Hong Kong University of Science and Technology

Bsc in Risk Management and Business Intelligence

RMBI4980 Report

Group 9

Project Title:

Construction of big data analysis platform for investors interested in housing in China

Supervisor: Mr. Wong Chi-Wing, Raymond

Student name: Chan Hin Yung Alice (20275833)

Yeung King Yiu (20276382)

Submission Date: 5th May, 2018

Content

Abstract

There are numerous real estate websites with different data formats in China. Some contain limited relevant housing market information. A comprehensive housing platform integrating property information from different sources and price influencers is lacking. Such information could be internal factors which are directly related to listed housings or external factors like nearby facilities. Despites property pricing models developed by researchers, these models are either in single dimension or too complicated for people to understand, which do not fully fit the needs of the platform users for direct application. In view of this, a prototype of housing search platform which includes data from 4 housing websites, supporting facilities and air quality index(AQI) was constructed. Beforehand, the selection of suitable visualization tool, choice of factors for performing visualization and data comparison analysis and the platform interface design were carried out to complete the platform. This paper would illustrate the procedural manual of using the platform.

Chapter 1: Introduction

1.1 Research Background

With the rise of e-commerce, online real-estate rental and trading platforms such as Fang.com(房天下), LEJU.com(樂居網) are gaining their momentum in China. According to iresearch.com (2017), Fang.com, the leading online real-estate information system in China, its number of month distinct visitors has already reached 1.2 billion. These online platforms enable users to filter out their ideal flats or apartments by criteria directly related to housing, for examples, district, monthly rent, apartment size. After applying the filter, details of the apartments such as floor plans, pictures of apartment, neighbouring facilities would be displayed. Users could have preliminary overview about the flat without doing research a lot by themselves.

Despite their convenience for users, online information system may start threatening the roles of real estate agents in recommending suitable flats and related information for investors. Daniel Lemire, a computer science professor at the University of Quebec, published an article: “Has the Internet killed real estate agents yet” on his blog (2017), questioning the necessity of real-estate agent in providing flat-related information to clients. From real-estate agents’ perspective, one way to counteract online information system is possibly more informative than existing online real-estate database. This prompts the values of a comprehensive, well designed real-estate information platform for the real-estate industries to visualize information and communicate with their customers.

From the perspective of property analysts, a research conducted by Catella Research, a leading financial advising and asset management firm pointed out the top 2 problems regarding transparency in the real estate sector. They are the lack of a central data resource, lack of standardized data. This also implies the demand from the real estate sector for a platform that could integrate multiple real-estate related data sources with consistent format.

The research from Catella Research also highlighted the potential of data-integration and data-visualization platforms for the real estate sector. The research revealed that almost a quarter of the interviewed firms in the real estate sector did not subscribed to any fee-charging real estate database. At the same time, about 16% the real-estate sector still did not hire any specialists for collecting and processing data. These figures imply a substantial market for companies which could offer suitable information platform for those unsubscribing firms to perform data collecting and processing work.

1.2 About research partner: Hangzhou Justar Technology Inc.

Hangzhou Justar Technology Inc. (Justar) is a Chinese company specializing in providing innovative system infrastructures and decision support softwares for financial institutions. Justar's team consists of professionals around the world. They mission in reversing the foreign-supplier dominating situation in Chinese FinTech sector. With their dedication, Justar has been honoured as Hi-Tech enterprise in Hangzhou Binjiang District. Since 2015, it has branched into Beijing, Shanghai, Hong Kong, offering customized and advanced risk management system and cloud financial analytic services.

Justar sees opportunities in developing a more comprehensive and user-friendly platform to visualize real estates' details for property analysts and agents. It, therefore, has decided to launch a project for a platform which dynamically integrates information from both listed flats and external sources.

1.3 Aim

Invited by Justar, students of Hong Kong University of Science and Technology majoring in Risk management and Business Intelligence are offered with an opportunity to cooperate with Justar. This sub-project thus was initiated, aiming at developing a platform prototype for Justar to visualize real estate information in Shanghai.

The ultimate goal of this platform prototype is to enable users to view information of a targeted real estate in Shanghai and evaluate its quality and features relative to other listed real estate effectively. Effectiveness here means reducing users effort for data collection, designing visualization method, comparative analysis of a flat's quality in different dimension. Therefore, the exact functions of the platform would include collecting data from different sources, displaying and comparing information of different real estates.

1.4 Objectives

To set up an effective real-estate information platform, indicators relevant for real estate evaluation and reliable data sources are selected, followed by the housing and indicators data visualization in a readable way.

Chapter 2: Literature review

2.1 Introduction

There have been many real-estate information exploration system websites in China. Many of them applied the dynamic query and natural language query interface. Dynamic query is an idea of databases visualization and search with the use of direct manipulation. It is applied in the form of sliders for users to tune their requirements and display the query result fulfilling their criteria. Natural language query allows people to post questions in English and then shows the requested information by transforming it into a logical database query (Shneiderman and Williamson, 1992). Though these concepts have been widely in use, many systems do not include indicators people also concern when they purchase properties. Thus, in this literature review, the indicators associating with the property price or investment value will be reviewed. The indicators can be classified into supporting facilities and air pollution factors in general. Besides, the concept of an effective information system is introduced as the reference for platform construction.

2.2 Supporting facilities factor

For supporting facilities factor, indicators related to the educational, transportation and sports facilities can be included in the platform.

2.3 Educational facilities indicator

Wen, Zhang and Zhang (2014) analyzed the relation between educational facilities and housing prices and drew a positive conclusion that primary and secondary school promotes a notable school district effect in China. They found that the property price in a school district rises by 2.02% or 5.443% respectively with the improvement of school banding by a level in primary and secondary school. The number of education institutions in a district also exerts effect on housing price such that the less remote an institution from the flat is, the higher the price. The property locating within 1 km to kindergarten, high school and college experiences boost in price by 0.3%, 2.737% and 5.443% respectively.
Nguyen-Hoang & Yinger(2011) found that rise in student test scores for a standard deviation can elevate the house values by almost 4%. This reflects the willingness of investors to pay for higher prices for education quality and accessibility and thus this indicator should be included in the platform.

2.4 Transportation facilities indicator

In terms of transportation facilities in China, they have mixed effect on property prices depending on their proximity to housing. When the distance between flats and high-speed rail stations is within the range of 0.891 km and 11.704 km, housing price increases owing to greater accessibility to other cities.

In contrast, when the distance ranges from 0.475 km and 0.891 km, housing price drops due to noise pollution (Geng, Bao and Liang, 2015). For effects of intra-city public transports, Fu et al. (2014) used the ClusRanking performances to differentiate the influences of geographic feature sets to property price. The study suggests that the effect of road network outruns bus stop, Point of Interest and subway, while the combination of all public transport has the greatest influencing power in both rising and falling markets. Therefore, the count and the distance to housing of these transportation means can be introduced to the platform.

2.5 Sports facilities indicator

Distance of sports facilities is found to have remarkable impact on prices of surrounding residential buildings (Feng and Humphreys, 2016). This research uses a spatial hedonic approach to figure out that for every 10% closer the distance from property to the facility, there is 1.75% escalation of housing price. It suggests that the distance of flat to the closest sports facilities can be one of the indicators.

2.6 Air pollution factor

Air pollution has been a severe problem in China that people particularly concern about the air quality of their living places (Chan and Yao, 2008). Chay and Greenstone (2005) investigated the capitalization of air quality into property price and found that people are willing to pay more for flats located in clean air zone, though this orientation is not very obvious. It is estimated that 0.2% to 0.4% growth in mean housing price is caused by 1 mg/m3 reduction in total suspended particulates (TSPs), having an elasticity of 0.2 to 0.35. Though the study found that the marginal gain from the drop in TSPs is lower in areas with serious pollution problem comparatively, there is an estimate of $45 billion total gain in property value in TSPs nonattainment counties in mid-1970s. Thus, putting air pollution index indicator in the platform is a possible option.

2.7 Information System Success Models

According to the Information System Success Models proposed (Delone & Mclean, 1992), 3 significant factors that determine an effective information system are: 1) information quality, 2) system quality and 3) service quality. Among the above 3 factors, system quality emphasize more about hardware support of the platform which is not the focus of this project. Information quality refers to **data relevance** and **data reliability**, System quality refers to the **data readability** of the platform. This provides a guideline for constructing an user-friendly housing platform.

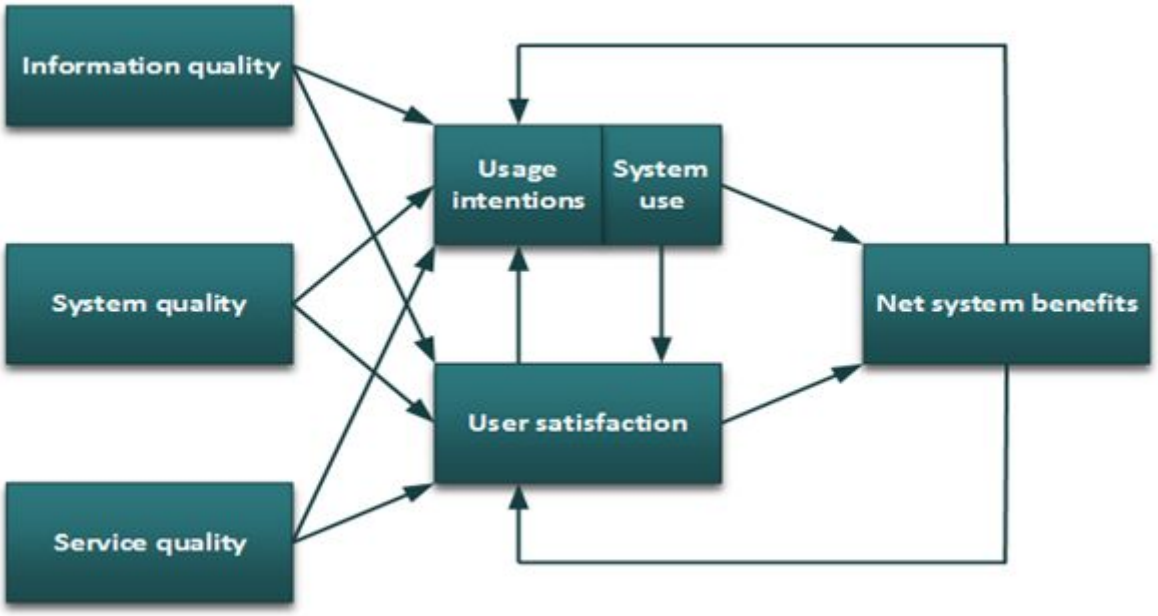


Figure 2.7: Information System Success Models

2.8 Conclusion

Various researchers have developed different real estate price estimation and investment valuation models. Wang et al. (2015) analyzed the impact factors by factor analysis method. The factors consist of economic factors such as per capita GRP and workers average wage, market factors such as area of housing in construction and sales area of commercial housing, and synthetic factor. Fu et al. (2014) focused on the impact of geographical features for properties appraisal. These valuation models all have high reference values, but there may be some factors omitted. Also, these models contain a basket of factors that may create confusion to the end users of the platform, so only the most determining indicators are selected for visualization and final score calculation in the platform to achieve the balance between the reference value of final score and the platform user experience.

Chapter 3: Methodology

3.1 Introduction

This section discussed how the final deliverable of this project: a prototype of a real-estate informative platform was constructed steps by steps. An overview of major procedures were explained first, followed by subjects and instrumentation. The implementation details of different procedures were then illustrated. The limitations and delimitation in this study were highlighted at the end of this section.

3.2 Procedures

Table 3.2: Procedures Overview

Stage	Task	Additional Description
1	Selection of indicators for data visualization	● External indicators ● Internal indicators
2	Data Collection	● Instrumentation ● Subject ● Data Crawling Process
3	Selection of data visualization platform	● Qliksense ● Elastic Search
4	Selection of data visualization formats	● Table ● Box-and-whisker diagram ● Treemap ● Heatmap ● Pie Chart ● Histogram ● Scatter-plot Diagram
5	Calculation of a property's scores in different dimensions	● Air quality ● Food ● Transportation ● Education ● Shopping

		• Healthcare
6	Gauge charts construction and final score calculation	
7	Ranking of property by final score	
8	Platform Construction	

3.2.1 Stage 1: Selection of indicators for data visualization

Information that a real-estate agent or analyst would be interested when evaluating a real estate is classified into 2 groups in this project. They are internal indicators and external indicators. Internal indicators refer to information that are directly related to a housing itself such as area and price. External indicators refer to environmental information such as air quality and supporting facilities near a housing. The indicators discussed below are transformed or analyzed and then included in the final visualization platform.

3.2.1.1 Internal indicators

3.2.1.1.1 Supporting facilities indicator

Supporting facilities within 1 km of a housing were points of interests to be included in the resulting data-visualization platform. They were identified with Google Places API. The reason of using Google Places API instead of Baidu API is that Google offer more free daily request quota. This is more suitable for trial-and-error stage of developing a information platform. “1 km” is chosen because this is about 13-15 minutes walking distance based on ordinary walking speed of an adult (Carey, 2015).

These supporting facilities are divided into 5 main groups are: 1) meal providers, 2) educational institutions, 3) stations of public transports, 4) health-care service providers 5) shops or shopping malls. Google has tagged locations on its map with a list of types already. The exact types of facilities corresponding to each group were deduced based on the specific list of types defined by Google Places API. The table below shows what types of facilities are on this project interests and which main groups they were classified into.

Table 3.2.1.1.1: Classification of Facilities

Grouping	**“Types” specified by Google Map**
Meal providers	• Meal_delivery • Meal_takeaway • Restaurant
Shops or shopping mall	• Bank

	• Shopping mall • Supermarket
Stations of public transports	• Train station • Subway station • Bus station
Educational facilities	• Schools
Health-care service providers	• Hospitals • Dentists • Doctors

3.2.1.1.3 Air quality indicator

The website http://aqicn.org/map/shanghai/hk/#@g/31.2345/121.4933/12z which is operated by the World Air Quality Index Project Team, is chosen to get the Shanghai air quality index (AQI) information from all Shanghai monitoring stations. The World Air Quality Index Project is a social enterprise project started in 2007, missioning in providing a unified transparent Air Quality information for the world wide. The data about Shanghai's air quality in this website sources from Shanghai Environment Monitoring Center. This website outperforms the official website by showing the AQIs of all monitoring station in Shanghai instead of a single general index for the city. By getting the AQIs of all AQI stations mapped to the closest real estate and AQI of Shanghai together, this platform enable user to compare the air quality of a real estate with that of the whole city. One key advantage is that users could figure out whether the air pollution surrounding a specific housing is a localized problem or city-wide problems. Real-estate agents and analysts could also tell how good the air quality of a housing is relative to other housings in the same cities.

3.2.1.2 Internal indicators

3.2.1.2.1 Fundamental Indicators

To decide which basic information about a real estate would be interested by users of real estate information platforms, this study has referenced to the interfaces of the top 4 rental or buy-and-sell real estate websites in China. They are Fang.com, Anjuke.com, Focus.com and Leju.com. The reasons for choosing these websites are because they share common positive characteristics that they all have large yet non-overlapping databases, being renowned for Chinese housing search engine and allowing the retrieval of data from them. Thus, a raft of data with possibly higher creditworthiness can be obtained.The data consists of: 1) District, 2) Estate, 3)Property Type, 4) Size, 5) Room Partition, 6) Floor, 7) Total number of floors in the building, 8) Property Age, 9) Orientation, 10) Selling Price and 11) Comments attributes.

3.2.2 Stage 2: Data Collection

3.2.2.1 Instrumentation

3.2.2.1.1 Website crawling tools

3.2.2.1.1.1 Python

Python is selected to perform website scraping. It is a powerful programming language which has great code readability, a simple syntax, and extensive free packages support from the standard library (Dalcín et al., 2008). This makes data crawling more convenient and cost-free.

3.2.2.1.1.2 BeautifulSoup4 (Python Libraries)

BeautifulSoup4 is a python library to extract the content from HTML and XML files to help the navigation, search and modification of the parse tree, which saves time and effort to crawl the useful information in websites. (Beautiful Soup Documentation, 2017)

3.2.2.1.1.3 Scrapy (Python Libraries)

Scrapy is also used to extract the content from HTML and XML files. It saves time in large scale web crawling and address web blocking problem.

3.2.2.1.1.4 Urllib.request (Python Libraries)

Urllib.request is a module in python standard library to aid the opening of URLs (20.6. urllib2 - extensible library for opening URLs, 2017).

3.2.2.1.1.5 Xlrd (Python Libraries)

The Xlrd module allows python to get data from excel files for further work.

3.2.2.1.1.6 Time (Python Libraries)

The time module was used particularly for regular pauses in the data crawling process.

3.2.2.1.1.7 Random (Python Libraries)

The random module was usually deployed with the time module to perform regular pauses within a random time selected in a preset time interval.

3.2.2.1.1.8 Threading (Python Libraries)

The threading module allows the creation of thread object which can execute multi-functions at the same time.

3.2.2.1.1.9 Googlemaps (Python Libraries)

Googlemaps is a python module for performing housing and facilities mapping. In the module, Geolocation function is deployed to find the coordinate of housings and air monitoring stations, and Places function is used to search for specific kinds of facilities within a specified radius of from a location.

3.2.2.1.2 Data storage tools

3.2.2.1.2.1 JSON

JSON (JavaScript Object Notation) is a data representation in Javascript. Using JSON data structures enables the transparent translation into the original data structures compatible to nearly all common programming languages. Its main advantages are that it has a higher degree of human readability and ease in parsing (Alexander, 2008).

3.2.2.1.2.2 CSV

CSV (Comma Separated Values) is a file format for data storage. It has an interface similar to excel. The advantages of using CSV are that they are easy to edit and can be processed by most softwares and programming tools.

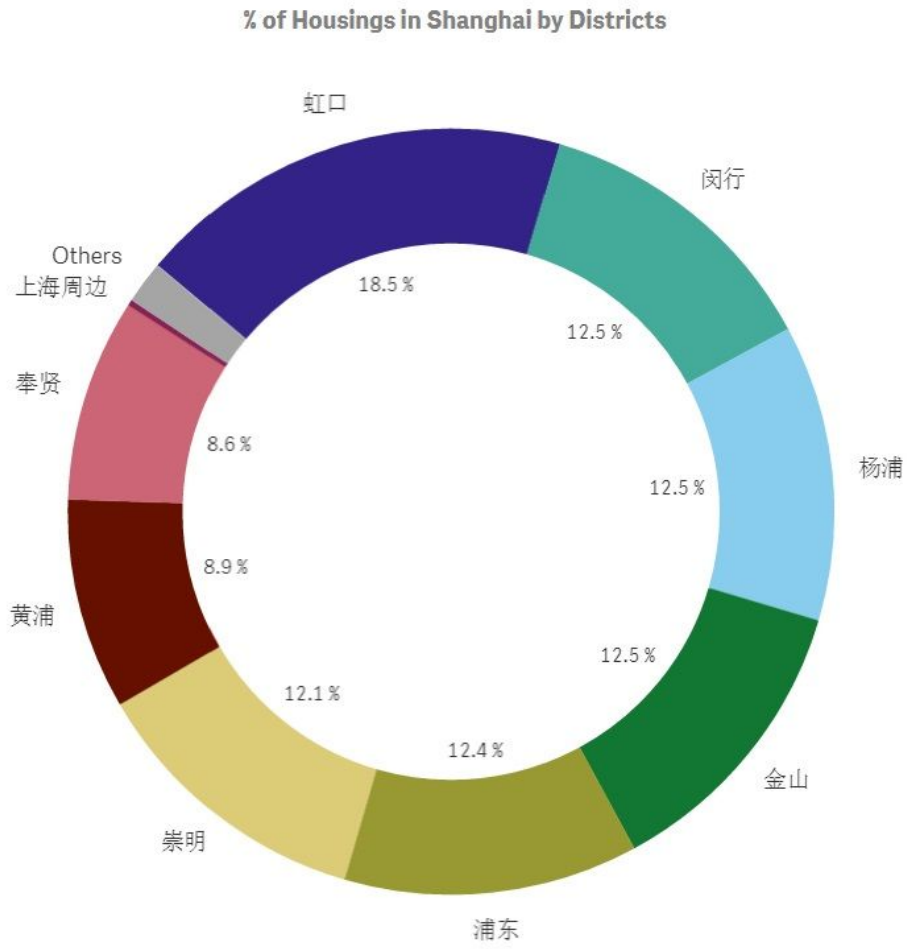
3.2.2.2 Subject

In April 2018, in total information of 23,477 selling 2nd-hand housings in Shanghai were collected from 4 online real-estate agencies through python program. About one-third of the data were from Anjuke.com and the another one-third were from Focus.com. Around 15% of the data source were from Fang.com and Leju.com. Details are shown in Table 3.2.2.2 below. They were 2nd-housings from 16 districts across Shanghai (one of them correspond to Shanghai's nearby region"'). They included different kinds of housings such as villa(别墅), duplex house (洋房). The proportion of different types of housing and proportion of housings in different districts analysed from raw data are shown in the Graph 3.2.2.2(a) and Graph 3.2.2.2(b).

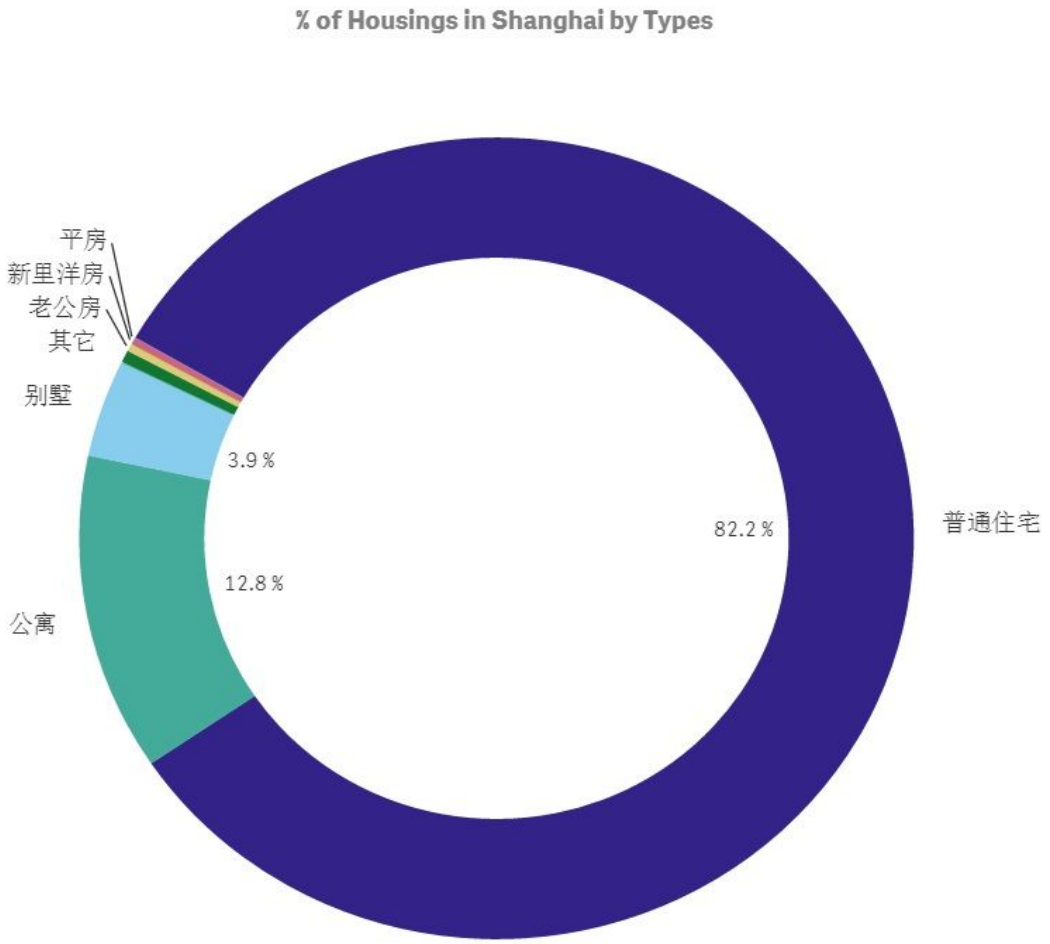
Table 3.2.2.2: Sources of Housing Data

Time	Website	No. of Housings	Number of Attributes
Early April	Anjuke.com	7,800	24
Early April	Fang.com	3986	26
Mid April	Focus.com	7786	24
Mid April	Leju.com	3905	23

Graph 3.2.2.2(a): Shares of Housings in Different Districts



Graph 3.2.2.2(b): Shares of different Types of housings



For Air Quality, Shanghai one week AQIs from 10 air quality monitoring stations were collected in late April from the http://aqicn.org/ website.

3.2.2.3 Data Crawling Process

3.2.2.3.1 Crawl Housing Data

Housing Data were collected from different only real-estate agencies by scrapy, a python libraries. In reality, these agencies created a specific catalog for all housings in a specific districts. They also created a catalog named as “all districts” which consists of proportion of housings from every district. With limited time and resource, housing data were only crawled from the catalogs named as “all districts” in the 4 mentioned real-estate agencies. It was assumed that proportion of real-estates from different districts in this catalog would be similar to that among all available housings in the websites. Graph 3.2.2.2(a) and Graph 3.2.2.2(b) above show that data collected still reflect a good mix of different types of real-estates from different districts.

3.2.2.3.2 Search for Supporting Facilities Near Every Housing

To search different kinds of facilities within 1km radius from a real-estate, locations of real-estates were required. Given that the estate where each listed housing is located is the most accurate locational

information available online, this project assumed the location of a listed housing's belonging estate is equal to the the listed housing's location. Distinct estate names then have been retrieved. Their geolocations could be obtained with the estate names through Google Map API's geocode function. Using these geolocations and others information as input arguments, Google Map API's places function has been executed to search for facilities nearby the inputted geolocations. The feedbacked facilities' information, for examples, names, geolocation was stored as a CSV.file.

3.2.2.3.3 Crawling and Mapping Shanghai Air Quality Index with Every Housing

With the geolocation obtained in the previous step, every housing would first be mapped to the closest monitoring station. The one week average AQI of the mapped monitoring station would be shown together with one week average AQI of the whole Shanghai City. Using one week average can eliminate short term fluctuations for the purpose of normalized air quality score calculation later on.

3.2.2.3.4 Data storage and Cleansing

After the website crawling procedures, relevant data was collected and stored. Then, entity resolution was done to combine housing data from different JSON and csv files into one table to facilitate the functioning of visualization tool afterwards.

3.2.3 Stage 3: Selection of data visualization platform

To screen out suitable visualization tools for building the platform, a brief review on future needs of BI (business intelligence) tools users was performed. BI Trend Monitor 2018, one of the larger BI market survey conducted by Business Application Research Centre's Research Study (2017), asked about 2800 users, consultants and vendors for their opinion about trends of BI market. Transforming Data With Intelligence™ (TDWI), an online BI educational and research institution, published an article discussing future trends in BI (Stodder, 2016). According to the market research and the article, 3 significant market trends are summarised.

The first trend is the emphasis on self-service BI. "Self-service" refers to the ability and ease for users to design visualization, transform and aggregate data with the BI tool. The second trend is strengthening data governance. "Data governance" refers to the ability and ease for users to control and ensure data consistency. The third trend is the incorporation of Cloud based analytics. This refers to the ability for users retrieve, process and visualize data from cloud server.

With reference to the above 3 key features, 2 suitable visualization tools have been discovered. They are Qliksense and Elasticsearch.

Concerning self service ability, both Qliksense and elasticsearch visualize data through a dashboard. Such setting enable users simultaneously displaying multiple data or charts in different formats on the dashboard. Users could design the interfaces of the dashboard by splitting the dashboard into different modules. One advantage of Qliksense over elasticsearch is that it is less code-based. Non-IT professionals could also design the layout, filter-and-aggregate dimension by simple drag and drop operation on the dashboard. Given that this 1-year project aims at building a preliminary visualization model, the more user-friendly and efficient design environment is more suitable for this project. Snapshots below are the dashboard samples of Elasticsearch and Qliksense respectively.

Figure 3.2.3(a): Elasticsearch's dashboard

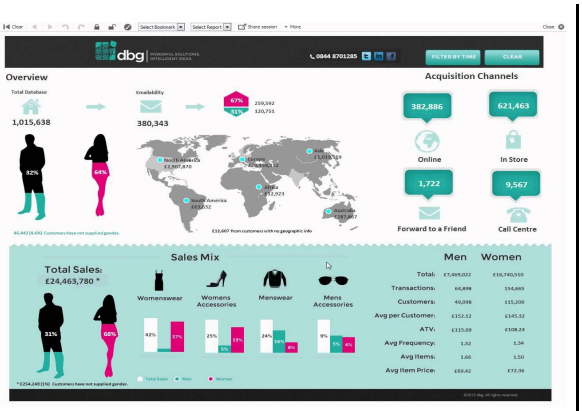


Figure 3.2.3(b): Elasticsearch's dashboard

In term of data governance, both elasticsearch and qlikview incorporate concepts of relational database like foreign key constraints, composite key constraints to help user ensure data integrity and consistency. They satisfy demand for data-governing feature on BI analytics tools.

In term Cloud-based functionality, both Elasticsearch and Qlikview enable connection to datamark in cloud servers. However, the extensibility of Qlikview seems better than Elasticsearch. The reason is that Qlikview also enables connection to 3rd-data analytics different engines such as SAS, R and python while elasticsearch only enables some of them. Since Justar, the industrial partner of this project, specializes in big data analysis, it is believed its professionals would like to apply more advanced data mining algorithm onto the platform. Thus, the higher extensibility of Qlikview would better match this project and Justar.

Based on the above comparison, this project decided to build a preliminary real-estate information platform with Qlikview.

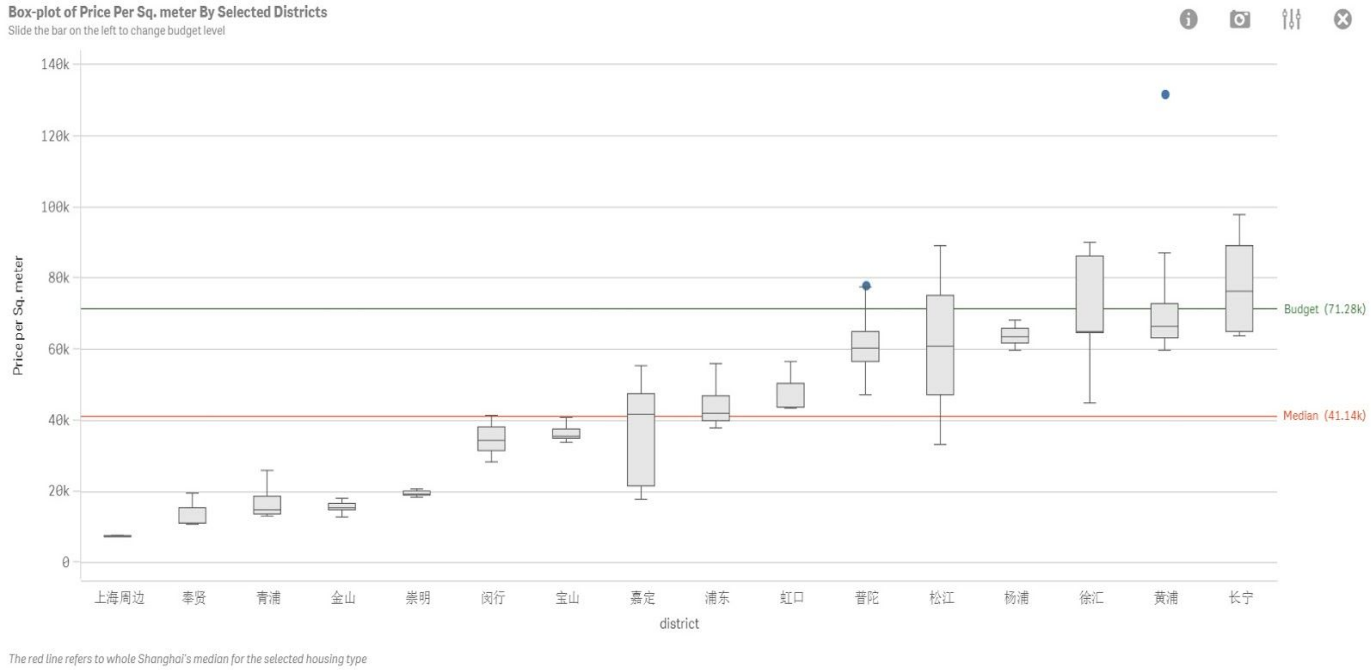
3.2.4 Stage 4: Selection of data visualization formats

3.2.4.1 Table

All housing attributes were displayed in table form in the platform.

3.2.4.2 Box-and-whisker diagram

Graph 3.2.4.2: An example of box-and-whisker diagram generated by Qliksense

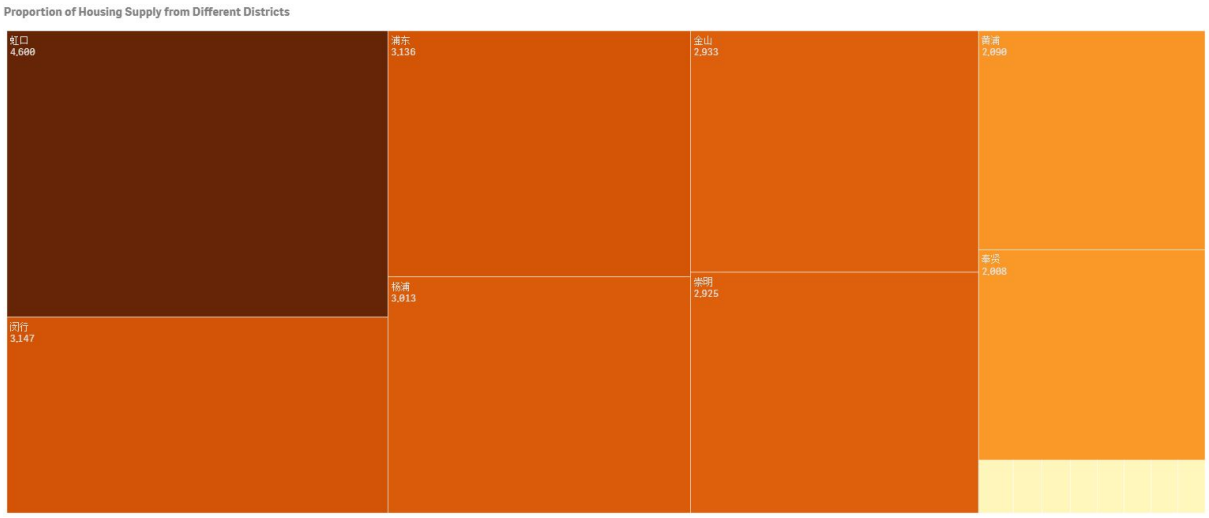


Box-and-whisker diagram was used in visualization of the distribution of price, price per square meter and property age across districts. Users could easily identify the district(s) satisfying their criteria on prices and property age on the box-and-whisker diagram by looking at the threshold (budget line) .

3.2.4.3 Treemap

The platform would further dissect the supply quantities by districts. Supply quantities by districts would be shown by a treemap as the following:

Graph 3.2.4.3: An example of Treemap generated by Qliksense



In the treemap, each rectangular block represents a single district. On each block, the exact supply quantity for the corresponding district would be stated as well. The purpose of using this treemap is to help real-estate analysts or agents quickly understand which districts are the main housing supply centres in Shanghai. Hence, they could have a brief idea on which districts provide more housing options for potential investors. The color of each block depends on its supply quantity. The greater the supply quantity of the corresponding district, the more intense the color of that district is.

3.2.4.4 Heatmap

The following heatmap would also be used to visualize different districts' supply quantities.

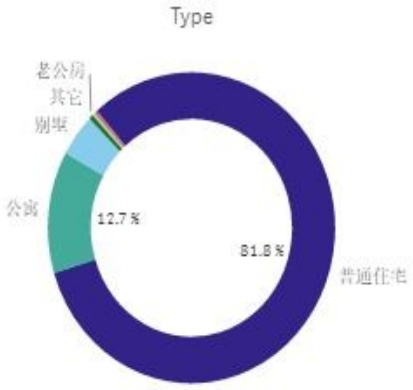
Graph 3.2.4.4: An example of Heatmap generated by Qliksense

Location of every district would be mapped and represented by a dot. Similar to the treemap, the greater the supply quantity in a district, the more intense the color of its corresponding dot is. The size of dots would grow with supply quantities as well. The advantage of the heat map is to show the supply

quantities of different districts together with geographical information. This could facilitates users' understanding about the distribution of housing supply in Shanghai geographically, especially for those unfamiliar with Shanghai.

3.2.4.5 Pie Chart

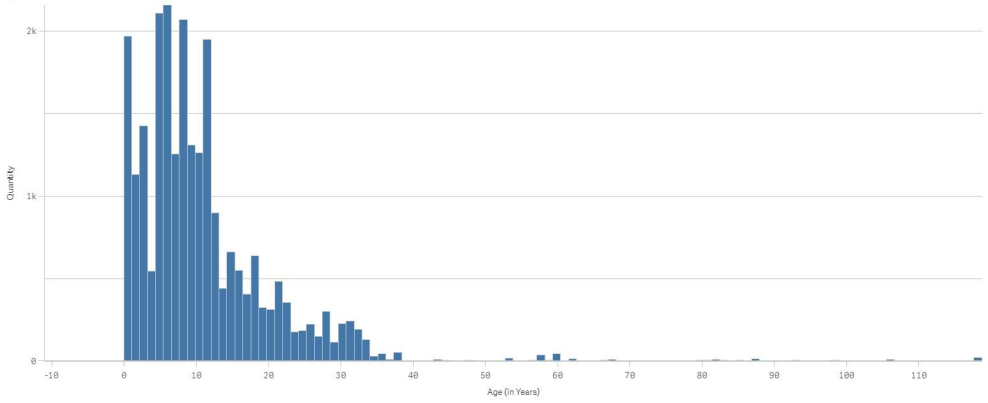
Graph 3.2.4.5: An example of Pie Chart generated by Qliksense



Housing attributes of Housing Types and Area were displayed in pie charts to view their proportions clearly.

3.2.4.6 Histogram

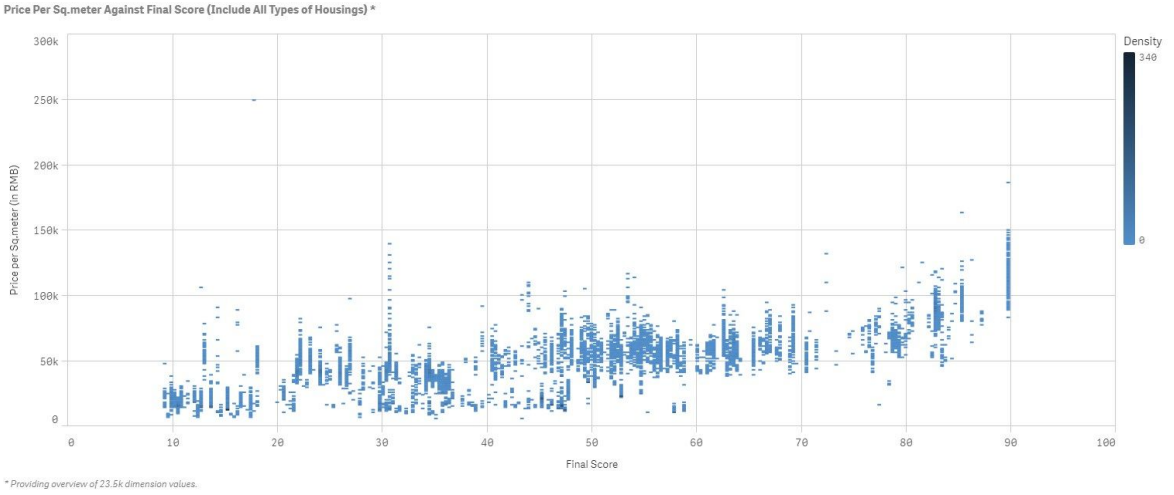
Graph 3.2.4.6: An example of Histogram generated by Qliksense



Housing attributes of Age, Selling Price, Selling Price per square meter were displayed in histograms to view its distribution clearly.

3.2.4.7 Scatter-plot Diagram

Graph 3.2.4.7: An example of Scatter-plot Diagram generated by Qliksense



The Price per square meter attribute was plotted against the Final score attribute for clear visualization of its trend.

3.2.5 Stage 5: Calculation of a property's scores in different dimensions

To enable easy comparison on the quality of different housings in different dimensions, the resulting platform would score a real estate in 6 aspects. These 6 aspects are: air quality, food, transportation, education, shopping and healthcare facilities. One key assumption is that property prices reflect quality of a real estate. Thus, scores of a property in different dimensions could be deduced by quantifying the relationship between indicators and property prices.These scores would be calculated based on a housing's attributes which are related to a specific dimension. Different mechanisms would be used for calculating the score in different dimensions.

3.2.4.1 Air quality

The normalized air quality score was calculated from the one week average AQIs from all air monitoring stations. Each air station had its own one week average AQI to compute its normalized air quality score. The official Shanghai air quality index (AQI) divided the air quality into six conditions which are 'good', 'moderate', 'unhealthy for sensitive groups', 'unhealthy', 'very unhealthy' and 'hazardous', with AQI of '0-50', '51-100', '101-150', '151-200', '201-250', '251-300', '>300' respectively. With reference to the scale, the average individual AQIs were normalized in the range of 0 to 350. The normalized air quality score was calculated as follows:

$$(1 - \frac{average\ AQI}{350}) \times 100$$

Such that the higher the normalized air quality score, the better is the air quality of a region.

3.2.4.2 Food

The Food dimension contained restaurants, food delivery kiosks and food takeaway kiosks within 1km of the property. The normalized food facilities score was calculated as follows:

$$\frac{(number\ of\ food\ facilities - minimum\ number\ of\ food\ facilities)}{(maximum\ number\ of\ food\ facilities - minimum\ number\ of\ food\ facilities)} \times 100$$

The maximum and minimum number of food facilities were found by comparing all datasets crawled from Google API. The formula gave a higher normalized score if there is higher number of food facilities, vice versa.

3.2.4.3 Transportation

The Transportation dimension contained bus stations, subway stations and train stations within 1km of the property. Different weightings were assigned in the transportation normalized score calculation in accordance to their impact on property value within 1 km radius. The values were specified with reference to a research on geographic dependencies on real estate value(Fu et al., 2014). The computation of the number of transportation facilities was as follows:

$$Number\ of\ transportation\ facilities = number\ of\ bus\ station \times 1.7 + (number\ of\ train\ station + number\ of\ subway\ station) \times 18.7$$

The normalized score of transportation facilities was calculated as follows:

$$\frac{(number\ of\ transportation\ facilities - minimum\ number\ of\ transportation\ \ facilities)}{(maximum\ number\ of\ transportation\ facilities - minimum\ number\ of\ transportation\ \ facilities)} \times 100$$

The maximum and minimum number of transportation facilities were found by comparing all datasets crawled from Google API. The formula gave a higher normalized score if there is higher number of transportation facilities, vice versa.

3.2.4.4 Education

The Education dimension contained all forms of education institutions within 1km of the property. The normalized education facilities score was calculated as follows:

$$\frac{(number\ of\ education\ facilities - minimum\ number\ of\ education\ facilities)}{(maximum\ number\ of\ education\ facilities - minimum\ number\ of\ education\ facilities)} \times 100$$

The maximum and minimum number of education facilities were found by comparing all datasets crawled from Google API. The formula gave a higher normalized score if there is higher number of education facilities, vice versa.

3.2.4.5 Shopping

The Shopping dimension contained banks, supermarkets and shopping malls within 1km of the property. The normalized shopping facilities score was calculated as follows:

$$\frac{(number\ of\ shopping\ facilities - minimum\ number\ of\ shopping\ facilities)}{(maximum\ number\ of\ shopping\ facilities - minimum\ number\ of\ shopping\ facilities)} \times 100$$

The maximum and minimum number of shopping facilities were found by comparing all datasets crawled from Google API. The formula gave a higher normalized score if there is higher number of shopping facilities, vice versa.

3.2.4.6 Healthcare

The Healthcare dimension contained hospitals, ordinary clinics and dental clinics within 1km of the property. The normalized healthcare facilities score was calculated as follows:

$$\frac{(number\ of\ healthcare\ facilities - minimum\ number\ of\ healthcare\ facilities)}{(maximum\ number\ of\ healthcare\ facilities - minimum\ number\ of\ healthcare\ facilities)} \times 100$$

The maximum and minimum number of healthcare facilities were found by comparing all datasets crawled from Google API. The formula gave a higher normalized score if there is higher number of healthcare facilities, vice versa.

3.2.6 Stage 6: Gauge charts construction and final score calculation

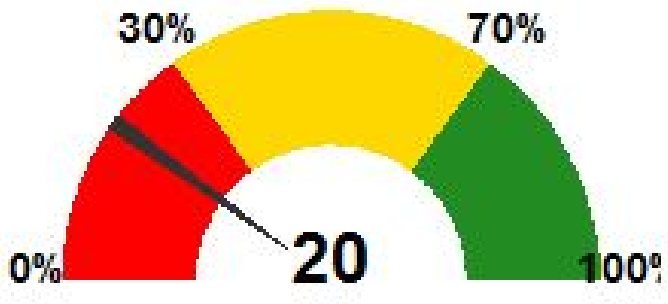


Figure 3.2.6: An example of Gauge Chart

The normalized scores in each dimension would be visualized in the form of a gauge chart which clearly graphed the relative strengths and weakness of each property. This helped speed up people's judgement on different flats as they could make a quick evaluation on the features of flats.
The final score of a property was calculated from the following formula:

Final Score = w0 × air quality normalized score + w1 × food facilities normalized score + w2 × transportation facilities normalized score + w3 education facilities normalized score + w4 shopping facilities normalized score + w5 healthcare facilities normalized score

The multiplier w0, w1, w2, w3, w4 and w5 were the weights of the respective normalized score. They were preset to be equal, i.e. ⅙ in the platform. The platform users can change these weights according to their preference.

3.2.7 Stage 7: Ranking of property by final score

When people search for an estate in a district, the search result will display the related properties from the highest to lowest final score. Other than the crawled information, the gauge chart, final score, ranking of the flat relative to other flats in the same district and in Shanghai will also be shown. This can help buyers to make a more informed decision.

3.2.8 Stage 8: Platform Construction

The following figures shows the interface of the platform which incorporates the elements aforementioned. In ‘Chapter 4: Result’ section the usage of the platform will be explained in detail.

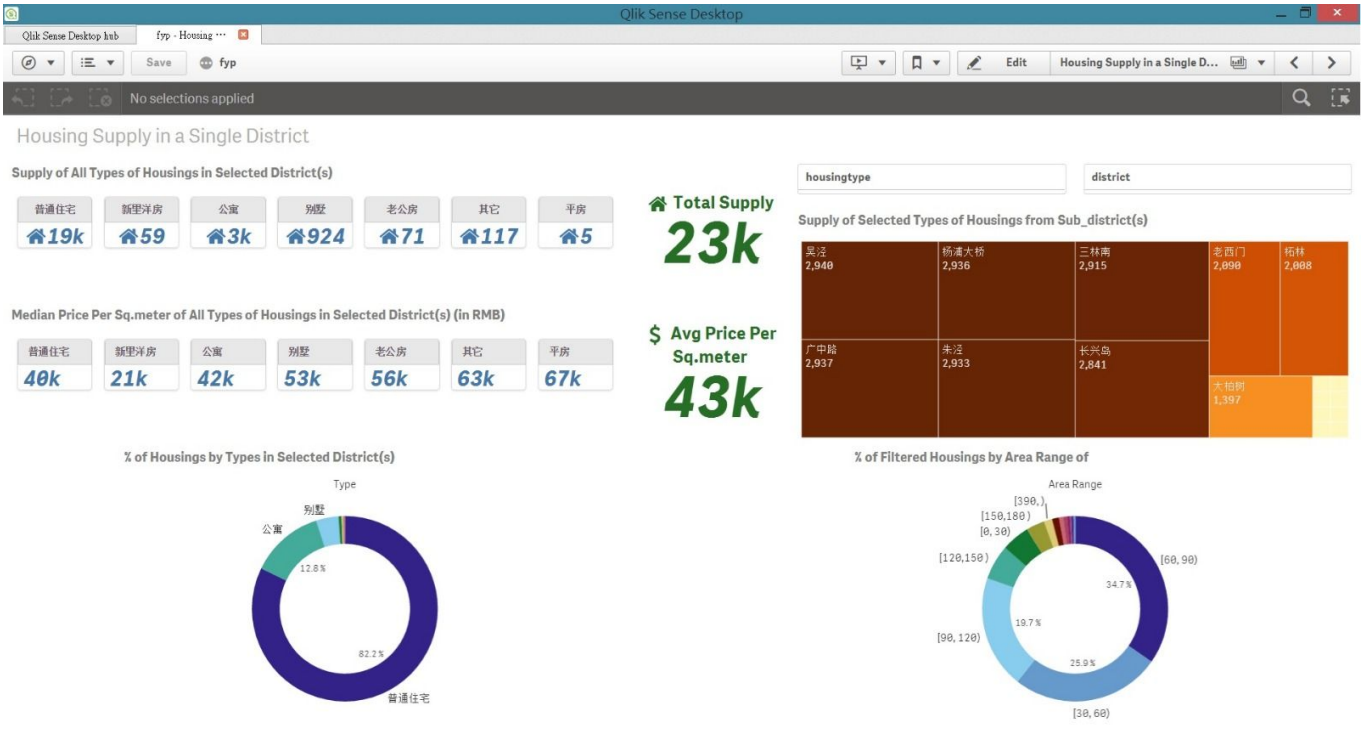


Figure 3.2.8: An example of a page in platform

Chapter 4: Results

This section explains the interface and functions of this project's resulting platform. This platform consists of multiple dashboards which are filled with varying data visualization. All the visualization are based on the data described in the “”Data Collection” section in “Methodologies”. Dashboards are divided into 4 batches serving different purposes. The 4 batches are “Overview of Housing Supply in Shanghai”, “Overview of Housing Supply in a Single District”, “Filter for Screening out Suitable Housings” and “Evaluation of a Single Housing”.

4.1 Overview of Housing Supply in Shanghai

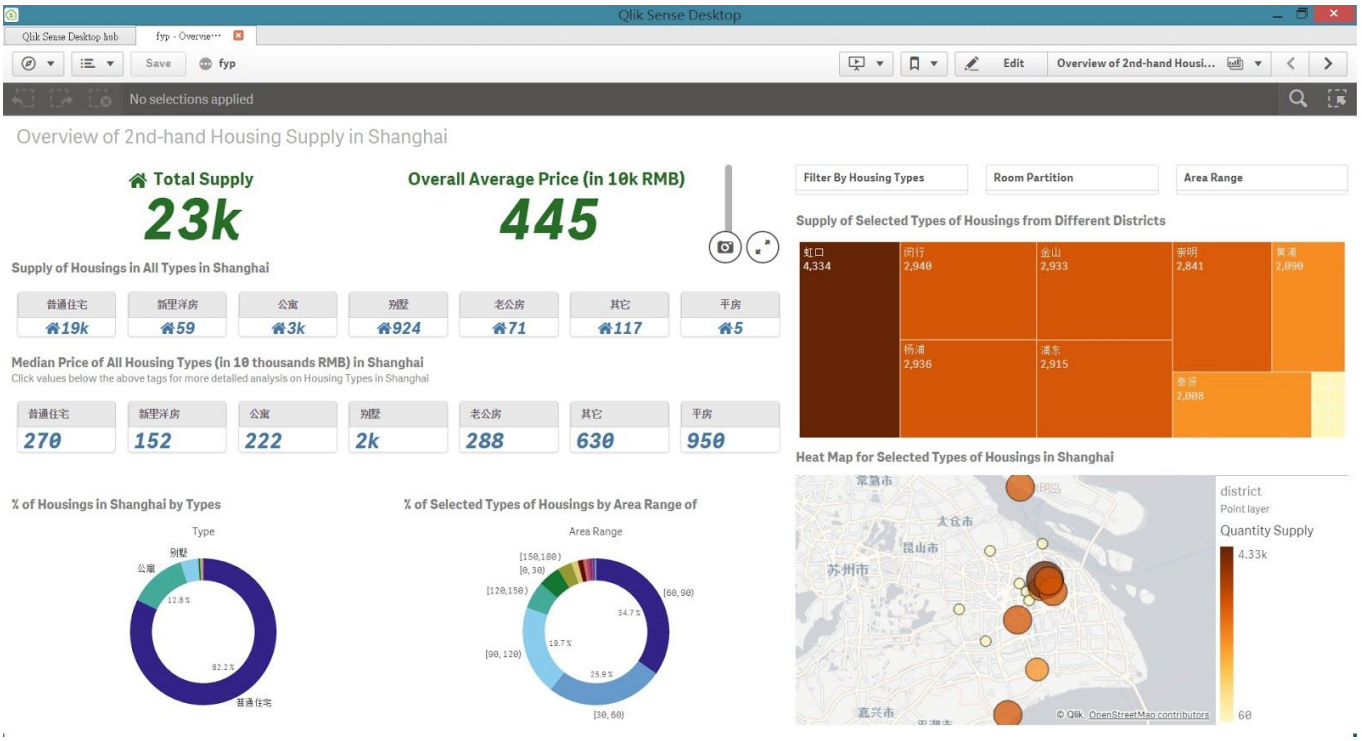


Figure 4.1(a): Homepage of the Visualization Platform

Figure 4.1(a) Shows the homepage of the resulting visualization platform. The goal of this page is to provide user a general picture about the supply of 2nd-hand housings in Shanghai.

There are two green key performance indicators (KPI) on the top-left corner of the. One of them highlights the total quantity supplied in Shanghai regardless to the types or located districts at a moment. Another one highlights the average price of all 2nd housings in Shanghai. If online housing data are collected regularly, the timely (e.g. monthly, weekly) percentage change of these 2 KPI could be displayed near them as well.

Under the 2 green KPI, there are 2 tables. The one above show quantity supplied of different types of housings. The one below show the median price of different types of housings. Similarly, timely percentage change could be displayed near the shown values if data are collected regularly.By clicking median price shown in the table below, users could be directed to another page for more detail analysis about the price of different types of housings in Shanghai. Its layout is shown with the next photo in this section.

Two pie charts are located at the bottom-left corner. The one on left describe the percentages of different types of housings in Shanghai. The other one on the right describe the percentages of housings with different sizes in Shanghai.

On the right, there is a treemap and a heat map. The treemap above illustrates the how housing supply is distributed across different districts in Shanghai. The heat map below provides similar insights but together with geographical location of different districts. There is a filter panel above these two charts. It provides 3 filter options: 1) filter by housing types, 2) filter by room partition, 3) filter by area range. When a user apply some filter options, the visualization of the treemap and the heat map changes accordingly.

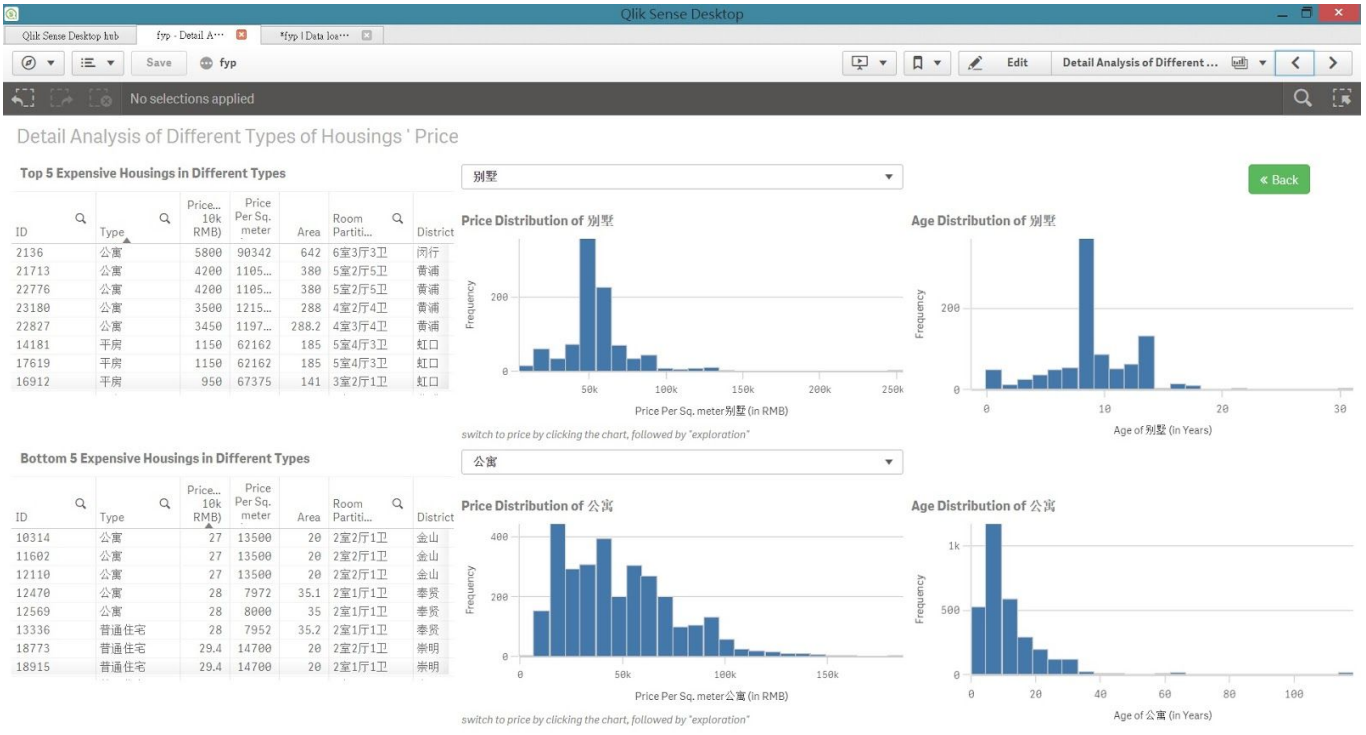


Figure 4.1(b): Detailed Analysis of Price of Differents Types of Housings

It has been mentioned that users would be directed to a new page if they click the median prices shown in a table at the Homepage.Figure 4.1(b) is that page to which users would be directed

There 2 tables on the left. The upper one lists information about currently top 5 most expensive housings of every housing type. The lower table lists similar information, but the information is about the bottom 5 expensive housing of every housing type.

On the right, there are 4 histograms, The upper two belong to 1 set of histograms and the lower two belong to another set. Above each set of histogram, there is a drop-down menu. The drop-down menu lists all possible housing types. Users could select a specific type in the menu to control what type of housings the set of histograms below are referring to. For each set of histograms, the histogram on the left reveals the price or price per square meter distribution of the selected housing type.The histogram on the left reveal the age distribution of the selected housing type.

Users could click the green “back” button at the top-right corner to go back to the homepage of the platform, continuing with the 2nd part of “Overview of Housing Supply in Shanghai”

.

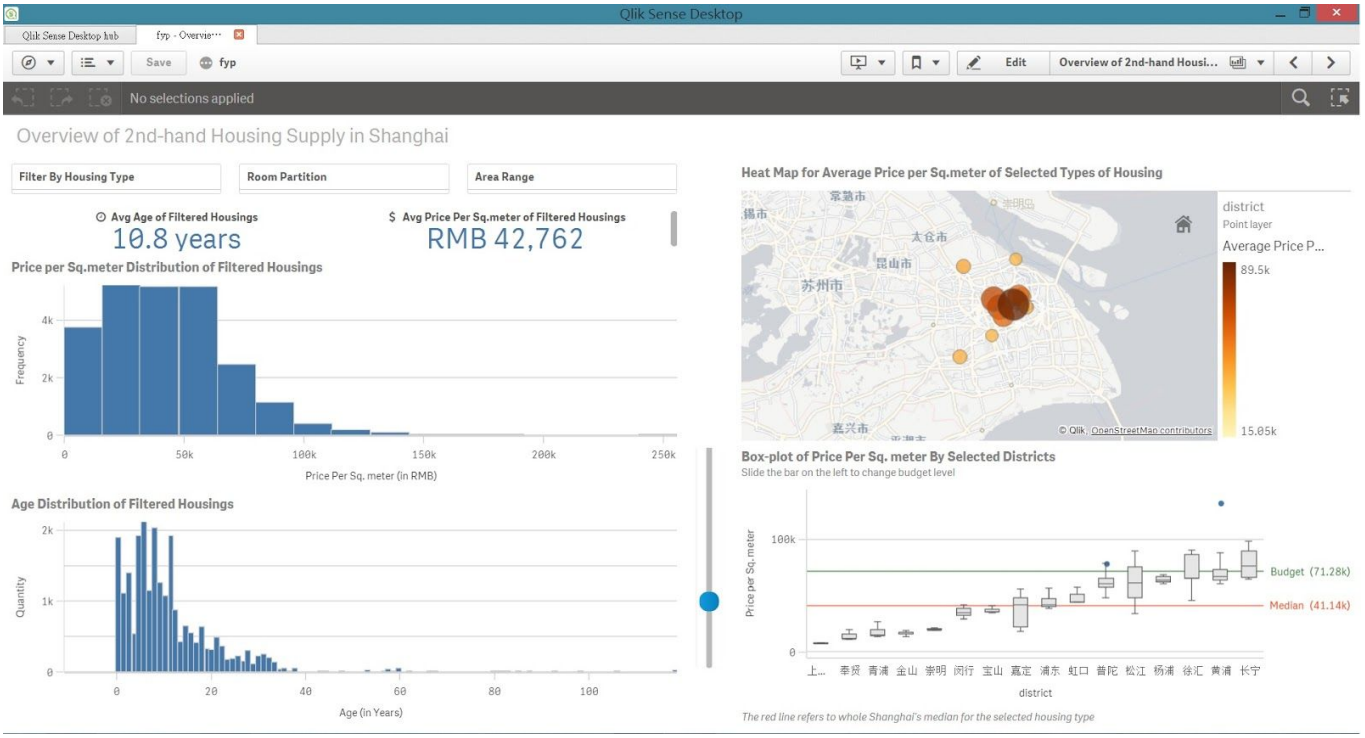


Figure 4.1(c): 2nd part of “Overview of Housing Supply in Shanghai”

Figure 4.1(c) illustrates 2ne part of “Overview of Housing Supply in Shanghai”. There is filter panel at the top-left corner of this page. Similar to the homepage, it allows users to filter by: 1) housing type, 2) room partition and 3) area range. All visualization on this page changes according to the applied filter options.

At the top-left corner of this page, average age average price per sq meter of the filtered housings would be highlighted. Under them, two histograms are shown. The upper one describes the distribution of price per square among filtered housings. The one below describes age distribution of these housings.

The heat map placed at the top right corner describe price level of housings across districts in Shanghai. Average price per square meter of housings in a district is used as measurement.

The box-plot at the bottom right corner roughly outline the distribution of price per square meter across different districts. Median price of the filtered housings is drawn as the red horizontal line for reference. The slide bar on the left of the box-plot control the level of the green reference line, which stands for the budget of the user.

4.2 Overview of Housing Supply in a Single District

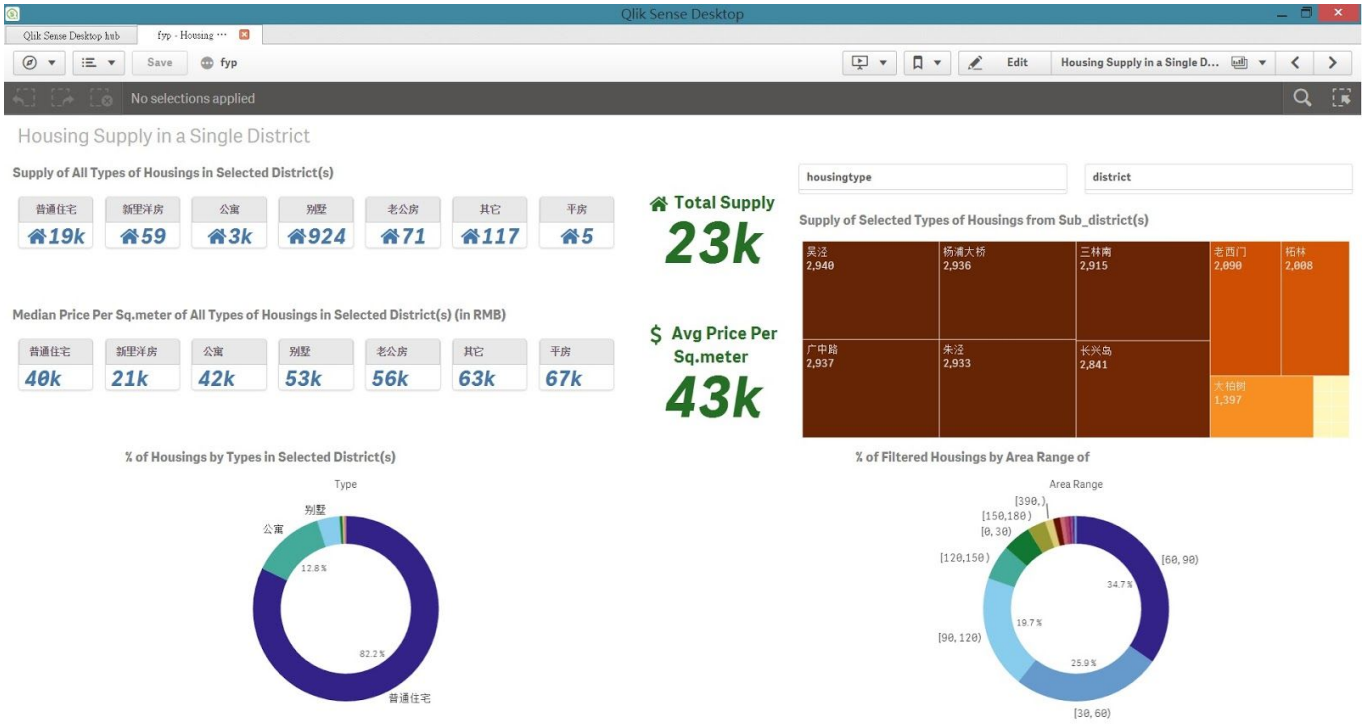


Figure 4.2: Overview of Housing Supply in a Single District

The dashboard shown in Figure 4.2 aims at providing a quick overview of housing supply in a single district in Shanghai. The layout is similar to the homepage. All visualization in this page changes according to the filter. That means visualization in this page is based on housing in the selected district.

Another the difference is the treemap. Unlike the treemap at homepage, this one describes how housing supply is distributed across different sub-districts in the selected district.

4.3 Filter for Screening out Suitable Housings

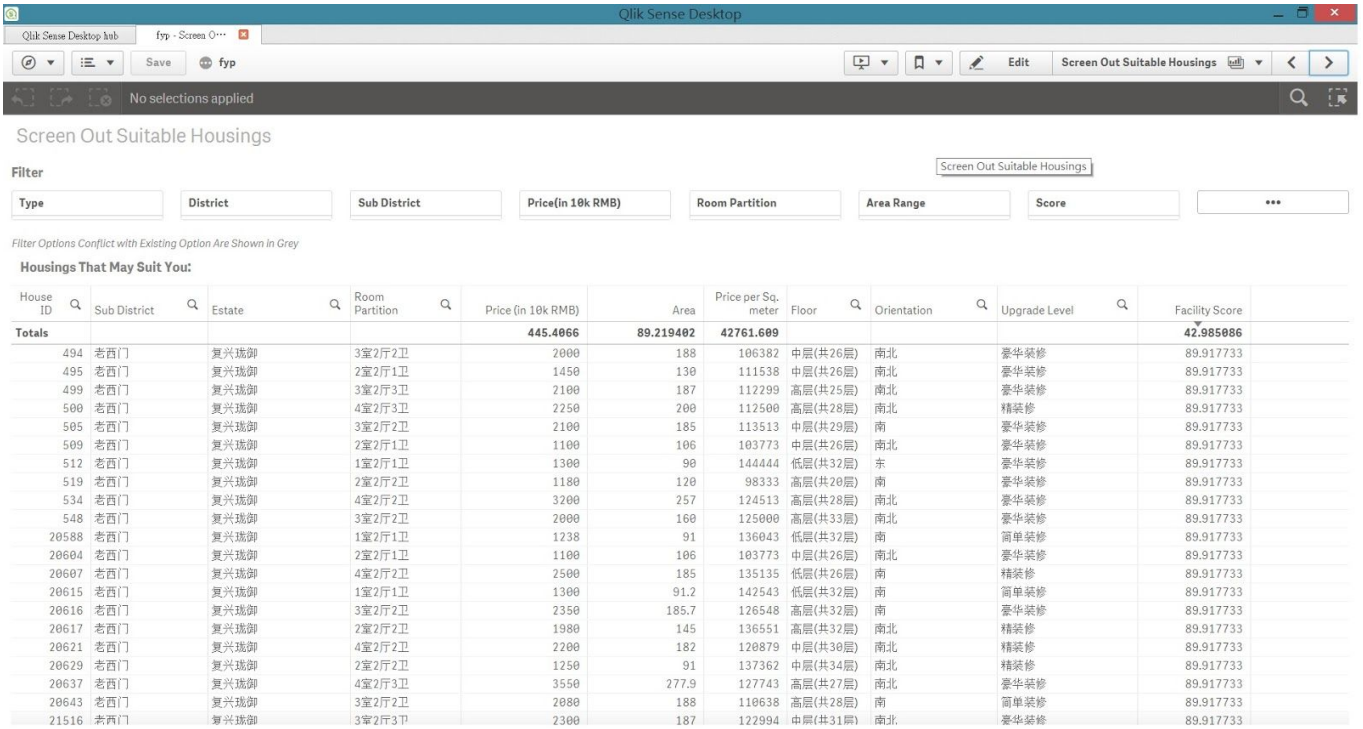


Figure 4.3: Filter out Suitable Housings through the Platform

Figure 4.3 shows a dashboard which is designed to help users filter out suitable housings for further detailed evaluation. The filter panel at the top of the page provide much more screening criteria. They include: 1) housing type, 2) district, 3)sub-district, 4) price, 5) room partition, 6) area range, 7)scores (rated by this platform), 8) price per square meter. The table below lists housings matching the screening criteria stated in the filter panel.

4.4 Evaluation of a Single Housing

Figure 4.4(a): 1st Page for Evaluation of a Single Housing

This page aims at providing more specific details of a single housing for a user's evaluation. In the previous page, users could obtain the house id of filtered-out suitable houses. At this page, user could input a house id into the input box at the top-left corner. Information of the corresponding house is shown in this page and subsequent pages.

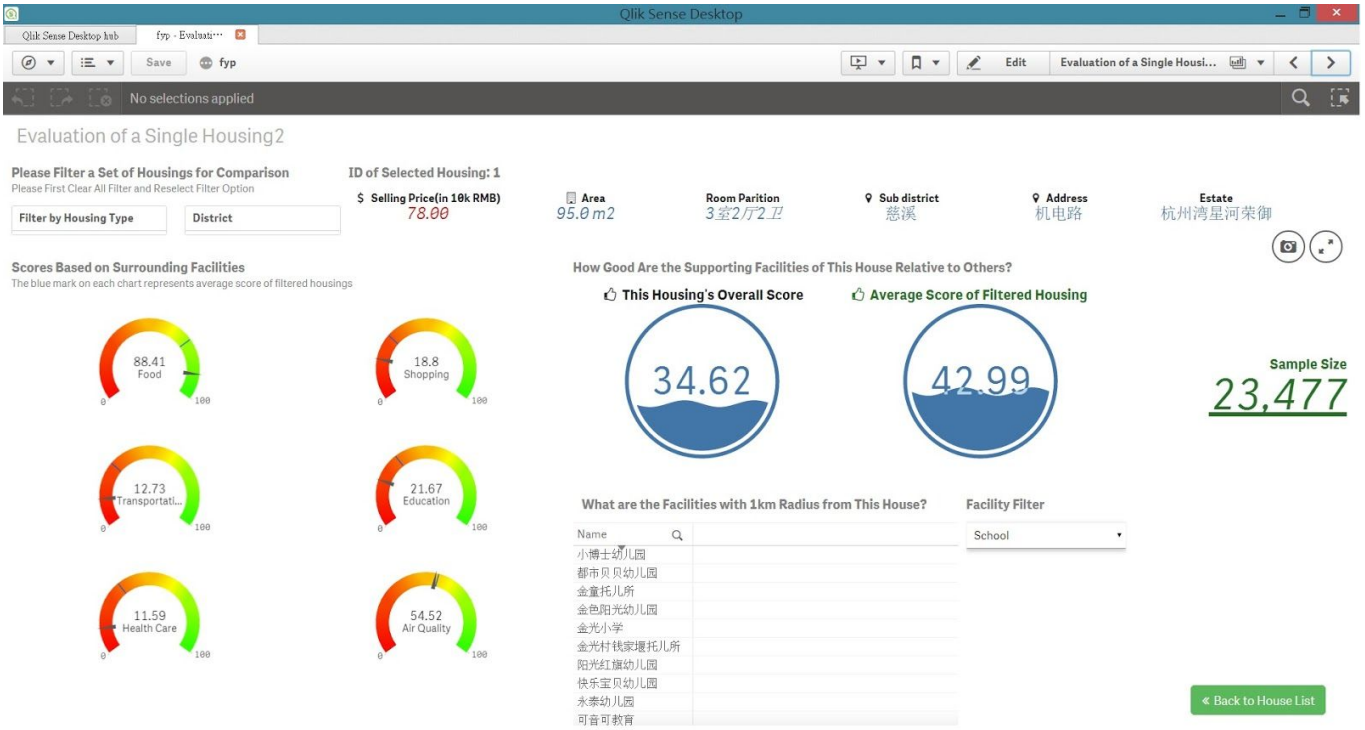


Figure 4.4(b): 2nd Page for Evaluation of a Single Housing

Figure 4.4(b) shows 2nd that help a user to evaluate a single house. The focus of this page is evaluating supporting facilities surrounding the targeted real-estate relative to other real estates in the same district.

At the top of this page, key information about the selected real-estate is first recapped. Users could identify if a correct house id has been inputted in the previous page.

The filter penal at the top-left corner enable users to specify what types of housings and housings in which districts they would like to compare with when evaluation the selected real-estates. Below the recapped information stand 2 water-tank gauge charts. The green number next to the water-tanks show the number of housings in the database matching the criteria specified in the filter penal

The water tank on the left shows the overall all rating of the selected housing. The one on the right shows the average rating of all houses which match the criteria specified in the filter panel. Mechanism used for calculating these rating could reference to the "Methodologies" section.

On the left of the water-tanks there are 6 gauges. Cursors on these gauges display the rating of supporting facilities surrounding the selected house in different dimension. They are: 1) food, 2) shopping, 3) transportation, 4) education, 5) health care. The last rating correspond to air quality around this selected house. On every gauge, there is a blue mark indicating average rating of all filtered houses.

At the bottom right corner of this page, there is a table listing facilities that are within 1km radius from the selected housing. The drop-down menu next to the table lists different facility types such as “restaurant”, “clinic”, “ school”... By choosing different facility types in the drop-down menu, user could check if there is a certain kind of facilities surrounding the selected house.

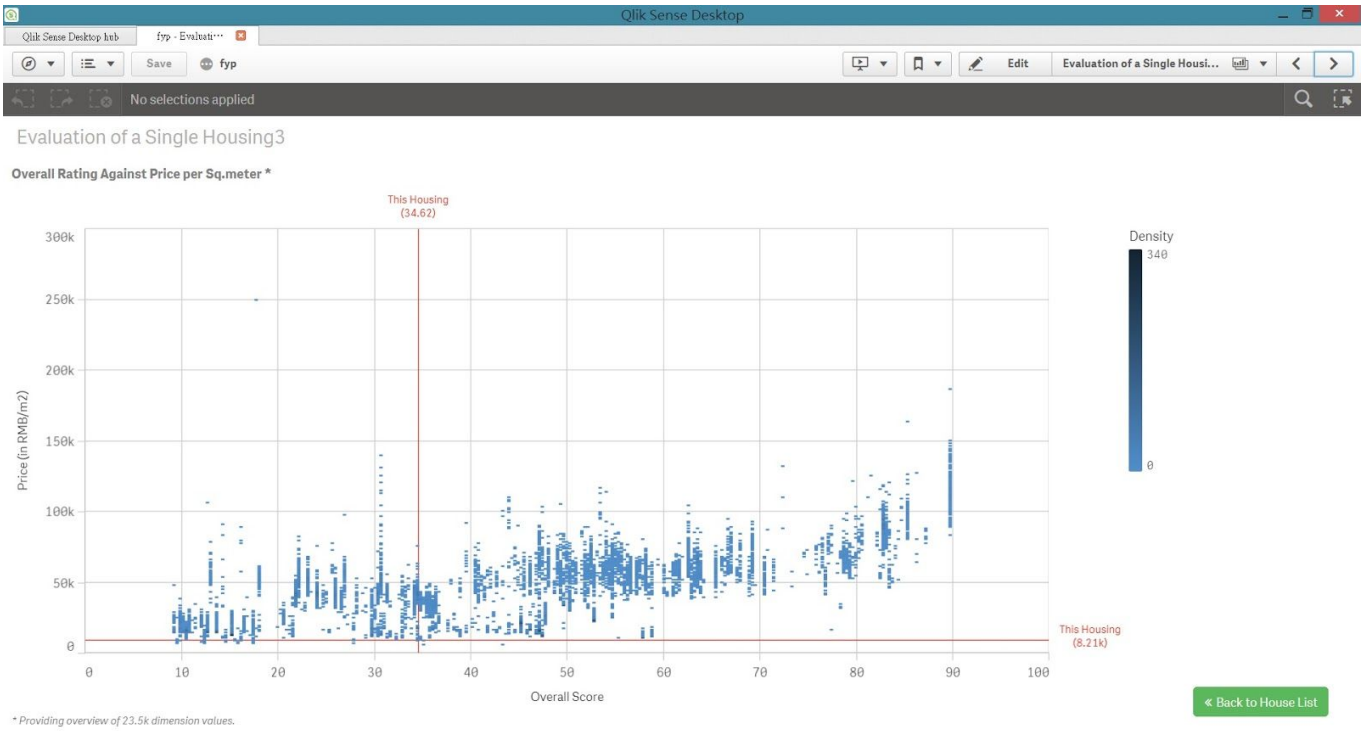


Figure 4.4(c): 3rd Page for Evaluation of a Single Housing

The last dashboard of this platform display a scatter chart. The vertical axis correspond to the price per square meter. The horizontal axis correspond to overall score. Each dot represents a real-estate that matches the filter criteria specified in last 2 pages. A set of perpendicular red lines are drawn to locate the selected housing.

Chapter 5: Discussion

5.1 Interpretation of result

This section discusses how the product of this project, a real-estate information platform help real estate agents communicate with clients and explore suitable housings for them. The following discussion would based on an imaginary client with 3 million budget looking for a ordinary housing (普通住宅) with 2 rooms for investment. The client does not have specific interests on a particular type of housing or district.

5.1.1 Possible Insights From Overview of Housing Supply in Shanghai

Figure 5.1.1(a): Key Performance Indicators Shown in First Page of the Platform

Figure 5.1.1(a) is visualization shown at the homepage of the platform. From this overview, a real-estate agent could make sense of the median price of different types of housing in Shanghai quickly. By comparing the client's budget with the median prices, the agent could figure if the client's budget is feasible for purchasing a ordinary housing (普通住宅). If it is not so feasible, the agent could manage the client's expectation by informing him earlier. The agent could also recommend the client to consider alternative types of housing or renting a house.

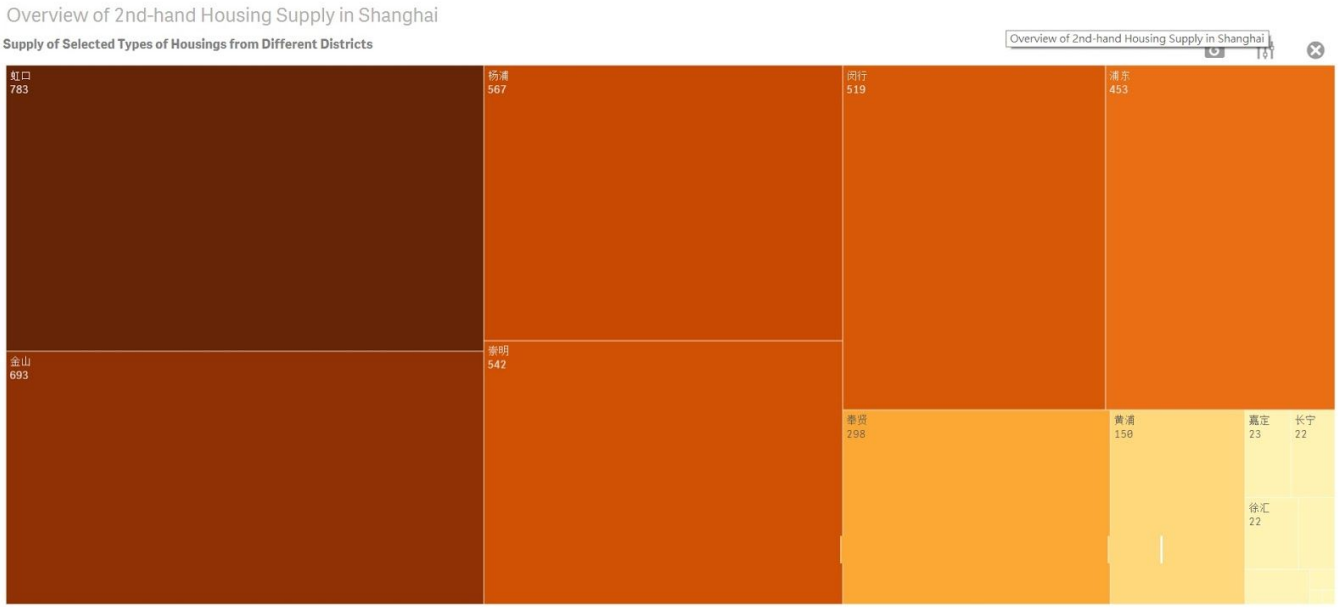


Figure 5.1.1(b): Treemap Shown in First Page of the Platform

Figure 5.1.1(b) is another visualization in the first page. By restricting the housing type to be ordinary housing, the property agent could examine how the supply of ordinary housings are distributed across districts in Shanghai through figure 5.1.1(b). The figure reveals that Hongkou (虹口), Jinshan (金山), Yang Pu (楊浦), Chongming(崇明), Minhang(閔行), Pudong (浦東) are major supply centre of ordinary housings. This could possibly imply more ordinary housings available among these districts for the client to choose. Hence, this figure could guide real-estate agent the starting point for explore suitable housings for the clients given that the client doesn't have distinctive preference.

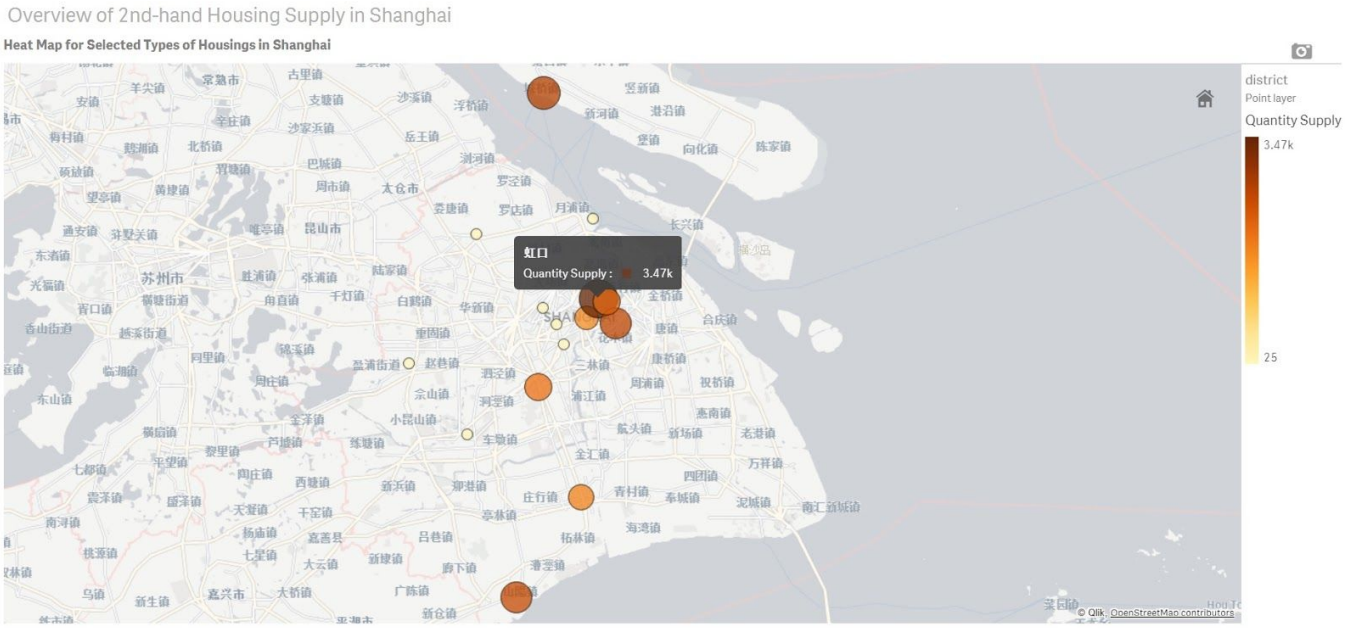


Figure 5.1.1(c): Heat map shown in First Page of the platform

Figure 5.1.1(c) is another visualization in the first page. For those potential buyers who are not familiar with Shanghai, it could be difficult for them to make sense about the exact location of the districts proposed by the previous treemap. Figure 5.1.1(c), a heat map showing the distribution of housing supply in a map, could help real-estate agent inform the client about the exact location of the districts.

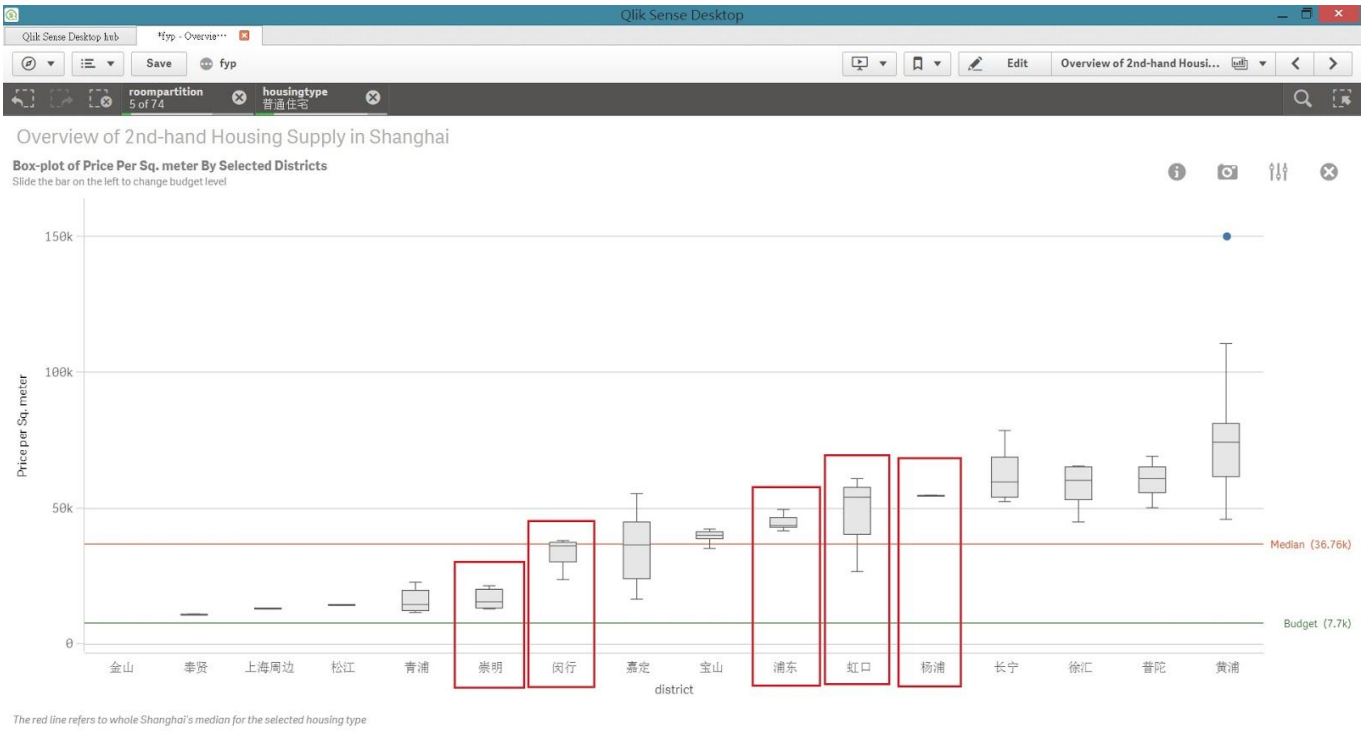


Figure 5.1.1(d): Box-and-whisker Diagram shown in First Page of the platform

In the dashboard “Overview of Housing Supply in Shanghai”, a box-and-whisker plot describing the rough price distribution of different districts is shown. From the figure above, it is clear that among those previously mentioned major supply centres: Hongkou (虹口), Jinshan (金山), Yang Pu (楊浦), Chongming(崇明), Minhang(閔行), Pudong (浦東), Assuming the client prefer saving money more than investing in a particular property market, the real-estate agents could recommend ordinary housings in Chongming for the client.

5.1.2 Possible Insights From Overview of Housing Supply in Chongming

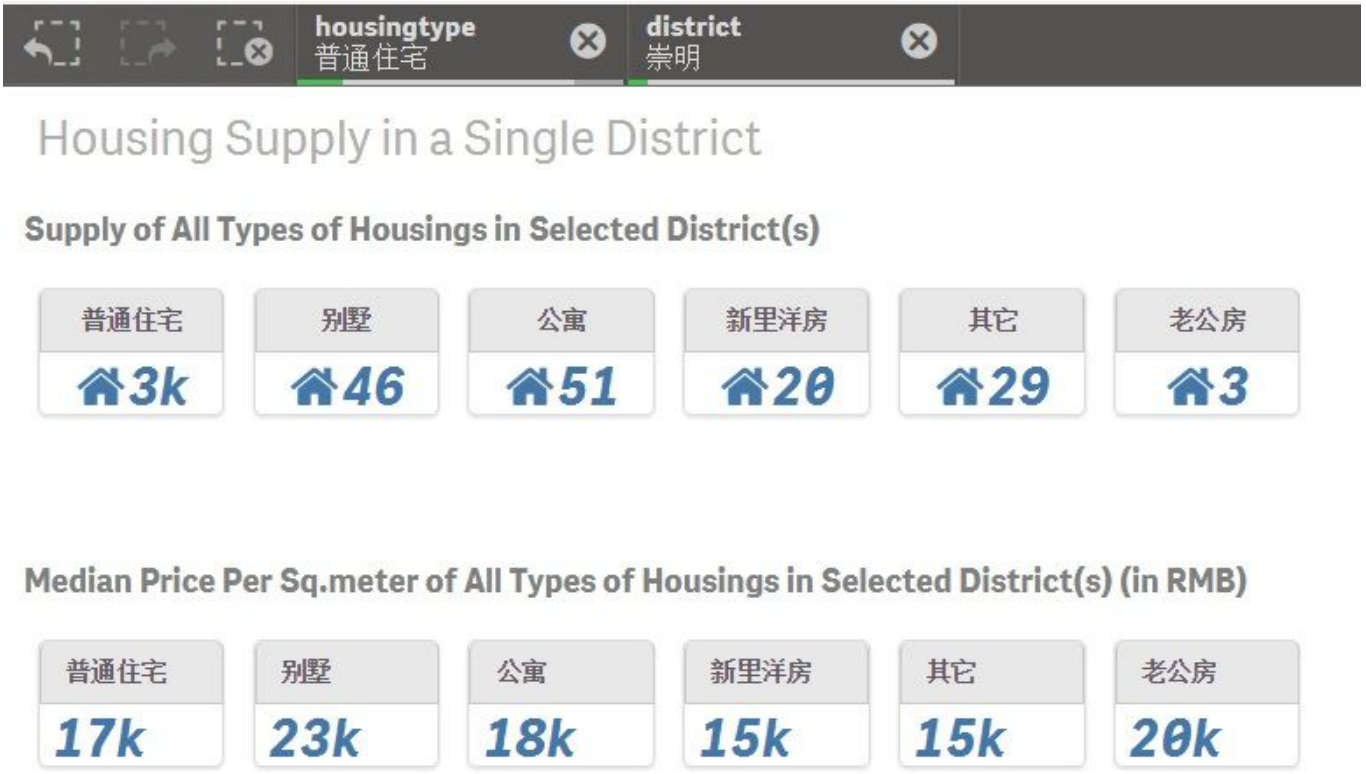


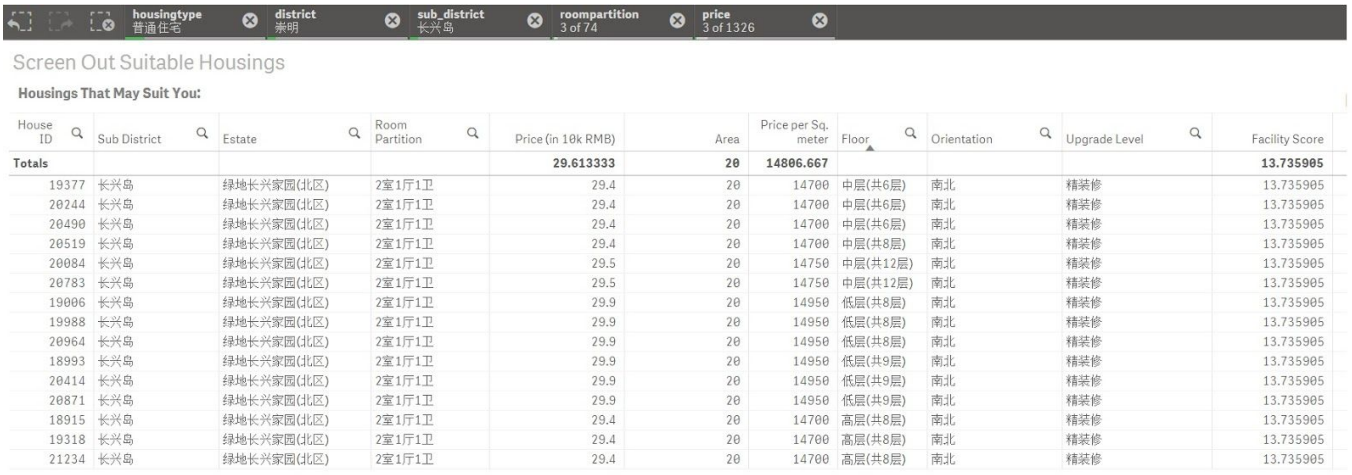
Figure 5.1.2(a): Housing Supply in Chongming

Alternatively, if a client already has clear targeted district, the real-estate agent could directly use the dashboard "Housing Supply in a Single District" by choosing Chongming in the filter panel. The statistics shown here are similar to those previously shown in Figure 5.1.1(a). This page shows the housing supply and median price per square meter information in the Chongming district. The real-estate agent could interpret the statistic in similar ways.

Figure 5.1.2(b): Supply of Selected Types of Housings from Sub_district

Now, the real-estate agent and the client have been guided to explore ordinary housing in Chongming by the data visualization platform. The next step is to reduce the scope of searching into sub-district level. In the dashboard "Housing Supply in a Single District", a treemap is shown also. At this time, the treemap is reflecting the distribution of ordinary housings among different sub-districts in Chongming, but not among districts in Shanghai. According to Figure 5.1.2(b), ordinary housings in Chongming are only available in the sub-district called Changxing Island (長興島). This suggests the real-estate agent and the client should search for suitable housing in Changxing Island.

5.1.3 Filtering out Suitable Housings in Sub-district


housingtype 普通住宅 | district 崇明 | sub_district 长兴岛 | roompartition 3 of 74 | price 3 of 1326

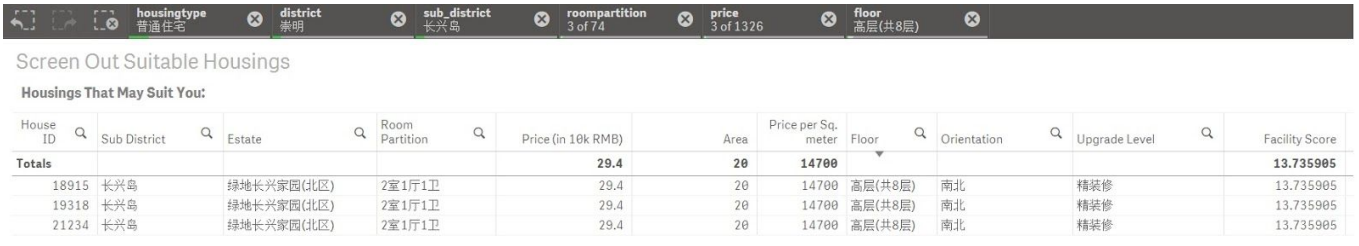
Screen Out Suitable Housings

Housings That May Suit You:

House ID	Sub District	Estate	Room Partition	Price (in 10k RMB)	Area	Price per Sq. meter	Floor	Orientation	Upgrade Level	Facility Score
Totals				29.613333	20	14806.667				13.735905
19377	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	中层(共6层)	南北	精装修	13.735905
20244	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	中层(共6层)	南北	精装修	13.735905
20490	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	中层(共6层)	南北	精装修	13.735905
20519	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	中层(共8层)	南北	精装修	13.735905
20084	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.5	20	14750	中层(共12层)	南北	精装修	13.735905
20783	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.5	20	14750	中层(共12层)	南北	精装修	13.735905
19006	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共8层)	南北	精装修	13.735905
19988	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共8层)	南北	精装修	13.735905
20964	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共8层)	南北	精装修	13.735905
18993	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共9层)	南北	精装修	13.735905
20414	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共9层)	南北	精装修	13.735905
20871	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.9	20	14950	低层(共9层)	南北	精装修	13.735905
18915	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905
19318	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905
21234	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905

Figure 5.1.3(a): Filter Table

In the dashboard “Screen Out Suitable Housings”, the real-estate agent and the client could apply the previously discussed screening criteria. Those criteria include Chongming as “district”, 長興島 as “sub-district”, 2-bedrooms as “Room Partition” and price under 3 million as criteria. If the customer has already decided which housing market or more specifically which sub-district he/she will choose, they can surf this page directly and put on filters on different attributes such as price and facility score to let the platform generate rankings for the housings, i.e. from the lowest price to the highest price and from the highest final score to the lowest final score in an attempt to provide them insights of an economical housing with low price and high score.


housingtype 普通住宅 | district 崇明 | sub_district 长兴岛 | roompartition 3 of 74 | price 3 of 1326 | floor 高层(共8层)

Screen Out Suitable Housings

Housings That May Suit You:

House ID	Sub District	Estate	Room Partition	Price (in 10k RMB)	Area	Price per Sq. meter	Floor	Orientation	Upgrade Level	Facility Score
Totals				29.4	20	14700				13.735905
18915	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905
19318	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905
21234	长兴岛	绿地长兴家园(北区)	2室1厅1卫	29.4	20	14700	高层(共8层)	南北	精装修	13.735905

Figure 5.1.3(b): Filter Results

After discussing with the client, it is assumed that the real-estate discover the client prefers units on upper floor over those on lower floor. This additional screening criteria is applied also. The list of suitable housings are eliminated in to those 3 described in Figure 5.1.3(b). The real-estate agent could now perform more details evaluation of these 3 housings for the clients.

5.1.4 Detail Analysis on Potential Housings

Figure 5.1.4(a): Detailed Analysis

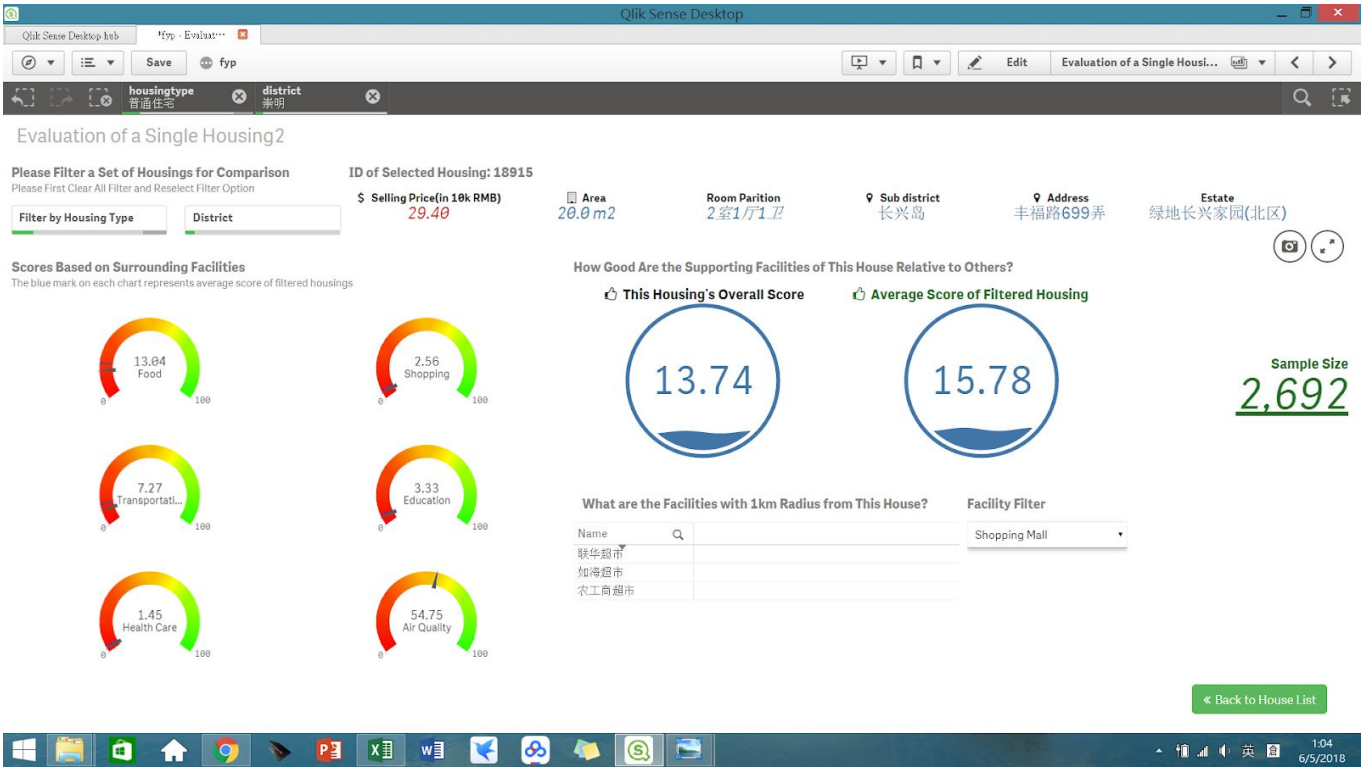


Figure 5.1.4(b): Detailed Analysis

In the dashboard “Evaluation of a Single Housing”, the real-estate agent could retrieve detailed information of a housing by inputting the ID of anyone of the 3 potential housings. Figure 5.1.4(a) and Figure 5.1.4(b) show how these information would be reported. Since those 3 potential housings belong to the same estate, most of their information would be similar. For illustration purpose, reports of only one of them are displayed here and in subsequent explanation.

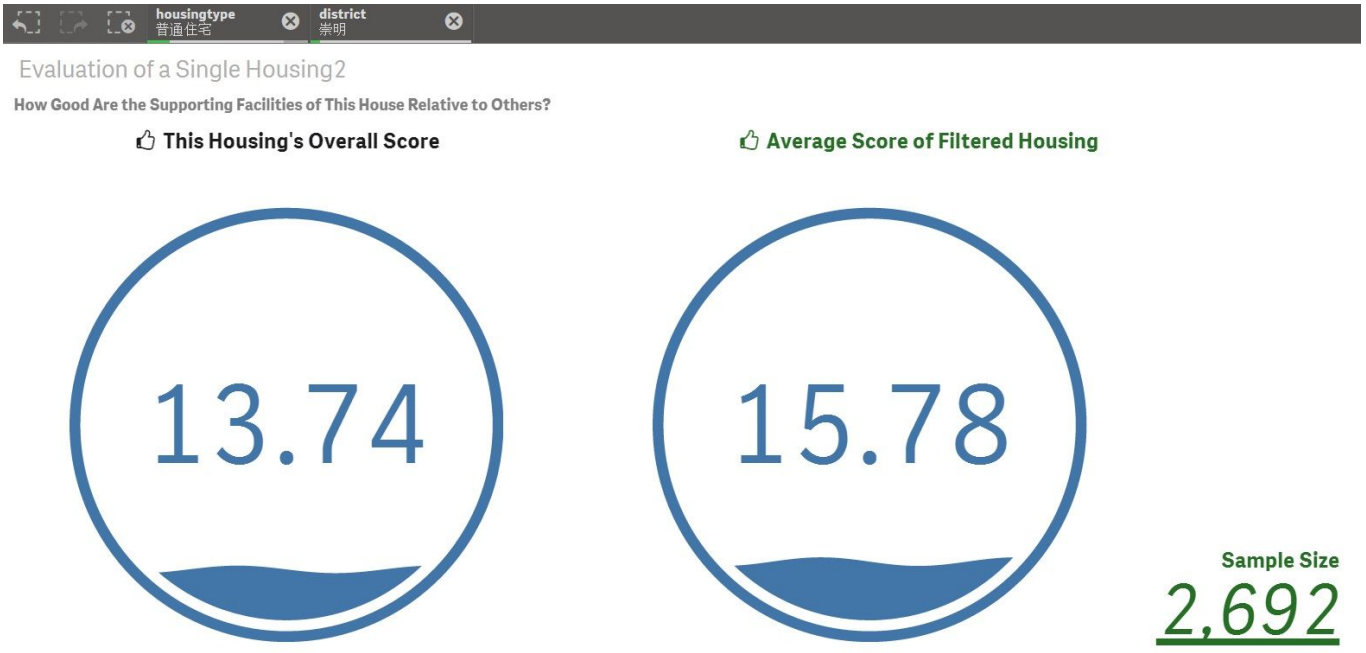


Figure 5.1.4(c): Overall Rating of the Selected Housing’s Supporting Facilities

As mentioned in the methodologies section, the resulting platform rate different housings based on availability of different supporting facilities. In Figure 5.1.4(c), the extremely low rating suggests that there are not many supporting facilities surrounding the selected housing. However, this may actually be a marco-problem concerning the whole district or sub district. The selected housing could be already relatively excellent among its neighbouring. Therefore, this visualization platform would also provide the average rating of all housings within the filtered district for more objective analysis. In this example, it is observed the rating of the selected house is actually very close the district’s average. If the real-estate agent find that the client really want to buy housings in Chongming, the agent could convince him to buy this selected housings with such a relative comparison. The sample size in green color is provided to inform platform users about the representative and reliability of the average rating.

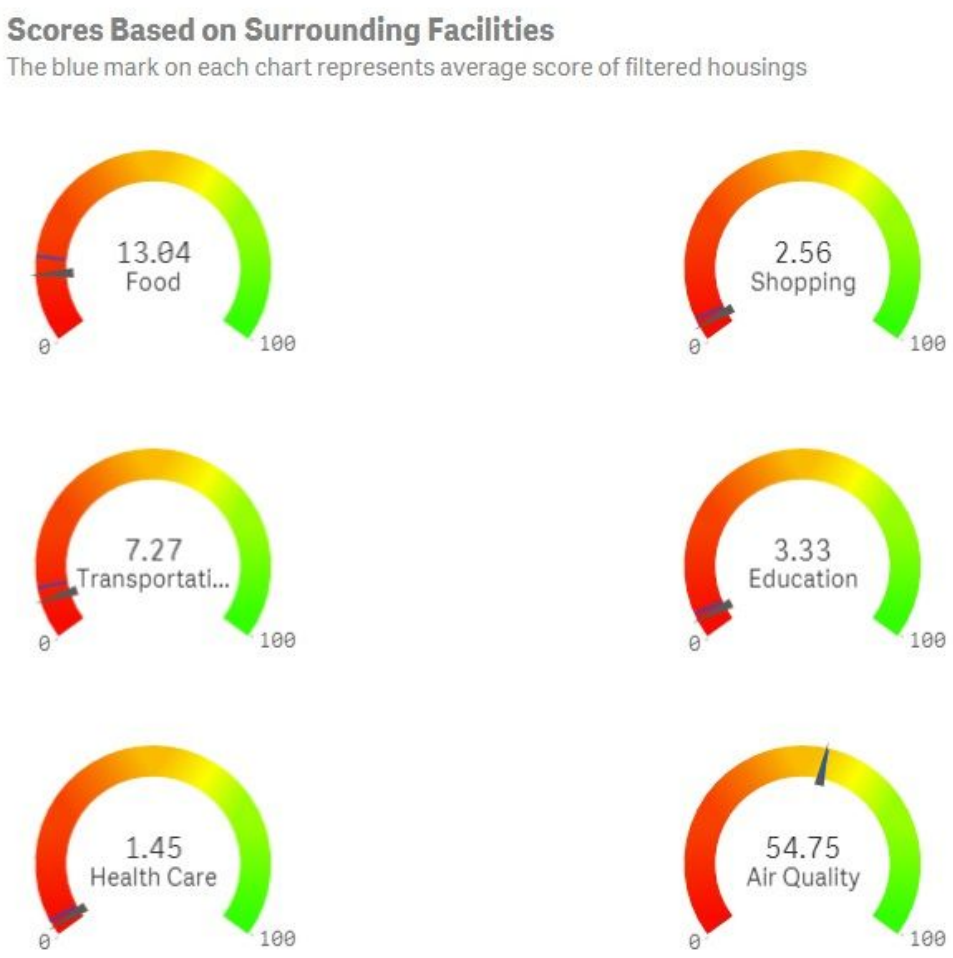


Figure 5.1.4(d): Rating of the Selected Housing in Different Dimension

There are also sub ratings for the selected housing in different dimension with gauge charts. Similarly, average ratings of housings in the same district are also provided. In case the client criticize that the selected housing lack of a kind of facilities, the agent could use the selected housing's performance relative to the average score to convince the client.

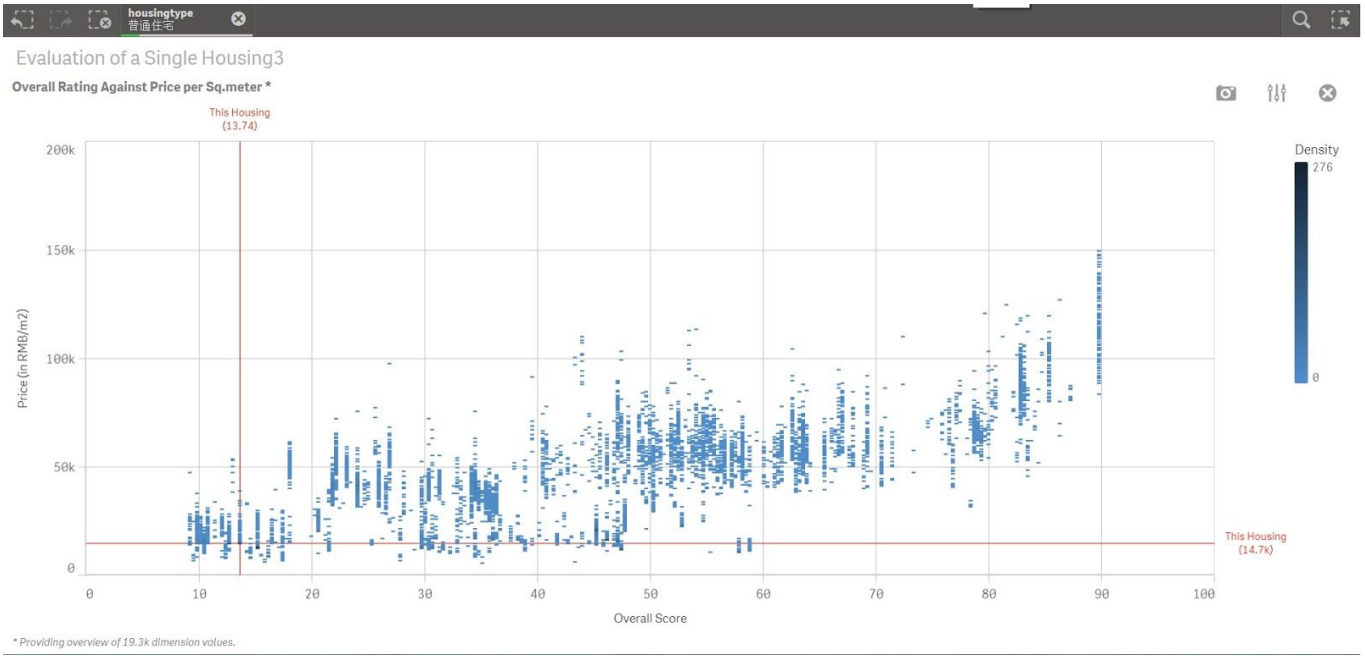


Figure 5.1.4(e): Scatter Plot of Price per m^2 against Score

The last part of the evaluation graph shows a scatter plot with price per square meter against score. Each point corresponds to 1 housing in the filtered district. It is observed that there is positive correlation between the score generated by the platform with the price per square meter. This suggest the score generated by this project's product could reflect the true value of every housing in certain degree. The intersection of the two red line is the point representing the selected housing.

To interpret this graph, the real-estate agent could compare the price of the selected housing with the price of other housings in the same district which share similar rating. That is to say checking if there are any other points along the vertical red line. If there are housings which share similar scores but are priced more expensively, this could indicate that the selected housing could possibly be underpriced. It might be more worth to purchase.

5.2 Limitations

Due to the fact that the acquisition of the data for the platform solely relies on external source, it imposes limitations on the comprehensiveness and accuracy of the data. Ideally, this platform would include as much timely and housing-relevant data as possible. However, this goal cannot be fulfilled fully owing to the constraints on time, nature of websites and availability of detailed data for other possible indicators.

Given that the time flame for this project just lasts for a year, only data from a district, instead of the whole set of Chinese property data, is crawled to be the pioneer data to accelerate the platform building. Some websites have high security protection which shield themselves from data leaks to other parties and thus cannot be crawled; even the website contains the data this project looks for. For other websites with lower protection, the data collected still cannot be updated every second in accordance to the

website changes because of the deficit in manpower to run the codes round-the-clock and the prevention of data collection blocking from the website. However, the data is kept updated in a timely manner for once a week so as to uphold the quality and enlarge the database continuously. Some indicators probably exert influencing power on people's judgment towards housing value but detailed sub-district information from a reliable third party source, such as the Chinese official sub-district noise pollution figures, are lacking. Thus, these indicators cannot be included in the platform.

Besides, the accuracy of the data is not guaranteed. The external data sources have server instability risks. If server of external data source crashes, the platform will then cannot retrieve updated data from them, leaving the platform with obsolete and inaccurate information. Also, the difficulty to do differentiation on scam data which are included in the platform from true data may devalue the platform. Nonetheless, this project chooses data sources from renowned housing websites and Chinese official data sources which are expected to have higher creditworthiness. This helps minimize the number of inaccurate data.

5.3 Delimitations

This project does not focus and obtain housing data from rental housing market as we want to use the property selling price as the single representative of the housing value instead of the inclusion of rental price.

Chapter 6: Implication

This platform could be potentially beneficial to Justar, the Government and the platform end users.

Justar can make use of the platform to identify the indicators users concern about when they consider buying properties. It can leverage the property scoring system to find out which dimension the end users treasure most by getting data from the search result in future, and construct respective campaigns and service fulfilling the end users needs.

The platform end users can make use of this platform to find housing. As this platform integrated data from different websites, they are more likely to find the house satisfying their needs via one platform instead of screening different platforms, and thus saving them lots of time. Also, they can refer to the property scoring system to screen out the factors they concern in the selection of flats.

The government can further modify the city planning by looking at the distribution of facilities nearby the houses via the platform. For instance, the government can know a region is lacking a type of facilities, e.g. hospital and they may consider building the hospital in that region. Viewing the average air quality in different stations, the government can formulate policies to improve the regions with low air quality such as increasing the greening area.

Chapter 7: Future Work

As this platform is the prototype for housing data visualization, only shanghai housing data was selected for the time being. To expand its usability to the whole China, data from all China districts could be collected. Besides, dynamic data can be included to regularly update the information in platform for better user experience.

Chapter 8: Conclusions

This project aimed at creating a new housing search platform different from the existing real estate websites. Various researches were done to select the best attributes and data visualization tool. This platform includes a wider range of indicators, larger and unified data base, more types of trend analysis and a novel approach to do spider diagram construction for search result ranking. With this platform design, it is hope that this platform can create more reference value and convenience to users and become the dominant housing search website in the future.

Chapter 9: References

20.6. urllib2 - extensible library for opening URLs. (n.d.). Retrieved November 22, 2017, from https://docs.python.org/2/library/urllib2.html

7.2. re - Regular expression operations. (n.d.). Retrieved November 22, 2017, from https://docs.python.org/2/library/re.html

Alexander, K. (2008). RDF/JSON: A specification for serialising RDF in JSON. CEUR Workshop Proceedings, 368, CEUR Workshop Proceedings, 2008, Vol.368.

Beautiful Soup Documentation. (n.d.). Retrieved November 22, 2017, from https://www.crummy.com/software/BeautifulSoup/bs4/doc/

Business Application Research Center. (2017). BI Trend Monitor 2018. Retrieved November 25, 2017, from http://barc-research.com/wp-content/uploads/2017/11/BARC-BI_Trend_Monitor_2018-Online.pdf

Carey, N. (2005). Establishing pedestrian walking speeds. *Karen Aspelin, Portland State University*, *1*(01).

Chan, & Yao. (2008). Air pollution in mega cities in China. Atmospheric Environment, 42(1), 1-42.

Chay, K., & Greenstone, M. (2005). Does Air Quality Matter? Evidence from the Housing Market. Journal of Political Economy, 113(2), 376-424.

Dalcín, Paz, Storti, & D'elía. (2008). MPI for Python: Performance improvements and MPI-2 extensions. Journal of Parallel and Distributed Computing, 68(5), 655-662.

Feng, X., & Humphreys, B. (2016). Assessing the Economic Impact of Sports Facilities on Residential Property Values: A Spatial Hedonic Approach. Journal of Sports Economics. doi:10.1177/1527002515622318

Fu, Y., Xiong, H., Ge, Y., Yao, Z., Zheng, Y., & Zhou, Z. (2014). Exploiting geographic dependencies for real estate appraisal. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD 14. doi:10.1145/2623330.2623675

Geng, Bao, & Liang. (2015). A study of the effect of a high-speed rail station on spatial variations in housing price based on the hedonic model. Habitat International, 49, 333-339.

Jing Han, Haihong E, Guan Le, & Jian Du. (2011). Survey on NoSQL database. Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on, 363-366.

Lee, Jen-Sin, & Kuo, Chin-Tai. (2010). Momentum Effect and Market Conditions. Chinese Economy, 43(2), 70-94

Li, H. (2007). International linkages of the Chinese stock exchanges: A multivariate GARCH analysis. Applied Financial Economics, 17(4), 285-297.

Nguyen-Hoang, & Yinger. (2011). The capitalization of school quality into house values: A review. Journal of Housing Economics, 20(1), 30-48.

Stodder, D. (2016) Top Trends in BI and Self-Service Visual Analytics. Retrieved November 25, 2017, from https://tdwi.org/articles/2016/12/16/top-trends-in-bi-and-self-service-visual-analytics.aspx

Wang Yaowu, Olofsson Thomas, Shen Geoffrey Qiping, & Bai Yong. (2015). Analysis on Impact Factors of Real Estate Price Based on the Factor Analysis Method. In ICCREM 2015 - Environment and the Sustainable Building - Proceedings of the 2015 International Conference on Construction and Real Estate Management, August 11-12, 2015, Luleå, Sweden (p. 1). American Society of Civil Engineers (ASCE).

Wen, H., Zhang, Y., & Zhang, L. (2014). Do educational facilities affect housing price? An empirical study in Hangzhou, China. Habitat International, 42, 155-163. doi:10.1016/j.habitatint.2013.12.004

Williamson, C., & Shneiderman, B. (1992). The dynamic HomeFinder: Evaluating dynamic queries in a real-estate information exploration system. Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 338-346.

Xu, X. E., & Chen, T. (2012). The effect of monetary policy on real estate price growth in China. Pacific-Basin Finance Journal, 20(1), 62-77. doi:10.1016/j.pacfin.2011.08.001

Yang, L., & Zhiqiang, H. (2012). On Correlation between RMB Exchange Rate and Real Estate Price based on Financial Engineering. Systems Engineering Procedia, 3, 146-152. doi:10.1016/j.sepro.2011.11.020

中华人民共和国环境保护部. (2012). *《环境空气质量指数（AQI）技术规定（试行）》*. HJ 633-2012

财经网产经. (2017). *艾瑞咨询最新数据出炉，房天下9月继续霸榜*. Retrieved November 23, 2017, from http://www.caijing.com.cn/20171027/4350678.shtml

周海波. (2009). 房地产价格影响因素的实证研究. *海南大學學報（人文社會科學版）*, *27*(5), 537-543

Chapter 10: Appendix

10.1 Code

10.1.1 Code for AQI crawling

```
import xlrd
import time
from random import randint
import urllib.request, urllib.error, urllib.parse
from bs4 import BeautifulSoup
import re
import csv
import random
import requests
import schedule
import threaded
from threading import Event, Thread, Timer

csv_file1 = "C:/airdata1.csv"
csv_file2 = "C:/airdata2.csv"
csv_file4 = "C:/airdata4.csv"
csv_file5 = "C:/airdata5.csv"
csv_file6 = "C:/airdata6.csv"
csv_file7 = "C:/airdata7.csv"
csv_file8 = "C:/airdata8.csv"
csv_file9 = "C:/airdata9.csv"
csv_file10 = "C:/airdata10.csv"
csv_file11 = "C:/airdata11.csv"
csv_file12 = "C:/airdata12.csv"


field_name1 = ["jinganjiancezhan"]
field_name2 = ["luwanshizhuanfuxiao"]
field_name4 = ["putuojiancezhan"]
field_name5 = ["xuhuishangshida"]
field_name6 = ["pudongjiancezhan"]
field_name7 = ["hongkouliangcheng"]
field_name8 = ["yangpusipiao"]
field_name9 = ["pudongzhangjiang"]
field_name10 = ["pudongchuansha"]
field_name11 = ["qingpudianshanhu"]
field_name12 = ["consulate"]

with open(csv_file1, "w", encoding = "utf8") as output:
    writer = csv.writer(output, delimiter = ',', lineterminator = "\n")
    writer.writerow(field_name1)

with open(csv_file2, "w", encoding = "utf8") as output:
    writer = csv.writer(output, delimiter = ',', lineterminator = "\n")
    writer.writerow(field_name2)
```

```
file_location = "C:/shanghaiair_link.xlsx"
workbook = xlrd.open_workbook(file_location)
sheet = workbook.sheet_by_index(0)

row_array=[]
for i in range(0,13):
    list=sheet.cell_value(i,1)
    row_array.append(list)

data_list1=[]
data_list2=[]
data_list4=[]
data_list5=[]
data_list6=[]
data_list7=[]
data_list8=[]
data_list9=[]
data_list10=[]
data_list11=[]
data_list12=[]

def crawl1():
    while(True):
        url1=row_array[1]
        headers = {'User-Agent':'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_
        request1 = urllib.request.Request(url1, None, headers)
        response1 = urllib.request.urlopen(request1).read()
        soup1 = BeautifulSoup(response1, "html.parser")
        jinganjiancezhan=''
        jinganjiancezhan=soup1.find("td", {"id":"cur_pm25"})
        jinganjiancezhan=str(jinganjiancezhan)
        jinganjiancezhan=jinganjiancezhan[72:]
        jinganjiancezhan=re.sub(re.compile("<.*?>"), "", jinganjiancezhan)
        print (jinganjiancezhan)
        data_list1.append((jinganjiancezhan))
        with open(csv_file1, "a", encoding = "utf8") as output1:
            writer = csv.writer(output1)
            writer.writerow([jinganjiancezhan])
        time.sleep(3600)


t1 = Timer(1, crawl1)
t2 = Timer(1, crawl2)
t4 = Timer(1, crawl4)
t5 = Timer(1, crawl5)
t6 = Timer(1, crawl6)
t7 = Timer(1, crawl7)
t8 = Timer(1, crawl8)
t9 = Timer(1, crawl9)
t10 = Timer(1, crawl10)
t11 = Timer(1, crawl11)
t12 = Timer(1, crawl12)


t1.start()
t2.start()
t4.start()
t5.start()
t6.start()
t7.start()
t8.start()
t9.start()
t10.start()
t11.start()
t12.start()
```

10.1.2 Code for Crawling Housing Data

10.1.2.1 scrapy_crawl.py (spider of the scrapy project)

```
# Author: Yeung King Yiu
# This this the spider used to crawl housing data

import scrapy, winsound
from capstone_crawl2.items import estate_item
from scrapy_splash import SplashRequest
from random import randint, shuffle


class WebpageSpider(scrapy.Spider):
    name = "estate_spider2"
    start_urls = ["https://shanghai.anjuke.com/sale/"]
    main_page = "https://shanghai.anjuke.com/"

    # Visit the catalog of a single district
    def parse(self, response):
        url_list = response.xpath("//div[(@class = \"items\") and (position() = 1)]/span[@class = \"elems-l\"]/a/@href").extract()
        # randomize district order
        shuffle(url_list)
        for link in range(len(url_list)):
            yield scrapy.Request(url=url_list[link], callback=self.parse_sub_district)

    # Visit the catalog of a sub_district
    def parse_sub_district(self, response):
        url_list = response.xpath("//div[@class = \"sub-items\"]//a/@href").extract()
        # randomize sub_district order
        shuffle(url_list)
        for link in range(len(url_list)):
            request = scrapy.Request(url=url_list[link]+"p50/", callback=self.parse_last_page)
            request.meta['link'] = url_list[link]
            yield request

    def parse_last_page(self, response):
        base_url = response.meta['link']
        page_url = []

        # find maximum page number in a sub_district
        max_number = response.xpath("//div[contains(@class, \"page\")]//i[last()-1]/text()").extract_first()
        if max_number != "50":
            max_number = response.xpath("//div[contains(@class, \"page\")]//a[last()]/text()").extract_first()

        max_number = int(max_number)

        for number in range(1, max_number+1):
            page_url.append(base_url + "p" + str(number) + "/")

        shuffle(page_url)
        for element in range(len(page_url)):
            yield scrapy.Request(url=page_url[element], callback=self.parse_page)

    # real function to crawl from housing catalog
    def parse_page(self, response):
        # get all listing housing on a single page
        # crawl pages are split into batches of size 20 to encounter dynamic links
        # crawling for batch1
        link_list1 = response.xpath("//li[(contains(@class, \"list-item\")) and (position() <= 20)]//div[(@class = \"house-title\")
        shuffle(link_list1)
        for element in range(len(link_list1)):
            yield SplashRequest(url=link_list1[element], callback=self.parse_details,
                                args={'wait': 1, 'timeout': 180}, endpoint='render.html')

        # crawling for batch2
        link_list2 = response.xpath("//li[(contains(@class, \"list-item\")) and (position() > 20) and (position() <= 40)]//div[(@cl
        shuffle(link_list2)
        for element in range(len(link_list2)):
            yield SplashRequest(url=link_list2[element], callback=self.parse_details,
                                args={'wait': 1, 'timeout': 180}, endpoint='render.html')

        # crawling for batch3
        link_list3 = response.xpath("//li[(contains(@class, \"list-item\")) and (position() > 40)]//div[(@class = \"house-title\")]
        shuffle(link_list3)
        for element in range(len(link_list3)):
            yield SplashRequest(url=link_list3[element], callback=self.parse_details,
                                args={'wait': 1, 'timeout': 180}, endpoint='render.html')

        winsound.Beep(2000, 1000)

    # sub function to crawl details of every listed housing
    def parse_details(self, response):
        # forming a dictionary of xpath
        field_name = ["estate", "location_part1", "location_part2", "yearbuild", "housingtype",


        item["sub_districtavg"] = response.xpath(xpath_dict["sub_districtavg"]).extract_first()
        #print(item["sub_districtavg"])
        #print(type(item["sub_districtavg"]))

        # After crawling a housing page,
        # there are 1/3 chance that the spider would go back to the main page of the platform
        # to prevent robotic accessing pattern
        if (randint(-67, 33) > 0):
            SplashRequest(url=self.main_page, args={'wait': 1}, endpoint='render.html')
        return item
```

10.1.2.2 items.py

```
# -*- coding: utf-8 -*-

# Define here the models for your scraped items
#
# See documentation in:
# https://doc.scrapy.org/en/latest/topics/items.html

# Author: Yeung King Yiu

import scrapy


class estate_item(scrapy.Item):
    # define the fields for your item here like:
    # name = scrapy.Field()

    # 屋苑
    estate = scrapy.Field()

    # 分區小區
    sub_district = scrapy.Field()

    # 分區
    district = scrapy.Field()

    # 地址
    address = scrapy.Field()

    yearbuild = scrapy.Field()

    housingtype = scrapy.Field()

    # 單位間隔
    roompartition = scrapy.Field()

    pricepersqmeter = scrapy.Field()

    area = scrapy.Field()

    # 座向
```

10.1.2.3 pipelines.py

```
# -*- coding: utf-8 -*-

# Define your item pipelines here
#
# Don't forget to add your pipeline to the ITEM_PIPELINES setting
# See: https://doc.scrapy.org/en/latest/topics/item-pipeline.html

# Author: Yeung King Yiu

import re
import json, csv

class CapstoneCrawlPipeline(object):
    # remove empty space character character in data
    def remove_empty(self, field):
        sample = []
        if isinstance(field, str):
            field = field.replace("\n", "")
            field = field.replace("\t", "")
            field = field.replace("\v", "")
            field = field.replace("\r", "")
            field = field.replace(" ", "")
        elif isinstance(field, type(sample)):
            for element in range(len(field)):
                field[element] = field[element].replace("\n", "")
                field[element] = field[element].replace("\t", "")
                field[element] = field[element].replace("\v", "")
                field[element] = field[element].replace("\r", "")
                field[element] = field[element].replace(" ", "")
        return field

    # function to check if a numerical data exist or not
    # if it exists, chinese unit would be removed
    def clean_numeric(self, data):
        if isinstance(data, str):
            if re.search(r'[0-9]+(\.[0-9]*)?', data):
                data = re.sub(r'[^0-9\.]', "", data)
        return data
```

```
# function for joining results obtained by extract() from a list into a stri
def list_to_string(self, data_list, separator):
    sample = []
    if isinstance(data_list, type(sample)):
        data_list = separator.join(data_list)
    return data_list

# a json file would be created and opened when the spider start crawling
def open_spider(self, spider):
    self.file = open("data.js", "a", encoding="utf-8")

# the json file would be closed when the spider finishes crawling
def close_spider(self, spider):
    self.file.close()

# final function to output result
def process_item(self, item, spider):
    item['estate'] = self.remove_empty(item['estate'])

    item['district'] = self.remove_empty(item['district'])

    item['sub_district'] = self.remove_empty(item['sub_district'])

    item['address'] = self.list_to_string(item['address'], "")
    item['address'] = self.remove_empty(item['address'])
    item['address'] = item['address'].replace("—", "")

    item['yearbuild'] = self.clean_numeric(item['yearbuild'])

    item['housingtype'] = self.remove_empty(item['housingtype'])

    item['roompartition'] = self.remove_empty(item['roompartition'])

    # in unit 元/平方米
    item['pricepersqmeter'] = self.remove_empty(item['pricepersqmeter'])
    item['pricepersqmeter'] = self.clean_numeric(item['pricepersqmeter'])

    item['area'] = self.remove_empty(item['area'])
    item['area'] = self.clean_numeric(item['area'])

    # store information into the json file after crawling a housing item
    line = json.dumps(dict(item)) + "\n"
    self.file.write(line)
    return item
```

10.1.2.4 settings.py

```
# -*- coding: utf-8 -*-

# Scrapy settings for capstone_crawl project
#
# For simplicity, this file contains only settings considered important or
# commonly used. You can find more settings consulting the documentation:
#
#     https://doc.scrapy.org/en/latest/topics/settings.html
#     https://doc.scrapy.org/en/latest/topics/downloader-middleware.html
#     https://doc.scrapy.org/en/latest/topics/spider-middleware.html

# Author: Yeung King Yiu

BOT_NAME = 'capstone_crawl2'

SPIDER_MODULES = ['capstone_crawl2.spiders']
NEWSPIDER_MODULE = 'capstone_crawl2.spiders'

SPLASH_URL = 'http://192.168.99.100:8050'
#SPLASH_COOKIES_DEBUG = True
DUPEFILTER_CLASS = 'scrapy_splash.SplashAwareDupeFilter'

# Crawl responsibly by identifying yourself (and your website) on the user-agent
#USER_AGENT = 'capstone_crawl (+http://www.yourdomain.com)'

# Obey robots.txt rules
ROBOTSTXT_OBEY = False

# Configure maximum concurrent requests performed by Scrapy (default: 16)
# CONCURRENT_REQUESTS = 8

# Configure a delay for requests for the same website (default: 0)
# See https://doc.scrapy.org/en/latest/topics/settings.html#download-delay
# See also autothrottle settings and docs
DOWNLOAD_DELAY = 4
RANDOMIZE_DOWNLOAD_DELAY = True

# The download delay setting will honor only one of:
#CONCURRENT_REQUESTS_PER_DOMAIN = 4
#CONCURRENT_REQUESTS_PER_IP = 16
```

```
# Disable cookies (enabled by default)
COOKIES_ENABLED = False

# Disable Telnet Console (enabled by default)
#TELNETCONSOLE_ENABLED = False

# Override the default request headers:
DEFAULT_REQUEST_HEADERS = {
   'Referer': 'http://www.baidu.com/'
}

RETRY_ENABLED = True
RETRY_TIMES = 5

# Enable or disable spider middlewares
# See https://doc.scrapy.org/en/latest/topics/spider-middleware.html
#SPIDER_MIDDLEWARES = {
#    'capstone_crawl.middlewares.CapstoneCrawlSpiderMiddleware': 543,
#}

USER_AGENT_LIST = "C:/Users/KingYiu/PycharmProjects/capstone_crawl/capstone_craw

# Enable or disable downloader middlewares
# See https://doc.scrapy.org/en/latest/topics/downloader-middleware.html
DOWNLOADER_MIDDLEWARES = {
    'scrapy.contrib.downloadermiddleware.useragent.UserAgentMiddleware': None,
    'random_useragent.RandomUserAgentMiddleware': 400,
    'scrapy_splash.SplashCookiesMiddleware': 723,
    'scrapy_splash.SplashMiddleware': 725,
    'scrapy.downloadermiddlewares.httpcompression.HttpCompressionMiddleware': 81
}

# Enable or disable extensions
# See https://doc.scrapy.org/en/latest/topics/extensions.html
#EXTENSIONS = {
#    'scrapy.extensions.telnet.TelnetConsole': None,
#}

# Configure item pipelines
# See https://doc.scrapy.org/en/latest/topics/item-pipeline.html
ITEM_PIPELINES = {
    'capstone_crawl2.pipelines.CapstoneCrawlPipeline': 300,
}

# Enable and configure the AutoThrottle extension (disabled by default)
# See https://doc.scrapy.org/en/latest/topics/autothrottle.html
#AUTOTHROTTLE_ENABLED = True
# The initial download delay
AUTOTHROTTLE_START_DELAY = 3
# The maximum download delay to be set in case of high latencies
#AUTOTHROTTLE_MAX_DELAY = 60
# The average number of requests Scrapy should be sending in parallel to
# each remote server
#AUTOTHROTTLE_TARGET_CONCURRENCY = 1.0
# Enable showing throttling stats for every response received:
#AUTOTHROTTLE_DEBUG = False

# Enable and configure HTTP caching (disabled by default)
# See https://doc.scrapy.org/en/latest/topics/downloader-middleware.html#httpcac
#HTTPCACHE_ENABLED = True
#HTTPCACHE_EXPIRATION_SECS = 0
#HTTPCACHE_DIR = 'httpcache'
#HTTPCACHE_IGNORE_HTTP_CODES = []
#HTTPCACHE_STORAGE = 'scrapy.extensions.httpcache.FilesystemCacheStorage'

HTTPCACHE_STORAGE = 'scrapy_splash.SplashAwareFSCacheStorage'
```

10.1.3 Code for Obtaining Latitude and Longitude of Estates (google_estate_coor.py)

```
# Author: Yeung King Yiu
"""
This purpose of this python program is to obtain coordinate of all distinct addresses/estates found in
the hosing data which are stored as a .csv file through google geocode API
"""

import googlemaps, csv, json, time

# set path for import and export csv
path_csv = "C:/Users/KingYiu/PycharmProjects/capstone_crawl2/capstone_crawl2/distinct_estate.tsv"
coor_csv = "distinct_estate_coor.csv"

# write the header of out put csv first:
with open(coor_csv, "w", encoding="utf-8") as write_head:
    writer = csv.writer(write_head, delimiter=",")
    writer.writerow(["estate", "district", "sub_district", "address", "lat", "lng"])
write_head.close()

# connect to google map server
gmap = googlemaps.Client(key="AIzaSyCMk2nGKkuocVRefiOobdzJRG1Tjvr-gps")

# read address from csv
with open(path_csv, "r", encoding="utf-8") as reading:
    reader = csv.reader(reading, delimiter="\t")
    full_json_list = []
    for link in reader:
        # ignore header line
        if link[1] == "estate":
            continue
        else:
            # transform address for better search
            address_info = link
            search_key_base = address_info[0]
            # use address + sub_district to find coor
            search_key = search_key_base + "," + address_info[2]

            # check and get coordinate of different addresses
            geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")

            # use alternative search methods if the previous one is unidentified

            tried_alternative = 0
            while(geocode_result is None or len(geocode_result) == 0):
                # search with address + district
                if tried_alternative == 0:
                    search_key = search_key_base + "," + address_info[3]
                    geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
                    tried_alternative += 1

                # search with estate + sub_district
                elif tried_alternative == 1:
                    search_key = search_key_base + "," + address_info[1]
                    geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
                    tried_alternative += 1

                # search with estate + district
                elif tried_alternative == 2:
                    search_key = search_key_base
                    geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
                    tried_alternative += 1
                else:
                    tried_alternative += 1
                    break

            # assign "" to lat and lng if the location is still unidentified
            if tried_alternative == 4:
                lat = lng = ""
            else:
                lat = geocode_result[0]['geometry']['location']['lat']
                lng = geocode_result[0]['geometry']['location']['lng']

            print(lat)
            print(lng)

            # write into csv file
            address_info.append(lat)
            address_info.append(lng)
            print(address_info)
            with open(coor_csv, "a", encoding="utf-8") as write_coor:
                writer = csv.writer(write_coor, delimiter=",")
                writer.writerow(address_info)
```

10.1.4 Code for Obtaining Facilities Surrounding an Estate (google_facilities.py)

```
# Author: Yeung King Yiu
"""
This purpose of this python program is to obtain coordinate of all distinct addresses/estates found in
the hosing data which are stored as a .csv file through google geocode API
"""

import googlemaps, csv, json, time

# set path for import and export csv
path_csv = "C:/Users/KingYiu/PycharmProjects/capstone_crawl2/capstone_crawl2/distinct_estate.tsv"
coor_csv = "distinct_estate_coor.csv"

# write the header of out put csv first:
with open(coor_csv, "w", encoding="utf-8") as write_head:
    writer = csv.writer(write_head, delimiter=",")
    writer.writerow(["estate", "district", "sub_district", "address", "lat", "lng"])
write_head.close()

# connect to google map server
gmap = googlemaps.Client(key="AIzaSyCMk2nGKkuocVRefiOobdzJRG1Tjvr-gps")

# read address from csv
with open(path_csv, "r", encoding="utf-8") as reading:
    reader = csv.reader(reading, delimiter="\t")
    full_json_list = []
    for link in reader:
        # ignore header line
        if link[1] == "estate":
            continue
        else:
            # transform address for better search
            address_info = link
            search_key_base = address_info[0]
            # use address + sub_district to find coor
            search_key = search_key_base + "," + address_info[2]

            # check and get coordinate of different addresses
            geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")

            # use alternative search methods if the previous one is unidentified
```

```
tried_alternative = 0
while(geocode_result is None or len(geocode_result) == 0):
    # search with address + district
    if tried_alternative == 0:
        search_key = search_key_base + "," + address_info[3]
        geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
        tried_alternative += 1

    # search with estate + sub_district
    elif tried_alternative == 1:
        search_key = search_key_base + "," + address_info[1]
        geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
        tried_alternative += 1

    # search with estate + district
    elif tried_alternative == 2:
        search_key = search_key_base
        geocode_result = gmap.geocode(address=search_key, language="zh-CN", region="cn")
        tried_alternative += 1
    else:
        tried_alternative += 1
        break

# assign "" to lat and lng if the location is still unidentified
if tried_alternative == 4:
    lat = lng = ""
else:
    lat = geocode_result[0]['geometry']['location']['lat']
    lng = geocode_result[0]['geometry']['location']['lng']

print(lat)
print(lng)

# write into csv file
address_info.append(lat)
address_info.append(lng)
print(address_info)
with open(coor_csv, "a", encoding="utf-8") as write_coor:
    writer = csv.writer(write_coor, delimiter=",")
    writer.writerow(address_info)
```

10.2 Data

10.2.1 Housing Raw Data

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	housingid	estate	location	yearbuild	housingtyp	roompartiti	pricepersqr	area	orientation	floor	upgradelev	refdownpa	sellingpoin	ownerview	supportingf	expertview	estateprice	price	district	parkingspa	greening	mgmtfee
2	1165371088	罗马花园	长宁一古	1997年	公寓	2室2厅2卫	54755 元/	143平方米	南北	中层(共17层)		235万	1.房东重	房东精装	1、周边生	【小区户	54755yuar	783万	长宁	99个	40%	5.50元/m²
3	1165340736	中友嘉园	闵行一古	2005年	普通住宅	3室2厅2卫	67200 元/	125平方米	南	低层(共12层)		252万	卖点：1、	房子是一	手买进，房	【小区户	67200yuar	840万	闵行	1188个	45%	1.45元/m²
4	1165322219	保利叶上	宝山一顾	2010年	普通住宅	3室2厅1卫	47191 元/	89平方米	南北	高层(共25层)		126万	高品质小	产权2人	，幼儿园：	【小区户	47191yuar	420万	宝山	暂无	40%	1.90元/m²
5	1165297272	新梅花苑	闵行一古	1996年	普通住宅	2室2厅1卫	45714 元/	70平方米	南	中层(共6层)		96万	新梅花苑	房东诚心	小区是199	【小区户	45714yuar	320万	闵行	暂无	30%	0.60元/m²
6	1165261673	翠竹苑	浦东一三	1997年	普通住宅	2室2厅1卫	55555 元/	63平方米	南北	低层(共6层)		105万	《翠竹苑	业主置换	①卖点一	【轨道交	55555yuar	350万	浦东	568个	35%	0.45元/m²
7	1165083478	合阳小区	普陀一甘	1995年	普通住宅	2室1厅1卫	56338 元/	71平方米	南	高层(共24层)		120万	1.小区：	业主情况	1.交通便	【小区户	56338yuar	400万	普陀	暂无	暂无	0.80元/m²
8	1164894426	保集欧都	宝山一杨	2011年	普通住宅	2室2厅1卫	31372 元/	102平方米	南北	中层(共12层)		96万	一、房型	房东因置	周边学校	【小区户	31372yuar	320万	宝山	602个	40%	1.30元/m²
9	1164788201	星颂家园	浦东一曹	2013年	普通住宅	2室2厅1卫	39750 元/	80平方米	南北	中层(共18层)		95.4万	此房一手	房东在市	交通；*可	【小区户	39750yuar	318万	浦东	500个	45%	1.11元/m²
10	1164753266	一品漫城	闵行一浦	2014年	公寓	1室1厅1卫	60000 元/	50平方米	南	高层(共15层)		90万	1、8号线	业主置换	1、交通：	【小区户	60000yuar	300万	闵行	3323个	36%	1.00元/m²
11	1164511238	果园小区	奉贤一西	1900年	公寓	2室1厅1卫	32170 元/	58.8平方米	南	低层(共6层)		57万	果园小区	业主因置	小区周边	【小区户	32170yuar	189万	奉贤	80个	35%	0.36元/m²
12	1164468690	华府穆园	普陀一武	2013年	普通住宅	1室2厅1卫	90243 元/	82平方米	南	中层(共18层)		222万	产证面积	此房业主	地铁邻地铁	【小区户	90243yuar	740万	普陀	500个	38%	4.50元/m²
13	1164393462	聚龙家园	浦东一花	2002年	普通住宅	2室1厅1卫	54666 元/	75平方米	南	高层(共6层)		123万	1.户型：	此房为业	【教育配	【小区户	54666yuar	410万	浦东	150个	30%	1.20元/m²
14	1164393462	聚龙家园	浦东一花	2002年	普通住宅	2室1厅1卫	54666 元/	75平方米	南	高层(共6层)		123万	1.户型：	此房为业	【教育配	【小区户	54666yuar	410万	浦东	150个	30%	1.20元/m²
15	1164269822	曹杨汽车	普陀一武	2000年	普通住宅	2室2厅1卫	44818 元/	110平方米	南	高层(共20层)		147.90万	结构:总高	业主置换	【交通】	【小区户	44818yuar	493万	普陀	50个	26%	0.80元/m²
16	1164229205	金汇丽舍	闵行一龙	2002年	普通住宅	2室2厅1卫	53286 元/	100.4平方	南北	高层(共13层)		160.50万	价格优势	业主是我	交通：近	【小区户	53286yuar	535万	闵行	350个	40%	1.44元/m²
17	1163513080	高乐小区	松江一松	1998年	普通住宅	2室2厅1卫	32207 元/	77平方米	南北	低层(共6层)		74.40万	1、户型	业主诚意	1.配套齐全	【小区户	32207yuar	248万	松江	暂无	暂无	0.95元/m²
18	1163397400	前响路198	浦东一合	1998年	公寓	2室2厅1卫	31944 元/	72平方米	南北	共5层		69万	1) 挂牌价	*方式：正	合庆配套	【小区户	31944yuar	230万	浦东	200个	38%	0.50元/m²
19	1163133986	梅园五街	浦东一源	1993年	普通住宅	1室1厅1卫	124358 元	39平方米	南	中层(共6层)		146万	目前梅园	房东在我	1，学校=	【小区户	124358yuar	485万	浦东	暂无	暂无	0.80元/m²
20	1163049400	大华锦绣	浦东一北	2008年	普通住宅	3室2厅2卫	73913 元/	92平方米	南北	高层(共6层)		204万	此房产证	房子目前	轨道：地	【小区户	73913yuar	680万	浦东	800个	45%	0.80元/m²
21	1163015625	象屿上海	松江一莘	2017年	其它	3室1厅1卫	41397 元/	93平方米	南	中层(共9层)		115.5万	象屿上海	象屿上海	象屿上海	【小区户	41397yuar	385万	松江	1085个	35%	2.80元/m²
22	1162993701	碧绿湖花	普陀一曹	2010年	普通住宅	2室1厅1卫	42000 元/	120平方米	南	中层(共10层)		26万	*成立至今	业主工作	1.商业配套	【小区户	42000yuar	84万	普陀	120个	41%	0.80元/m²
23	1162772595	锦兰苑	杨浦一黄	2003年	普通住宅	2室2厅1卫	87640 元/	89平方米	南北	中层(共6层)		234.00万	房屋卖点	8号线，黄	兴公园，	【小区户	87640yuar	780万	杨浦	250个	40%	0.80元/m²
24	1165371088	罗马花园	长宁一古	1997年	公寓	2室2厅2卫	54755 元/	143平方米	南北	中层(共17层)		235万	1.房东重	房东精装	1、周边生	【小区户	54755yuar	783万	长宁	99个	40%	5.50元/m²

10.2.2 Location Data of Real Estates

	A	B	C	D	E	F
1	estate	district	sub_distric	address	lat	lng
2	新寺中街[illegible]	奉贤	柘林	新寺中街[illegible]	30.86992	121.4663
3	锦汇华庭	金山	朱泾	众安街10[illegible]	30.89581	121.1628
4	和瑞雅苑	杨浦	杨浦大桥	锦州湾路2	31.2727	121.544
5	秋潭苑	松江	佘山	江秋路19[illegible]	31.09318	121.1727
6	康乐新村	奉贤	柘林	康乐新村	30.27664	120.102
7	杭州路90[illegible]	杨浦	杨浦大桥	杭州路90[illegible]	30.17995	120.2611
8	秀洲阁	金山	朱泾	秀州街45[illegible]	30.87255	120.7139
9	淀滨宝岛[illegible]	青浦	金泽	宝银路17[illegible]	31.11272	120.9238
10	金溪三村	青浦	金泽	金溪三村	31.03601	120.924
11	金溪五村	青浦	金泽	培育路12[illegible]	31.03505	120.9246
12	楠林水岸	青浦	金泽	沪青平公[illegible]	31.06728	120.9251
13	海图公寓	青浦	金泽	宝银路13[illegible]	31.11348	120.9254
14	丰泽湾花[illegible]	青浦	金泽	锦商公路4	31.12144	120.9323
15	商榻老街[illegible]	青浦	金泽	商榻北路2	31.12737	120.9341
16	淀滨新村	青浦	金泽	岑中路96[illegible]	31.06509	120.9672
17	淀山湖别[illegible]	青浦	金泽	金商公路1	31.10783	120.9762
18	国际华城	青浦	金泽	沪青平公[illegible]	31.08554	120.9799
19	幸福新村	金山	朱泾	亭枫公路1	30.89115	121.0166
20	欧风小区	青浦	金泽	欧欧风路1	30.97346	121.0249
21	湾塘新村	青浦	金泽	湾塘新村1	31.00701	121.0396
22	练塘小区	青浦	金泽	下塘街20[illegible]	31.00783	121.0405
23	练北小区	青浦	金泽	练新路124	31.00783	121.0405
24	东风街小[illegible]	青浦	金泽	东风街2_1	31.01217	121.0431

10.2.3 Data of Facilities Surrounding Real Estates

	A	B	C	D	E	F	G	H	I
1	estate	type	sub_type	lat	lng	faci_id	faci_name		
2	新寺中街[illegible]	transportati	bus_station	30.86894	121.4578	28e43841e	新寺加油站		
3	新寺中街[illegible]	transportati	bus_station	30.86428	121.4603	481fc7d43	寺中路新寺中街		
4	新寺中街[illegible]	transportati	bus_station	30.86499	121.4585	f627c4670	新寺		
5	新寺中街[illegible]	health_care	hospital	30.87217	121.4608	219ff022d	新寺镇卫生院		
6	新寺中街[illegible]	food	restaurant	30.86471	121.4594	3c9738f72	川海家常菜		
7	新寺中街[illegible]	food	restaurant	30.86469	121.4592	38d1436da	新寺大厨房		
8	新寺中街[illegible]	food	restaurant	30.86493	121.461	7eb06c3f5	楼中楼酒家		
9	新寺中街[illegible]	food	restaurant	30.86491	121.4609	a45ec0aa6	龙辉家常菜		
10	新寺中街[illegible]	food	restaurant	30.865	121.4605	986da2551	三味快餐		
11	新寺中街[illegible]	food	restaurant	30.86489	121.4599	8010780e8	宇翔快餐饮食店		
12	新寺中街[illegible]	food	restaurant	30.86391	121.4607	e167c16b8	朋友酒家		
13	新寺中街[illegible]	food	restaurant	30.8635	121.4606	cf949ee6df	安徽土菜馆		
14	新寺中街[illegible]	food	restaurant	30.8618	121.4636	43a9cf05a	明新羊肉菜馆		
15	新寺中街[illegible]	food	restaurant	30.86154	121.4631	13f1fe237	瑶静酒家		
16	新寺中街[illegible]	shopping	bank	30.86465	121.4597	20935f542	上海农商银行		
17	新寺中街[illegible]	shopping	bank	30.86453	121.4598	3a591746f	上海农商银行新寺支行		
18	新寺中街[illegible]	shopping	bank	30.86526	121.4588	debe884a0	中国邮政储蓄银行		
19	新寺中街[illegible]	shopping	bank	30.86371	121.4603	a8a4bed39	中国农业银行		
20	新寺中街[illegible]	shopping	shopping_r	30.8644	121.4602	89e91e5f0	春鑫电器百货		
21	新寺中街[illegible]	shopping	shopping_r	30.86499	121.4593	84fc0e099	鸿新商场		
22	新寺中街[illegible]	school	school	30.86529	121.4619	441407735	奉贤区新寺学校		
23	新寺中街[illegible]	school	school	30.86522	121.4623	5ffba77853	新寺中学		

10.2.4 Air Quality Station Data

	A	B	C	D	E	F	G	H	I	J	K	L
1	station_ID	0	1	2	3	4	5	6	7	8	9	10
2		jinganjianc	luwanshizh	putuojianc	xuhuishang	pudongjian	hongkoulia	yangpu	pudongzha	pudongchu	qingpudian	consulate
3		173	182	179	179	171	165	190	165	187	173	174
4		199	184	205	201	196	195	187	191	221	189	195
5		199	184	187	203	186	221	186	195	185	185	191
6		182	182	190	197	183	189	185	179	177	198	192
7		157	160	159	177	147	160	171	171	173	166	161
8		163	168	167	176	170	152	171	178	173	165	175
9		171	168	167	181	177	156	172	180	173	171	175
10		171	168	167	181	177	156	172	180	173	171	175
11		171	177	167	187	177	170	172	180	173	171	183
12		183	182	186	193	180	174	187	185	184	178	183
13		188	189	187	201	185	185	197	191	190	195	183
14		188	189	187	201	190	185	198	191	190	197	187
15		192	190	192	216	190	190	198	188	175	199	194
16		194	197	191	202	187	189	201	191	174	190	178
17		194	197	191	202	187	189	201	191	174	190	178
18		165	157	168	163	159	159	158	155	156	170	155
19		165	155	164	163	159	158	160	157	157	170	160
20		162	158	164	165	159	161	161	164	159	170	162
21		162	158	164	165	159	161	161	164	160	170	177
22		164	159	164	162	160	162	161	160	160	170	164
23		158	158	155	158	156	159	158	157	155	161	175
24		161	158	159	164	157	154	159	153	158	165	175