common theme throughout all reviews regardless of housing types and location. This way, the measurement between all listing will be equivalent and thus makes more sense to compare one listing to another.

Conclusion:

Overall, there seems to be an underlying theme of all comments regardless of housing types and location. However, due to the amount of time given to this project and the long process of learning about the relationship between comment and rating, the baseline model built for the prediction purpose did not give us a satisfying result. However, the lack of connection between comments and rating score has provided us with a new perspective which is to think about the reliability of the rating and ask if Airbnb can come up with a more realistic, and honest rating scale for evaluation purpose. The work of research does not stop here. There is more to find and more method to try out.

References:

[1] L. Susan. "Multi-Class Text Classification with LSTM" *Medium*. 9 April, 2019.

[2] D. Blei, A. Ng, and M. Jordan. "Latent Dirichlet Allocation." *Journal of Machine Learning Research*, 3:9931022, January 2003.

[3] D. Cai, Q. Mei, et all. "Modeling Hidden Topics on Document Manifold." Department of Computer Science, University of Illinois. CIKM 2008.

[4] B. Sanjai, M. Fenna, et all. " Sentiment Analysis with Long Short-Term Memory networks." *Semantic Scholar,* 2018.