Renting Prices Analyses in Big Cities of China

Team Members:

Zhang Yiming (1801212981) Yu Xihang(1801212965) Wang Bingquan (1801212932) Liu Anyi (1801212888)

Zhang Yiming is responsible for data crawling and storage; Yu Xihang is responsible for geographic information; Wang Bingquan is responsible for data visualization and feature engineering; Liu Anyi is responsible for model.

1. Background and motivation

1.1 Background and motivation

With the house price in China growing rapidly, more and more young people chose renting as the long-term solution for housing. By the end of 2018, the total number of renters in China reached 190 million, and the market size reached more than 1 trillion yuan.

According to the data of US Census Bureau and Zillow, the proportion of rental households will increase with the increase of Housing Price-to-Income Ratio. During 2008 to 2015, as the housing Price-to-Income Ratio growing from 110 to 150 in America, the proportion of rental families in America get to 37% from 31% [1] (*Picture* 1). We have reason to believe that this trend will also happen in China.

In fact, according to a forecast of Yinhe Security in 2018, domestic rental population will reach 270 million, and the market size will exceed 4.6 trillion [2]. This makes it very meaningful to study rental behavior in China, which will help the government to make reasonable policies to meet the housing needs of young people under this trend.

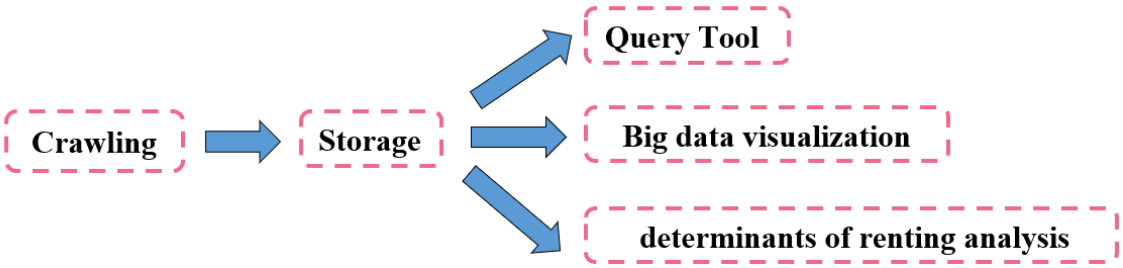
Apart from the rental behavior, we are very interested in the determinants of price of renting. Price-to-rent ratio in China has long been below 2% in the first-tier cities from 2013 (*Picture* 2), however the ratio in America is more than 4%. This fact makes us wonder how renting prices are determined in China.

To sum up, our motivation is to analysis the behavior of renting in big cities in China and find the determinants of the renting price

1.2 Goals and Task Breakdown

Based on our motivation, we divided our project into three processing goals:

Firstly, we want to use scrapy framework in python to crawl the renting information on Anjuke.com and store them in suitable forms. At the same time, we can provide users a search tool to meet accurate renting needs such as location, room type, distance to commercial districts. Secondly, we are going to integrate the information and show them in form of thermodynamic pictures with the support of AMAP API. At last, we want to utilize AMAP API to collect geographic information as much as possible such as the distance to major hospitals or primary schools and combine them with the data crawled from Anjuke to build a model to explain the generation of house renting price. With the help of the model, we want to be able to predict the changes of reasonable renting price of a certain district when the economy expanding and construction of new facilities



The workflow of our project is shown in the *Picture* 3(Appendix 1).

2. Data crawling and storage

2.1 Data Crawling

We use ***crawl.py*** to crawl the renting information of Beijing, Guangzhou and Shenzhen from Anjuke. The website provides the basic information such as price, area, location, floor, elevator, window orientation. We collect all of them use the function

get_item_info in ***crawl.py.***

There is one thing need to be concerned about is that Anjuke use font encryption to anti crawling, when means the font file is encrypted by Base64 which is a group of binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation. The mapping order in the font encryption file is not constant and will change each time when the page is refreshed.

However, if the website wants to display on the page, it still needs to import the font package, so we just need to download the font package and transform the corresponding relationship to get the correct content. To solve this problem, we import base64 library of python and get the base64 code by function ***decode*** in ***Decode.py*** based on the method on CSDN [4].

The data we collect using ***crawl.py*** is shown in the *Table* 1(Appendix 2).

2.2 Data Storage——MongoDB

After we can collect data, we need a suitable database to storage them. We hesitated between the MySQL and MongoDB which represent the SQL database and NoSQL database respectively. After our detailed analysis of our demand, we choose MongoDB, the reasons are below [3]:

Firstly, at the beginning of the project, the specific format of data cannot be clearly defined, we don't know how many data we need finally considering that we need to keep adding new content to our model. In the traditional relational database, a count type operation will lock the data set to ensure more accurate in the "current" situation. However, when data is constantly updated and growing, this "more accurate" guarantee has little significance and will cause great delay on the contrary.

Secondly, MongoDB store hot data in physical memory makes the reading and writing of hot data very fast. It's very essential considering that we need keeping read the location information in the database to get the longitude and latitude of every house with the help of AMAP API and write the result into the database. MongoDB is more efficient under this circumstance.

Thirdly, we need to use AMAP API to collect the geographic information. The return value of AMAP API is Json which is much easier to use in MongoDB than MySQL.

The data storage example is illustrated in picture 4 in Appendix 2.

2.3 Geographic information

Apart from the renting information, geographic information is also important. We need to consider the distance between renting house and service infrastructures in the given city. The first step is to get the coordinate of each type's location in this city. There are five types of location, house location, hospital location, metro station location, school location and shopping mall location. The distance information can be calculated from the coordinate of house location and the other four types' location. The name and address of the other four types' location is also important. There are two important

The POI function provided by Amap can be used to get the coordinates. There are two functions – Pinpoint Searching and Key-words Searching. The Pinpoint Searching function ***get_coordinate*** in ***metro_station.ipynb*** and ***hospital_city.ipynb*** returns coordinates by given the city and location name. It can be used to generate the house location coordinate by inputting house address which can be crawled from the website. The query provided by Amap and crawling command is the method to get the coordinate which is shown in the *Picture* 5 (Appendix 3). The other function is Key-words Searching function ***get_poi_page*** in ***app.py***. It returns a list of all geographic information in different locations by given the city district postcode and location type which is key-word inputted. The input is city district and location type. The output is JSON geographic data, including coordinate of each location in given district which is shown in the *Picture* 6. (Appendix 3).

There are two problems in using the Key-word searching. In shopping mall location, the problem is there are too much address in a single city, and the shopping mall has a pattern that several malls/markets are clustered in a same location. The solution is to drop duplicated clustered malls/markets which distances are less than 1km. In hospital location, the problem is different. There is no obvious clustered pattern in hospital. The problem is that Amap gets a lot of small clinics which we do not need. The solution is to crawl hospital names from another website named A-hospital, then use Pinpoint Searching function to generate coordinates.

3. Data Visualization

When we get the renting houses information with longitude and latitude, we could draw a scatter plot in the map to see the distribution of these renting houses. This scatter plot map could help us find some relationships visually. We can use AMAP API to plot the location of every renting houses in the map with html language which is implemented by the html files in data

visualization file. The distribution is shown in the *Picture* 4 (Appendix 3).

We can see that they are focus on some certain areas like Xi Li which are residence communities. What's more, we want to know which factors can affect the renting prices and This picture also plot the prices distribution. The color represents the price; it is deeper with higher prices. The prices are separated into 8 groups and the numbers in each group are same. If we classify them by price percentage, it may cause that some groups have very large size and some groups just have several samples, which affects the effectiveness of price presenting. We can see that the differences between price levels of different district are not quite large. However, the prices seem to be higher in dense area. This may because the demand in these places are higher which results higher prices and more supply. We also plot the distribution of metro stations, hospitals, shopping malls and schools with AMAP API in *Picture* 8(Appendix 4) .

4. Feature Engineering

The *Table* 2(Appendix 5) shows the renting information we have. The address contains the district, sub-district, road and community name. We cannot use the data in the table to input our model directly and we need to convert them into factors. For the room type, we separate them into two factors which are the number of bedrooms and living rooms. The room area and the building height could be use directly. The room height, renting type, room face, and district could be convert to dummy variables. To prevent the multicollinearity, one category in each dummy variable is dropped. For those services facilities, we calculate the distance between them and the renting houses by longitude and latitude. We use the distance from the nearest services as factors. Also, we use the number of services facilities within 1 kilometer as another factor. This is done by the function ***feature*** in the file ***feature_engineering.py*** Now we have totally 33 features in Shen Zhen which is shown in the *Table* 3(Appendix 5) and we input all of them in our model. The model will find out which variables are redundant.

5. Model

After factor engineering, we use OLS regression, Ridge regression, LASSO regression and PCA to test to train the data of ***Shenzhen, Beijing*** and ***Guangzhou*** in the file ***SZ.py, BJ.py, GZ.py*** respectively.

5.1 OLS regression

The results of OLS regression are as follow:

	Shenzhen	Beijing	Guangzhou
R-squared	0.858	0.606	0.743
Adj. R-squared	0.856	0.601	0.736

The coefficients and their p-values are shown in the *Table* 4(Appendix 6). Setting the significance level equals 0.05, those variable in red are insignificant in our model. According to the table, we can find that the factors influencing the rent vary on cities, indicating that people in different cities have different preferences.

As for Shenzhen, the adjusted R-Squared has reached 0.856 and we find the following five rules based on the results. Firstly, the rent is positively related to the number of bedrooms. Generally, the joint rent is more common than the whole rent and too much living room is considered as a waste. Also, the rent is positively related to the total height of the building. To some extent, the total height of the building can represent the quality of the house and we can say that the apartment in high rise is much better than the bungalow in villages in cities. Secondly, the more metro, CBD, hospital and school within 1 km and the closer to those infrastructures, the higher the rent. What's more, the rent is relatively related to the room area. We try to explain in the view of renting behavior. Since the housing price-to-income ratio is relatively high in Shenzhen so people do not have the ability to rent the large room and willing to pay a certain premium to smaller room. Thirdly, the joint rent is more expensive than whole rent. Relatively, Nanshan District has the highest rent and Pingshan District has the lowest rent. Lastly, the room face, room floor and whether the elevator is equipped are not significant for rent.

As for Guangzhou, the adjusted R-squared is only 0.736. We find three things different from Shenzhen. The first thing is that the room area now is positively related to the rent. We infer it is because there are more village in cities in Guangzhou and the whole is also lower than Shenzhen. So, people have the ability to rent a lager room. The demand is larger, and the rent is also higher. Secondly, the low floor rent is more expensive than high floor and the rent of house with elevator is significantly higher. We guess is because there are more elder people in Guangzhou and the lower floor or the house with elevator is more convenient for them. Lastly, we can find that the room face is all insignificant in Shenzhen while the room facing west is

much cheaper in Guangzhou. It may because that Guangzhou is too hot in summer and the living in the room facing west is intolerable.

As for Beijing, the adjusted R-squared is 0.601. The interpretability of our model is weak, we guess is because there are some other unknown factors in Beijing that lead to unreasonable house price, like speculation and the rent in Beijing is also influenced by some cultural history. Still, we also find one thing interesting. We can see that all those variable related to the distance are insignificant while the only significant factor is the distance to school. It is because the school district room is the key consideration for most tenants in Beijing, which has weakened the influence of other factors.

5.2 PCA, Ridge, LASSO

After normalization, we use PCA, Ridge and LASSO to do the regression and get the results as follow. We set the contribution of PCA reach 0.9999 and use five different λ: 0.001, 0.01, 0.1, 1, 10 to train the model and get the best λ for Ridge and LASSO.

R-Squared

	Shenzhen	Beijing	Guangzhou
Ridge	0.842	0.580	0.732
LASSO	0.857	0.605	0.742
PCA	0.844	0.477	0.702

Adjusted R-Squared

	Shenzhen	Beijing	Guangzhou
Ridge	0.840	0.575	0.724
LASSO	0.855	0.600	0.735
PCA	0.843	0.472	0.797

5.3 Population density and per capital GDP.

To see the relation between rent and population density and per capital GDP, we do another OLS regression in the file ***population&GDP.py***. As we expected, the higher the population density, the higher the per capita GDP and the higher the average rent. More explicitly, the relation between supply and demand and the economic level can determine the rent to a great extent.

	Population	Population density	GDP	GDP per capita
Coef	0.2771	0.1707	-0.1355	0.2715
P value	0.103	0.004	0.659	0.028

6. Conclusion

Based on our model, we can find the influence factors that explain the rent or house price and try to predict it in future data. For example, if there are some facilities in construction, our model can quickly react to it and find out undervalued rent. Besides, people in different cities have different preferences. Also, this can provide the basis for further investigation and research for specific cities.

7. References

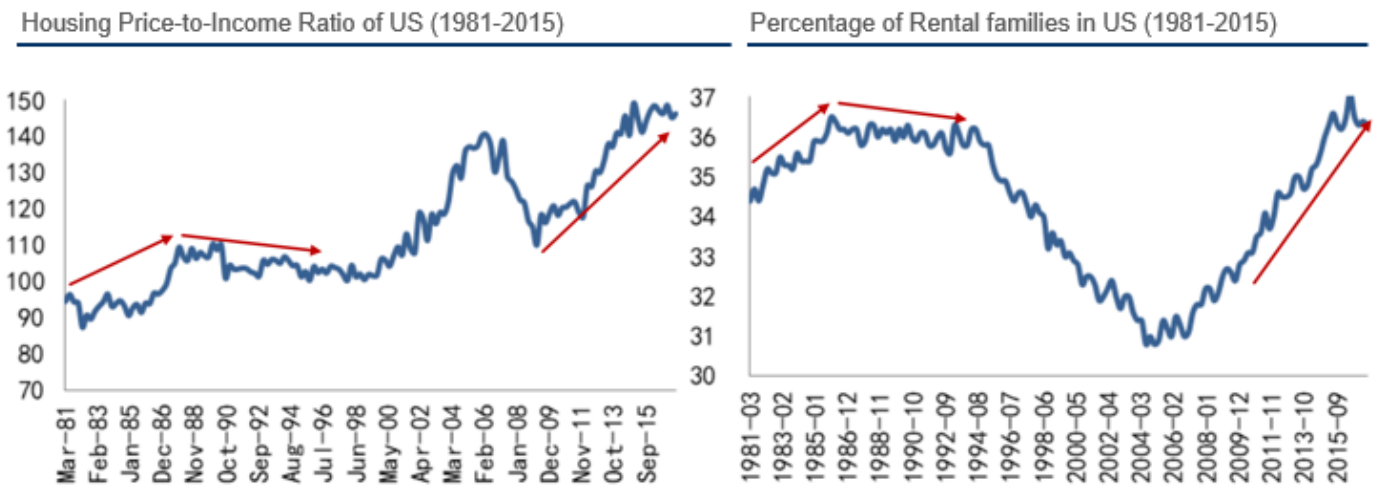
[1]兴业证券-房屋租赁系列报告一：美日历史，中国未来！-20170924

[2]银河证券-房地产行业：万亿级市场起航，租赁证券化将进一步夯实地产长效机制-180425

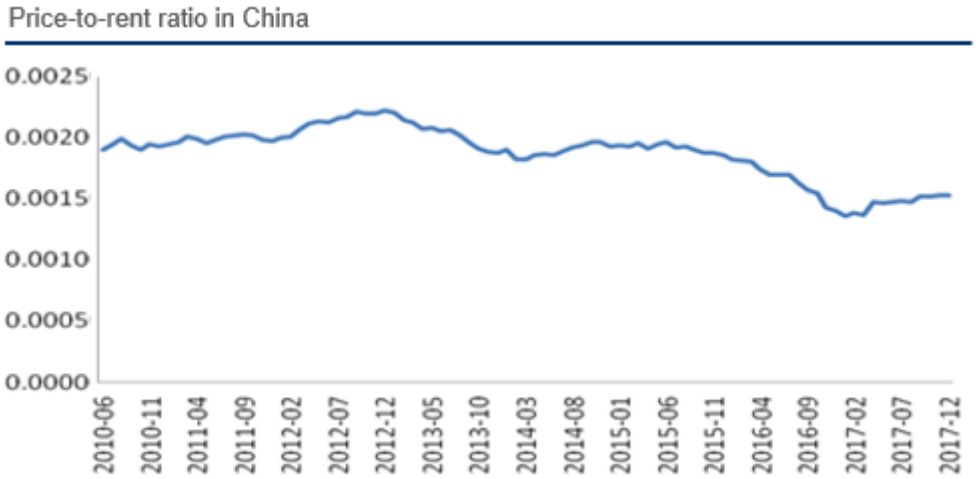
[3] MongoDB 官方网站：https://www.mongodb.com/

[4] Python 爬虫实现破解 58 同城加密内容；https://blog.csdn.net/MG_ApinG/article/details/88952932

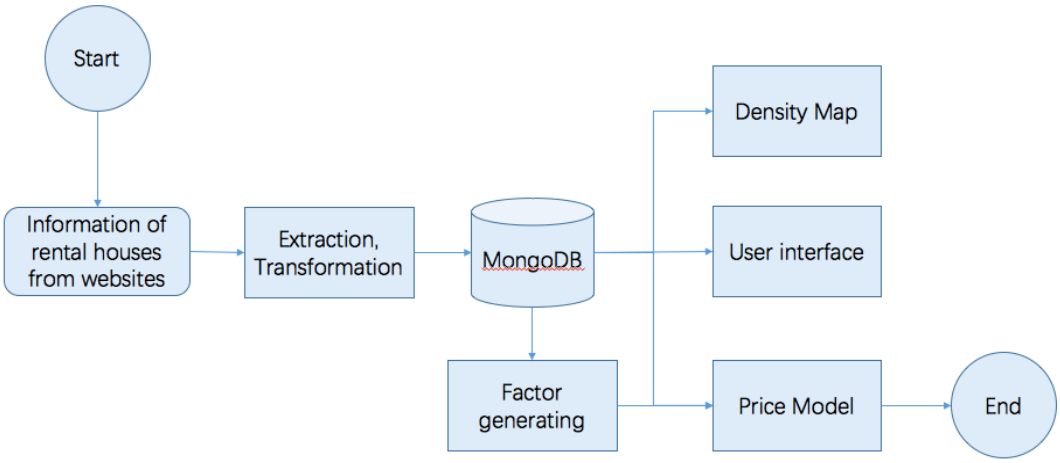
Appendix 1



Picture 1： Housing Price-to-Income Ratio and Percentage of Rental families in US (1981-2015)



Picture 2： Price-to-rent ratio in China

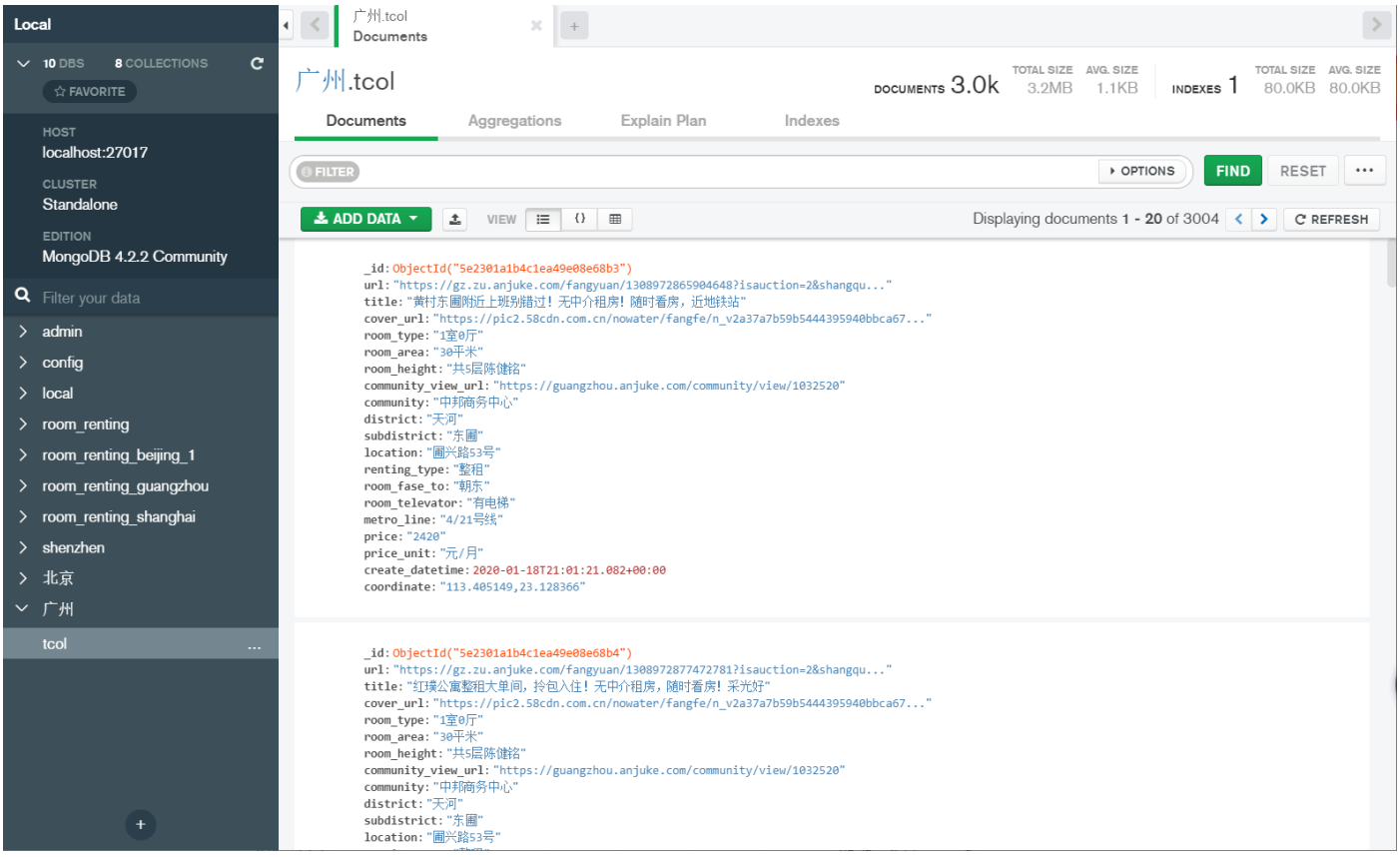


Picture 3 Workflow

Appendix 2

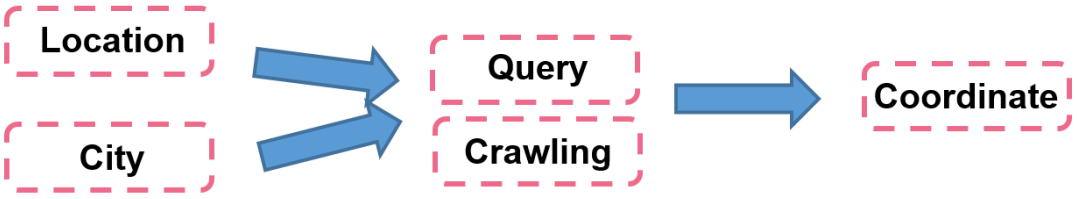
Category	Explain	Example
url	url of the house	https://sz.zu.anjuke.com/fangyua ···
title	title of the information	（首月免租） 年前不租房 年后没房租 数量有限 先到先得
cover_url	url of the cover picture	https://pic2.58cdn.com.cn... ..png
room_type	type of the house	3室内1厅
room_area	area of the house	41.4平方米
room_height	floor of the house	中层(共11层)
community_view_url	url of the information of community	https://shenzhen.anjuke.com/community...
community	community the house belonging to	春华四季园
district	district the house belonging to	龙华
subdistrict	subdistrict the house belonging to	民治
location	location of the house	龙平东路203号
renting_type	type of renting	整租
room_face_to	orientation of the window	朝东
room_elevator	whether has elevator	有电梯
metro_line	metro line nearby	4/5号线
price	price of the renting	1930
price_unit	price unit	元/月
create_datetime	create time of the information in database	2020-01-18T20:33:17.571Z""

Table 1 Crawling Data

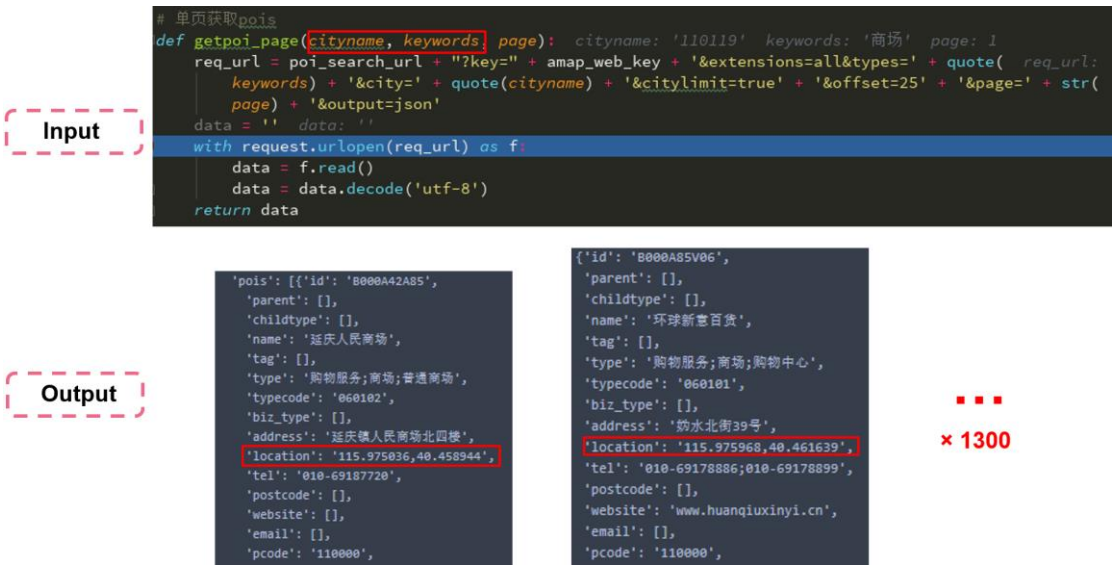


Picture 4: MongoDB

Appendix 3

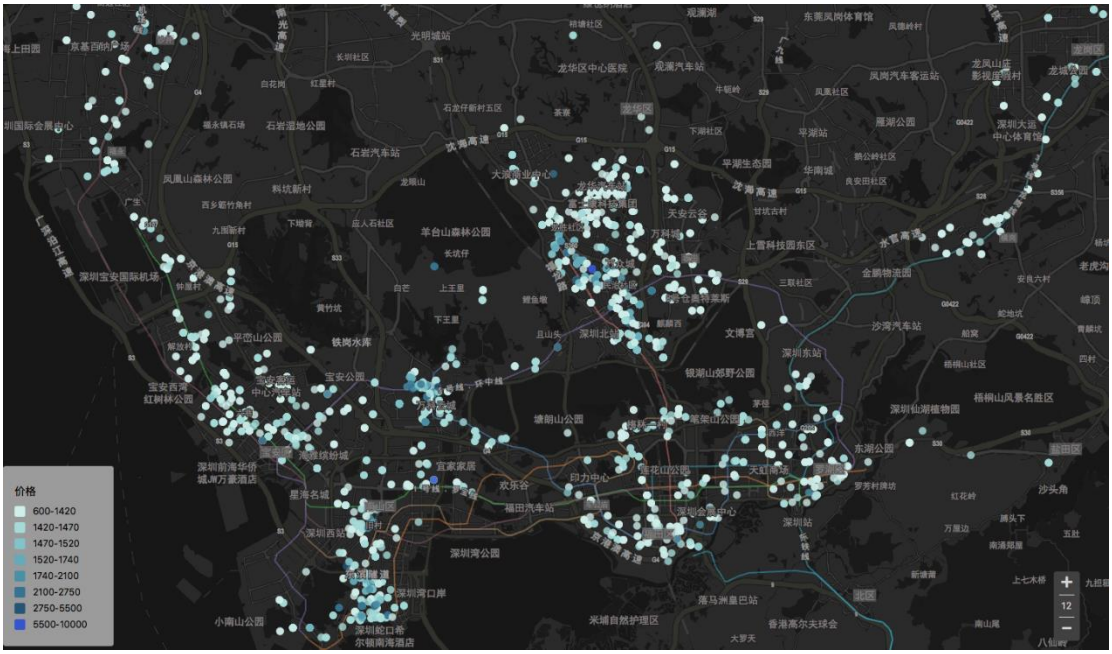


Picture 5 Pinpoint Searching Workflow

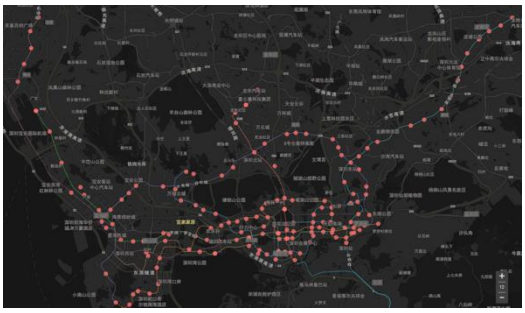


Picture 6 Key-words Searching Work Flow

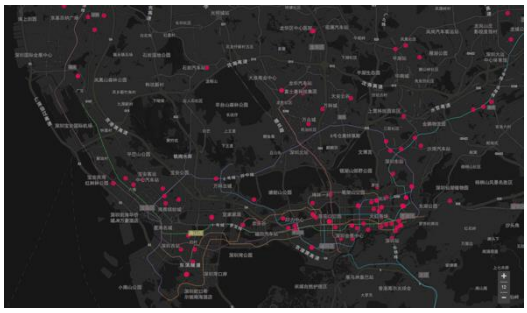
Appendix 4



Picture 7 Renting Houses Distribution

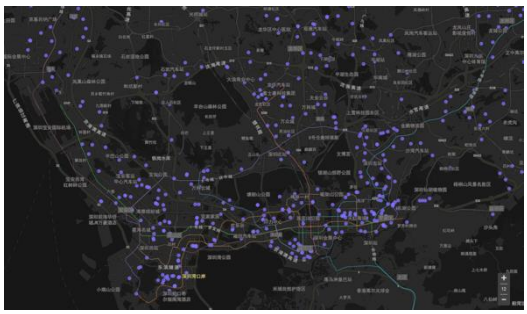


Metro stations



Hospitals

Shopping Malls



Schools

Picture 8 Service Facilities Distribution

Appendix 5

Room type	Room area	Room height	Building height	Renting type	Room face	Price	Longitude	latitude	Address
Three bedrooms and one living rooms	19 Square meters	Middle	7 floors	Whole	East	1900	114.0192	22.6325	Tong Bo Community, MingKang Road, Ming Zhi, Long Hua

Table 2 Renting Houses Information Example

Feature Name	Type	Feature Name	Type
num_bed_room	Float	hosp_distance	Float
num_living_room	Float	hosp_number	Float
room_area	Float	school_distance	Float
total_height	Float	school_number	Float
metro_distance	Float	height_type_low	Dummy
metro_number	Float	height_type_high	Dummy
CBD_distance	Float	room_face_type_east	Dummy
CBD_number	Float	…	
renting_type_whole	Dummy	district_NanShan	Dummy
		…	

Table 3 Total Features

Appendix 6

	Coefficients			P-value		
	Shenzhen	Beijing	Guangzhou	Shenzhen	Beijing	Guangzhou
Num_bed_room	9.4625	22.7128	3.6852	0	0	0
Num_living_room	-8.7710	-7.6554	-3.1832	0	0.383	0
Room_area	-0.5837	-0.4024	0.3969	0	0	0
Total_height	0.0870	0.0788	0.3556	0	0.577	0
Metro_distance	-1.6321	-5.0453	-0.2035	0.003	0.236	0.684
Metro_number	0.9204	0.3892	1.3385	0.018	0.547	0
CBD_distance	-0.5759	-3.7527	-3.8835	0.464	0.230	0
CBD_number	0.5378	0.0833	0.0908	0	0.022	0.636
Hosp_distance	-1.2262	-0.1143	-0.7888	0.004	0.855	0.026
Hosp_number	1.0045	0.5796	0.1649	0.001	0.243	0.202
School_distance	-2.8531	-0.4754	-1.1456	0.002	0.002	0.397
Sch_number	0.5950	1.4693	0.1066	0.001	0	0.282
Low floor	-1.9801	3.2891	0.7326	0.235	0.067	0.021
Underground	-	-	-1.212e-14	-	0.551	0.001
High floor	-0.1350	0.8318	-1.8547	0.856	0.001	0.007
Face southeast	-4.1617	-2.616e-14	3.2578	0.536	0	0.164
Face east west	-43.2242	-8.698e-14	-2.5718	0.125	0	0.467
Face south north	5.7630	138.6479	0.7863	0.404	0	0.742
Face east	-27.9446	25.3238	3.3689	0.316	0	0.184
Face north	-29.5440	25.8780	7.1163	0.201	0	0.005
Face south	-26.8677	22.2020	1.3561	0.060	0	0.550
Face west	-29.9623	20.0750	5.4853	0.735	0	0.077
Face northwest	3.5618		0.3593	0.724	0	0.906
Face southwest	1.7792	1.157e-13	5.4195	0.880	0	0.074
Whole rent	-21.3306	189.1190	-5.5135	0	0	0.083
Elevator	-1.5221	2.1322	5.0657	0.236	0.240	0

Table 4 Partial Coefficients & P-Value