NLP Based Airbnb Data Analysis and Modeling

11911317 Qin haocheng, 11911325 Liu zeyang, 11911120 Ao zitian, 11911126 Hu hongwei

May 29, 2022

Abstract

We analyzed the Airbnb data set based on NLP. We looked into the data and model it from the perspective of users, hosts, and the company. From the user's point of view, we use linear model to study the factors affecting rental prices, which can be used to determine whether the price is reasonable. From the perspective of a host, we used logistic regression to study how to get a higher rating for their bB and how to become a superhost. From a corporate perspective, we designed a recommendation system against the review data set and tested its performance.

1 Introduction

1.1 Background

With the development of transportation and economy, more and more people would like to travel in their spare time. It is no doubt that hotel reservation is an important problem in traveling, and more and more people will try to solve this problem online. Among lots of the the online booking application, Airbnb is a famous global platform for booking short stays around the world. There are a great number of data being generated in the process of booking trades such as host data, listing data. It is interesting and meaningful for us to reasearch on the data.

1.2 Data description

We are given a dataset which contains reviews data, future 365 days price data and detailed listings data. For the listing data, it consists of three parts. One is the host data, there records detailed information of the host who host the listing. Another one is information about listing itself, such the price, rooms number and so on. The last one would be regarded as the historical information, there is a lot of comments and ratings left by historical users. The detailed variable information is also shown in the following figure.

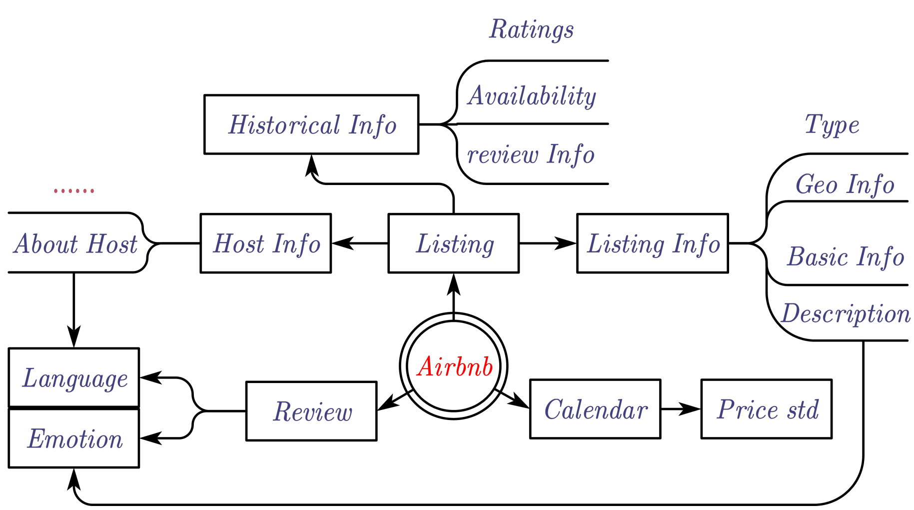


Figure 1: The overall of the airbnb data variables.

2 Exploratory data analysis

2.1 Data Procession

For the text data Review, Host_about, Description, we should use NLP methods for embedding to get more features for us to analysis, or there will be a significant waste of imformation.

Generally, we cut the reviews into words and drop out the meaningless ones. For the word vector, we then can use a pretrained LSTM [1] model to learn its emotion score, which we will not introduce here; Also we can use the word vector to detect it's language, we use regular expression for Chinese, and then search in wordlist for English if it's not Chinese. The reason we give a prior to Chinese is that'a a chinese international city [2].

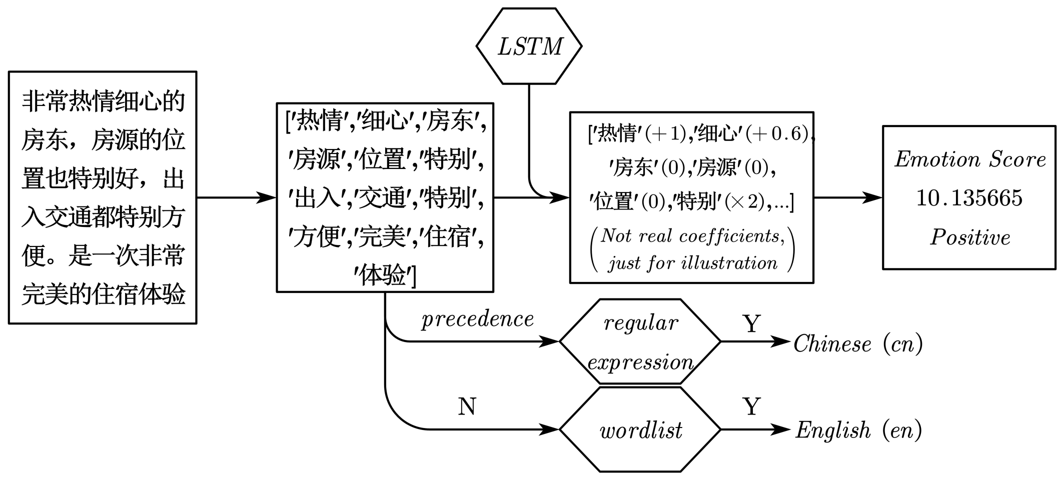


Figure 2: NLP steps

We then scale it test the mean of scores for English reviews Chinese rweviews. It's interesting that we chinese people tend to give a more positive review while the western foreiner seems much more strict. For this, we should scale the score with in each language.

MEANS PROCEDURE

分析变量 : score						
ch_en	观测数	均值	标准差	标准误差	t 值	Pr > |t|
cn	104390	2.9769214	3.4044170	0.0105369	282.52	<.0001
en	14263	2.3528602	2.5268157	0.0211577	111.21	<.0001

Figure 3: T test result by SAS

We also did this procedure to the feature "host_about", for which we want to know wether the self introduction will have some influence; And for "description", for which we want to know whether the description will have an effect. Also we can derive keywords like "subway", "Disney", etc.

For the review score, we can take the mean for each listing as "listing_review_score", and for each kind of scores for listing, we can take the mean for the score of each host.

As for another dataset, "Calendar", we also want to derive some imformation from it, and our strategy is to get the variance of the price for each listing.

So finally, we add 11 more variables into the dataset (another is length of the list of amenities), and we found some of them are actually significant. We also dropped some unimportant features, like "listing_url", "last_scraped" and "name", then take the interception of the listing without any miss value.

2.2 EDA

We would like to make an exploratory data analysis on all data, so we can get a better understanding of the data.

- **For price data** The price data records the price trends in future 365 days within 2875 different listings. Looking at the data trend, it is easier to find that most of the listings prices would not change continually. We counted the number of the price change time in future 365 days for every listings as the change frequency. Then divide listings into different groups by the follow rules table.

Frequency number	0	0∼5	5∼10	10∼ 50	> 50
Group	never	rarely	sometimes	usually	often

Table 1: Group rules

The result would be shown in the follow pie plot, more than 70 percent listings price change times are less than 5. And it's also tell us the value of the price change times, we would like to use the variance of the price change data as the price change factor of every listings.

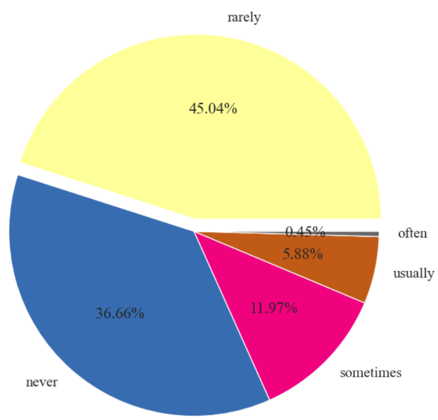


Figure 4: Pie plot of the price change times.

- **For reviews data** The reviews data contains lots of comment text by the historical customers. We want to do two things within this text data, on the one hand, we want to get the word frequency statistic at the whole comment text, the result is shown at the figure bellow, from the word cloud figure, it is easy to see that the "Fangdong" may be an important factor for customers, and other factors like "Fangjian","Ganjing" are also show great importance.

  On the other hand, we want to get the emotional scores behind the texts. Here we divide the texts into two groups by the language the comment was written. After we get the scores of all the texts, we find an interesting little result which is that the emotional scores are related to the language used. We test this result by t-test, the p value is less than 0.05.

- **For listing data** We would do the exploratory data analysis of the listing data at the follow three parts.
  - **The host portrait** We select all the variables related with the "host", such as host id, host response time.

    From the pie plot from the host data, a subset of the listing data, which contains 7742 different hosts in the listings.csv. And from the pie charts, we could see many useful information about hosts.
    * There are only one in four hosts could be identified as super host by airbnb.
    * Half of the hosts only host one listing, and a little part of the hosts host more than 50 listings.
    * Almost all hosts have done id verification.
    * For the respond time, there exits a large number of NA. we would look them as "Unknown".

Figure 5: Word cloud of the whole comments

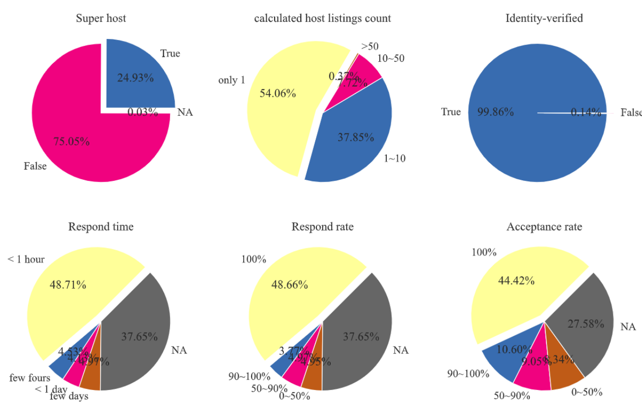


Figure 6: Host portrait

* Half of the hosts respond rate and the acceptance rate are more than 90 percents. we would fill the NA with their mean value.

We also converted "host since" to the number of years he or she registered as host. We could see that most host registered years are less than 6 years, and a few host registered to be host more than 10 years from the follow figure.

– **The listing itself** As can be seen from the heat map, most of the listings are concentrated in "Pudong", and there are not many listings in other areas. It also shows a power law distribution in the price distribution in the listings in the Shanghai. From this two charts, we could know that the location and price of the listings concentrated in some place or interval.

– **The historical information** It is no doubt that the most valuable historical information is the scores information. We draw a radar map of the scores information as follow. The median of all the scores are higher than the mean of all the scores commonly.

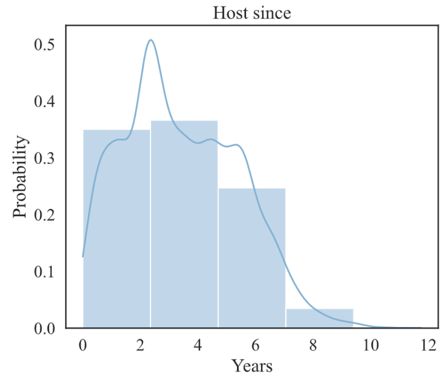


Figure 7: Host since years

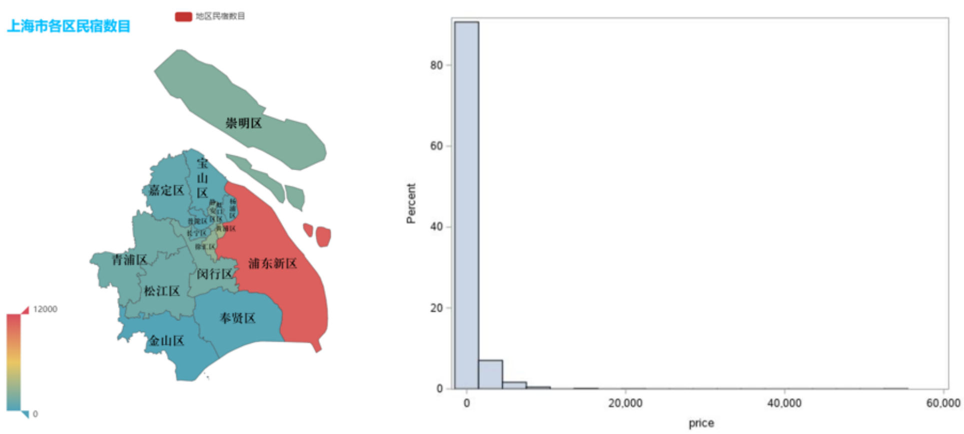


Figure 8: Heat map and hist plot of the listings in Shanghai

3 Discussion

Now we have a rough understanding of the whole data. After discussion within our group, we came up with three modeling problems.

- For host, which factors are important to be make the host be regarded as super host.
- For customers, which factors are important to effect the price of the listings.
- For airbnb, how to recommend suitable housing to customers.

We would answer these three questions in the next statistical modeling part.

4 Statistical modeling

4.1 The model about host

In this section we want to know which factor will influence the level of host. The first question is the index of host's level. From the origin website of raw data, we could know that "superhost" is an honor for all hosts which means that the host is in a high level. We want to find the variables that have an influence on "superhost" to help hosts to find a quick way to make them be a popular host and they can get more money from renting a house. Besides of the given norm named "superhost", we could find another norm to analysis how to make your career of host well.

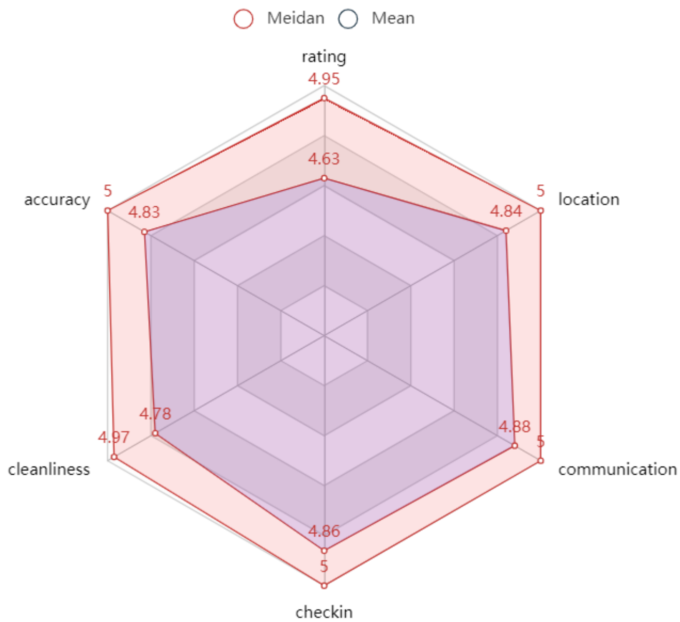


Figure 9: radar map of all the scores

4.1.1 Data set cleaning and the creation of new data set

The raw data set containing too many variables which are not relate to host but relate to house, we should pick the variables we need and combine with the score of text analysis. From raw data set, we pick following variables into new data as figure following, as to the score from text analysis, we pick two variables of all whose name is "host readme score" and "host house score". To find out that whether language has influence on the evaluating of host, we add "host lang" which expresses the language spoken by host. Upper is the

Field	Type	Calculated	Description
host_id	integer		Airbnb's unique identifier for the host/user
host_url	text	y	The Airbnb page for the host
host_name	text		Name of the host. Usually just the first name(s).
host_since	date		The date the host/user was created. For hosts that are Airbnb guests this
host_about	text		Description about the host
host_response_time	text		The time lag the host promise to respond messages.
host_response_rate	numeric		The rate at which the host respond messages.
host_acceptance_rate	numeric		That rate at which the host accepts booking requests.
host_is_superhost	boolean [t=true; f=false]		Whether the host is a superhost
host_total_listings_count	integer		The number of listings the host has (per Airbnb calculations)
host_identity_verified	boolean [t=true; f=false]		Whether the host's identity has been verified
host_readme_score	**numeric**		**The score about the positivity of self description of host**
host_house_score	**numeric**		**The score about the average score of each house from a host**
host_lang	**string**		**Language spoken by the host**

Figure 10: Variables from raw data set

variables we will use from the raw data set, the all have correlation about host. the two variables about score will be added behind these variables. They are all variables we will use to construct our model. To get a data set which contains all variables as Figure, we should merge them in accordance with "host id", they are totally 7742 hosts so our data set's dimension is 13×7742. After the combination of the extra 3 variables, they are many missing value in these columns.

4.1.2 Missing value handling

If there is missing value whose type is numeric, we choose the average value of all observations about this variable. But for "host readme score" and "host house score",they comes from emotion analysis about text, "host readme score" means the score of host's self description and "host house score" means the score about the average score of each house from a host. We could not simply use average value of all observation, because there will be a large difference between different hosts and it will cause a large error about our model. So we could only delete the observations with missing value of these two variables. After dealing with missing value, we only have 594 observations remaining with 13 variables. Although we lost too many information about raw data set, the result data set could be accepted because the number of observation is more than 20 times of number of variables.

4.1.3 Data overview

After the deletion, we get a data set containing 594 observations. We want to check the roughly distribution of them for dealing with them better. Here are some histograms of them.

For "host response time" and "host response rate", the meaning of these two variables is same which describes the average frequency of a host reply their message. So we remove "host response rate" and simplify "host response rate",in our raw data, they are 4 categorical types in this variable, we can draw a histogram about it. We can see that "within an hour" take an important part of it, so we can renew the

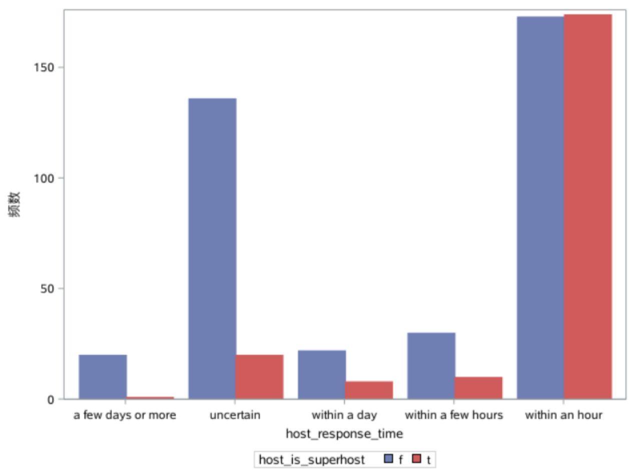


Figure 11: Histogram of "host response time"

label of it as "quick" means that the host could reply message quickly. Set the other type "slow" except "uncertain", by the way, "uncertain" means missing value but we have given them a label so we need not to remove them from our data set. We can do the same thing about "host acceptance rate" because there are a lot of observations' value is 1, they are continuous variables so we set 1 as "always", set 0.5-1 as "usually", set 0-0.5 as "sometimes".For "host since", it represents the date when this host be a host. We transfer the date represent year between today and the day that he became a host. After the arrangement, the model has been simplified, and we could use these variables to do our model construction.

4.1.4 Model construction of super host

They are two things we want to do. First of them, we want to construct a great model that could fit data best. The first thing is to choose a fitted model which could divide response variable into two groups. We choose logistic model, because our response variable is "host is superhost", value of this variable is 't' and 'f', in our model, we have some categorical variables, but we still could apply logistic regression on this question.

$$log(\frac{p_i}{1-p_i}) = \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_{13} x_{i13}$$

p_i is the probability of i-th observation is super host, while $p_i > 0.5$ we will think that this observation could be identified as super host. We apply logistic regression on the data which contains 594 observations. The result could be showed in SAS studio. The shown model has been simplified which means that we have not use all 13 variables in this logistic regression. The variables been used are all significant to response variable which means that p-value is not so large. The variables with insignificant level will be delete. Result of this logistic regression analysis could be shown as follow: We can explain the variables we have selected:

3 型效应分析			
效应	自由度	Wald 卡方	Pr > 卡方
host_since	1	2.8480	0.0915
host_response_time	2	55.6981	<.0001
host_acceptance_rate	2	3.1496	0.2071
host_readme_score	1	5.4629	0.0194

Figure 12: Logistic regression model in SAS

最大似然估计分析						
参数		自由度	估计	标准误差	Wald 卡方	Pr > 卡方
Intercept		1	-2.1097	0.4349	23.5304	<.0001
host_since		1	0.1232	0.0730	2.8480	0.0915
host_response_time	quick	1	1.0434	0.1454	51.4832	<.0001
host_response_time	slow	1	-0.2321	0.1985	1.3674	0.2423
host_acceptance_rate	always	1	0.1912	0.1806	1.1210	0.2897
host_acceptance_rate	someti	1	-0.4950	0.2968	2.7815	0.0954
host_readme_score		1	0.0705	0.0302	5.4629	0.0194

Figure 13: Logistic regression model in SAS

- ”host since”: because we transfer the date when a host became a host into the year from today, so the earlier he became a host, the larger ”host since” will be. We can find that the estimate of β is 0.1232 means probability will increase when ”host since” increase, so the sooner one becomes a host, the more probability he will become a super host.
- ”host response time”: we can see that there are 4 type of categories, if ”host response time” is ”quick”, the estimate is 1.0434, means that the quicker you reply the message from guests, the more excellent you are. When label is ”slow”, the estimate is -0.2321, means you will less likely be a super host. But it also might has no influence on your way to be a super host because the p-value is larger than 0.2, we can not say that β has a large probability that it is larger than 0.
- ”host acceptance rate”:Two labels estimate conforms to our daily cognition,because if a host is more acceptance, it means he/she has more probability to become a super host, but we can not say that a host will be absolutely a super host if he/she is more acceptance because the p-value is not less enough to convince us.
- ”host readme score”: it is an interesting conclusion because if a self description from a host could get a high emotion score ,he will have more probability to get the honor, the p-value is less enough to make us believe that the influence exists. It inspires all hosts to make a good self-introduction.

The Wald confidence interval will be shown in the appendix. The main model about getting the independent variables complete, we have to test the accuracy and predictive power of our model.

4.1.5 Performance of logistic regression model

From the logistic regression model, we have selected 4 variables that have significant influence on the response variable, and in reality, we can also find a good explanation on them. We want to test the accuracy of our model, we could divide these 594 observations into a train set and a test set. We set $\frac{3}{4}$ of them into train set and $\frac{1}{4}$ of them into test set. The accuracy will be shown as the following figure.We can see that the accuracy is 0.783, the accuracy seems to a big number. The prediction performance of our model is good.

4.2 Linear models about price

In this section we want to analysis the model of price of listings per night based on the interaction of the three data sets.

4.2.1 Data procession

- **variable procession** There are total 2632 listings in the interaction of the three data sets with 60 variables. First we delete "listing url", "name", "host id", "host url", "host name", "picture url", "longitude", "latitude", "property type" and "host total listings count". Here we should note that we delete "host total listings count" because there are contradictions between "calculated host listings count" and "host total listings count" and when we compared both data with the real data in the website we found "calculated host listings count" was more in line with reality.

  Then we replace "host about" by the "hostscore", a numeric variable with range $\mathbb{R}$, and "hostlang", a categorical variable, which mean the emotional score of "host about" from emotional analysis and the language of "host about" respectively. Similarly we replace "description" by "desscore" and "deslang". And replace "neighbourhood overview" by "restaurant", "scenic", "subway" and "shopping", four binary variables with 1 meaning there is description of the corresponding things in their "neighbourhood overview". Moreover we delete"last scraped", "host since", "first review", and "last review". Instead, we add "last vacant duration", "age", "cumulation duration". The calculation formula is as follows:

$$\begin{aligned} last\ vacant\ duration &= last\ scraped - last\ review \\ cumulation\ duration &= first\ review - host\ since \\ age &= last\ scraped - host\ since \end{aligned}$$

  Besides we delete "amenities" and replaced it simply by the number of amenities mentioned in it. We broke "bathroooms text" into three variables: "bathrooms shared", "bathrooms half" and "bathrooms num". "bathrooms shared" is a category variables divided into three categories means the bathrooms are private or shared or neither. "bathrooms half" is a category variable means whether the bathrooms are half. And "bathrooms num" specifies the number of the bathrooms.

  At last we add "review avg" and "price std". "review avg" means the average emotional scores of past reviews of the listings. "price std" means the standard error of the past price of each listings

- **missing value procession** Due to the too large number of variables and too little information about the price, so we simply take the mean of the variable to replace the missing value except with a few exception. For the extreme value of "price", we restrict that $price < 5000$. Based on this limit, we delete the data when information of "bedrooms", "beds" or "bathrooms shared" are missing.

- **BoxCox** For the distribution of "price", we do BoxCox on "price" and we choose $\lambda = -0.3$ empirically and the result is shown in Fig.14.

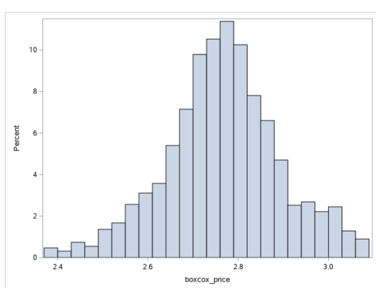
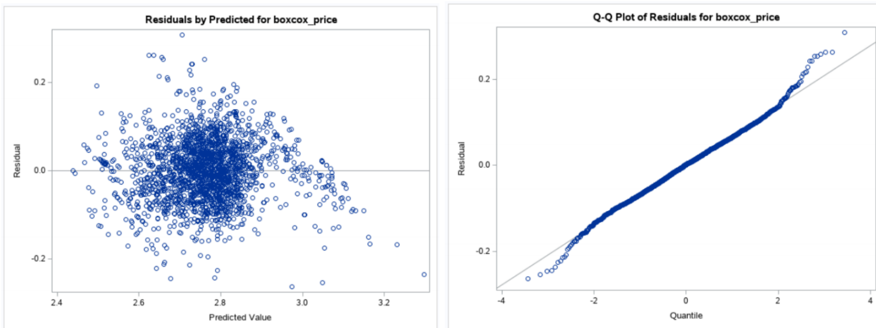


Figure 14: BoxCox

4.2.2 Variable selection

First we do a rough linear regression about Boxcox transformed "price" among all the variables except "id" and here in Fig.15 we can find the assumptions of normality and homoscedasticity cannot be verified, but we can also see that the R^2 is relatively high.



R-Square	Coeff Var	Root MSE	boxcox_price Mean
0.689911	2.559161	0.070709	2.762991

Figure 15: Rough Linear Regression

The GLMSELECT Procedure

LASSO Selection Summary

Step	Effect Entered	Effect Removed	Number Effects In	AIC
0	Intercept		1	-6673.4128
1	accommodates		2	-7026.8911
2	bathrooms_shared_0		3	-7185.5214
3	room_type_Entire home/apt		4	-7865.4943
4	room_type_Shared room		5	-8152.8941
5	price_std		6	-8402.4776
6	district_Jiading		7	-8472.6349
7	amenities_num		8	-8509.3472
8	review_scores_location		9	-8517.9446
9	district_Chongming		10	-8540.5838
10	district_Songjiang		11	-8590.5179
11	hostscore		12	-8592.5090*

* Optimal Value of Criterion

Selection stopped at a local minimum of the AIC criterion.

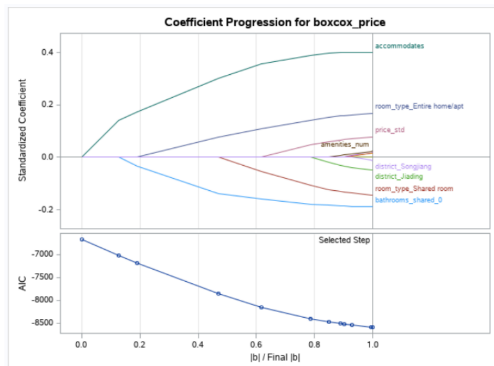


Figure 16: The Lasso

Then we perform variable selection by the Lasso and the criteria to choose the model is AIC. Here is the output in Fig.16

So our model will choose five continuous variables: "accommodates", "hostscore", "amenities", "review score location" and "price std", and three categorical variable: "bathrooms shared" (where we only consider if the bathrooms is shared or not), "room type" (where we only consider if the room is entire home/apt or shared room or not) and "neighbourhood" (where we only consider if the district is Jiading or Songjiang or Chongming or not).

4.2.3 Variable transformation

By the scatter plot of each variables shown vs BoxCox transformed "price", we use the transformation as shown below:

$$\begin{aligned} log\ accommodates &= log(accommodates); \\ log\ price\ std &= log((price\ std) + 0.001); \\ e\ review\ scores\ location &= exp(review\ scores\ location); \\ log\ hostscore &= log(hostscore + 3.9); \end{aligned}$$

After variable transformation, from Fig.17 that there significant difference in regression if "price std" is equal to 0, so in latter modeling section we will break this data set into two parts based on whether "price std" is equal to 0 and we will do linear regression respectively.

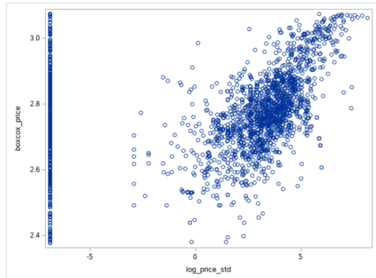


Figure 17: Box-Cox price vs log price std

4.2.4 Interaction Detection

In this section we will indicate whether there are interaction effects between variables. We will use interaction plot to see if the effects exist between a continuous variable and a categorical variable. For interaction effects between continuous variables or between categorical variables, we just add all the parts in to the model to see if it is significant. Fig.18 list interaction plots with significant interaction.

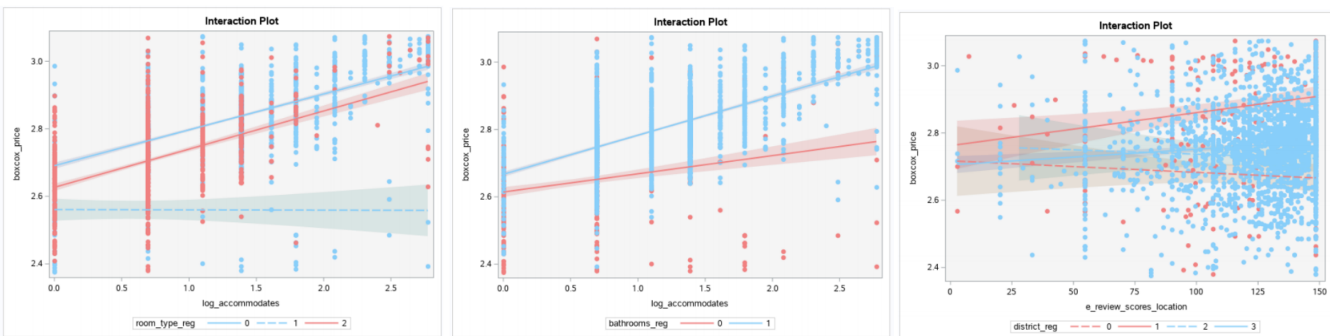


Figure 18: Interaction Plot

4.2.5 Modeling with "price std" not equal to 0

- **Model specifying** Firstly we added all the interaction terms into the model to regress and then deleted all the non-significant interaction terms to get the proposed model:

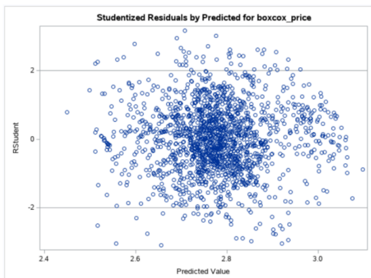
$$\begin{aligned} boxcox\ price = & \ log\ accommodates + room\ type + bathrooms + log\ price\ std \\ & + amenities\ num + e\ review\ scores\ location + log\ hostscore + district \\ & + (log\ accommodates) * (room\ type) + (log\ accommodates) * bathrooms \\ & + (e\ review\ scores\ location) * district + bathrooms * district \\ & + (log\ accommodates) * (log\ price\ std) + (log\ accommodates) * (log\ hostscore) \\ & + (log\ price\ std) * (log\ hostscore) + (log\ accommodates) * (e\ review\ scores\ location) \\ & + (log\ price\ std) * (amenities\ num) \end{aligned}$$

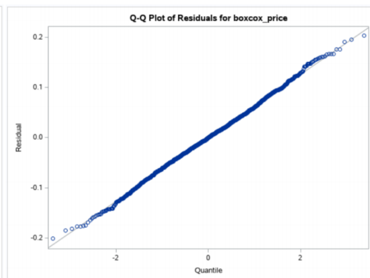
- **Model diagnosis** To erase the heteroscedasticity to some extend, we first perform weighted least square estimation (WLS). Secondly, based on this estimation, we extracted the outlier by restricting $Studentized\ Residuals,\ r^{stu} > 3$. There are total 11 outliers in this model. After checking these outliers one by one, we found that outliers shown in Table.2 either have extreme prices, or prices that are quite different from today's prices. So it is reasonable for us to delete these value.

  After deleting, we performed WLS based on the deleted data and got the beautiful results shown in Fig.19.

id	price_data	price_now
12595458	776	776
13335349	2489	2288
14845272	120	249
15212733	1880	1880
19479033	68	91
19691965	969	1288
22368417	270	343
22807599	69	69
24973155	112	106
25014281	4714	3000
25249046	300	429

Table 2: Outliers





The GLM Procedure

Dependent Variable: boxcox_price

Weight: w10_1

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	25	7961.54006	318.46160	195.08	<.0001
Error	1646	2686.98186	1.63243		
Corrected Total	1671	10648.52191			

R-Square	Coeff Var	Root MSE	boxcox_price Mean
0.747666	45.80865	1.277666	2.789138

Tests for Normality				
Test	Statistic		p Value	
Shapiro-Wilk	W	0.99869	Pr < W	0.2413
Kolmogorov-Smirnov	D	0.016031	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.087095	Pr > W-Sq	0.1728
Anderson-Darling	A-Sq	0.562825	Pr > A-Sq	0.1485

Figure 19: Linear Regression Result

4.2.6 Modeling with "price std" equal to 0

However, when we run the same procedure on the data with "price std" equal to 0 (the only difference is the model will not cover the variable "price std"), the model did not fit well (with $R^2 < 55\%$ and did not pass the normality test). It is reasonable and acceptable since when standard error equal to 0, it is a natural idea to assume that the price will not change in the following period of time.

4.3 Score analysis of house

The data set of this section is from the front section.The only difference is that we divide the "review scores rating" into 2 groups, one of which is larger than 4.8 we named it "t", the other label is "review scores rating" less than 4.8 whose name is "f". We choose this standard because airbnb think the house when "review scores rating" larger than 4.8 is a good house. We apply logistic regression on this data set, because we put review scores rating as a dummy variable. After deleting the insignificant variables, we could get a regression model whose variables are all significant. Result could be shown as following picture. In this regression model, we set score larger than 4.8 is "1" so the probability represents the probability that the score is larger than 4.8. All variables are significant, we can make some conclusion about what we see. For example, "price" has significant influence on the judge of score, the score will increase when price increasing, it is a obviously conclusion because higher price means greater serve, but the coefficient is not very large because there are many host only increase their price but the serve can not match the high price, so the score will not increase very large. We can check the positive and negative of β_i to judge the effect of variable i to the response variable "score".

最大似然估计分析					
参数	自由度	估计	标准误差	Wald 卡方	Pr > 卡方
Intercept	1	-91.3753	4.4751	416.9115	<.0001
host_total_listings_	1	-0.0181	0.00570	10.1161	0.0015
price	1	0.000232	0.000069	11.2206	0.0008
number_of_reviews_lt	1	0.0280	0.00836	11.2139	0.0008
review_scores_accura	1	4.7852	0.6074	62.0739	<.0001
review_scores_cleanl	1	6.8839	0.5093	182.6821	<.0001
review_scores_checki	1	7.2382	0.8074	80.3585	<.0001

Figure 20: Logit regression

5 Recommendation System

In fact, we can regard the dataset "review" as a perfect weighted bipartite graph. Each user chooses his favorite B & B with the emotional score as the weight. This is a very classic data structure. We can analyze each user and find other accommodation that he may be interested in. This is the role of the recommendation system.

Here we partially use and improve a classic recommendation system with good results. [3]

The idea of this system is very simple. If two users often stay in the same B & B, they can be considered to have the same preference for B & B. we can recommend the B & B that one user likes to the other with a higher weight.

It is very simple to do this. After user a has checked in a B & B, we set the weight of the B & B to 1, and those who have not checked in to 0. (in our model, it is set to 1+ emotional score, and the higher the emotional score means that they prefer a certain type of B & B), so the value can be evenly distributed to each user who has checked in the B & B through reverse propagation, and a value similar to "similarity" can be obtained. Naturally, when we propagate the similarity to the B & B, we get the preference value of the first-order iteration. This process can be carried out continuously, but according to the experiment, the effect of two transmissions is often the best, which is also our choice.

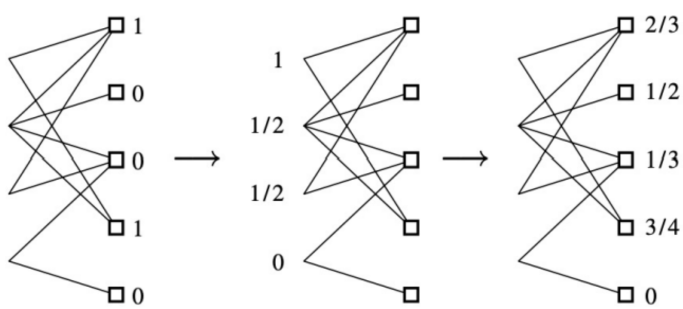


Figure 21: The overall of the airbnb data variables.

For the sake of calculation, we only take a subset of the original data set when validating the model. On this subset, we hide some edges as the test set. We use this system for the training set to test whether we can recommend the B & B on the test set where he originally stayed.

The following is the result of a user. One of his check-in records was covered up, but our model found it well. And finally our system detected it.

It is too complicated to analyze the results per user, so we won't do it here, but at least our model is reasonable and effective. And it runs on SAS.

listing	trainset	groundtruth	predictscore
4146179	1	1	3.120896
16862460	1	1	3.120896
18721703	1	1	3.120896
8608503	1	1	3.120896
25239905	0	0	0.555556
20262903	0	0	0.555556
8007495	0	0	0.555556
19864661	0	0	0.555556
13482023	0	0	0.311111
12573681	0	1	0.277778
18348042	0	0	0.066667
10031709	0	0	0.066667
24169044	0	0	0.066667
6221854	0	0	0.066667
22580681	0	0	0
19530841	0	0	0
22232182	0	0	0
22073511	0	0	0
21928607	0	0	0
18812186	0	0	0

Figure 22: The overall of the airbnb data variables.

6 Conclusion

Many people have done relevant analysis on such data sets, but we have achieved some better results. In terms of superhost and scoring, there is not much work involved, and the explanations are not as good as ours. In terms of price forecasting, we have got a better linear model with normality and homoscedasticity satisfied well. On the other hand, in the related work, except for the group using the deep learning model, there is no group using natural language processing. In addition, we also creatively simulated the recommendation system using SAS, and achieved good results.

References

[1] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[2] h. https://github.com/redtreeai/easy_text_emotion/tree/master/emotion_dict/cn. y. . . redtreeai, title = Easy text emotion.

[3] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang, "Bipartite network projection and personal recommendation," *Physical review E*, vol. 76, no. 4, p. 046115, 2007.

A Appendix

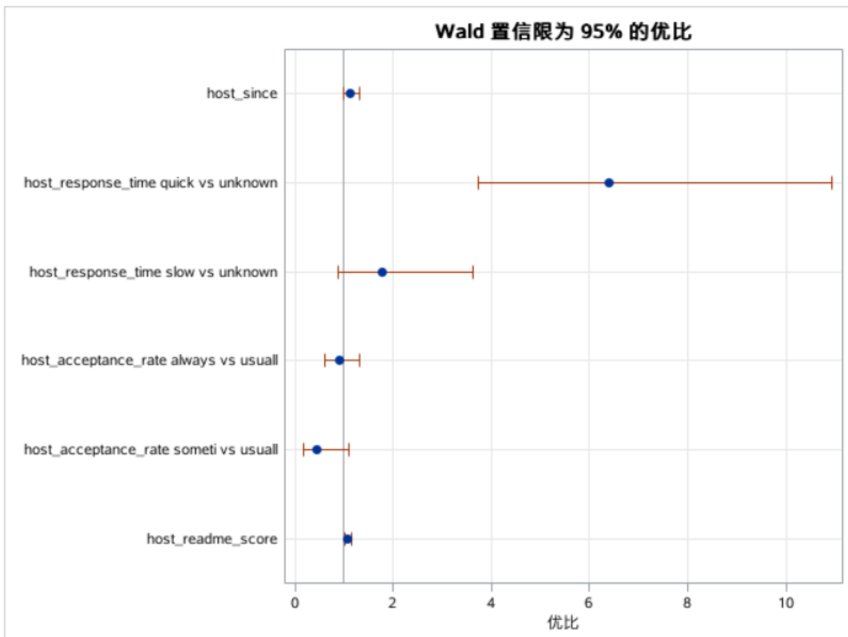


Figure 23: Wald confidence interval of each variable