Project Report of Ride-Hailing Data Analysis and Visualization

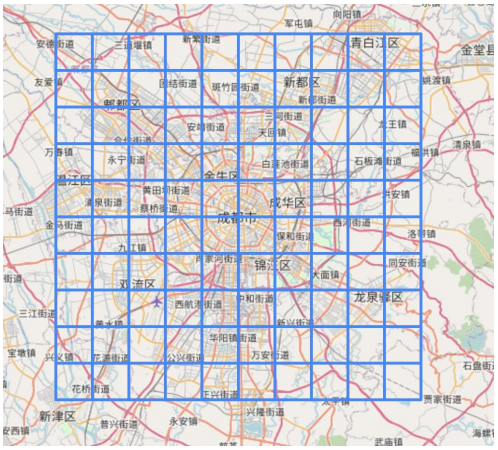
519030910273 张亦昕

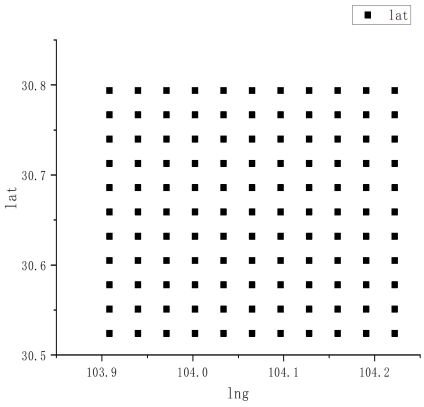
I. Opening

With the development of the technology of internet and computing, online ride-hailing has brought great convenience to our daily life. However, some of the online hailing systems are facing negative comments because of certain challenges such as the information gap between the supply and demand. Therefore, as the final project of CS241 Principles and Practice of Problem Solving, we students are offered with the order data of a period in Chengdu City, to play a role as a data analyst, providing help to the company by analyzing and visualizing the order-related data.

II. Introduction

Due to the complexity of various orders, in order to meet the requirements of refinement, the main urban areas of Chengdu is divided into 10*10 grids, as shown in the picture below. Based on the latitude and longitude of each lattice given, the grids can be roughly drawn by the Origin software. Each of the grid takes up 0.03136 longitude and 0.02698 latitude.





According to the task requirements, in the program that I have done, the following functions are implemented:

1. Load the data charts from the disk
2. Draw a line chart to show the shifting trend of the number of orders according to time, the range of which can be set discretionarily.

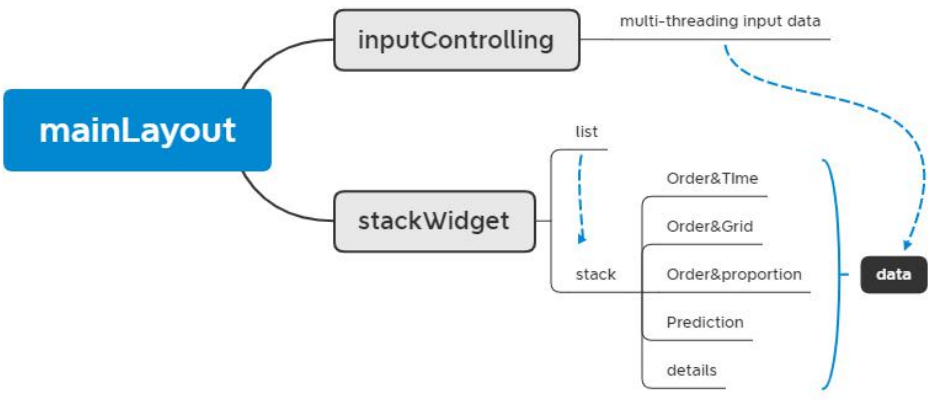
3. Draw a line chart to display the distribution of orders in Chengdu City, the range of which can also be adjusted.
4. Use a pie chart to display the proportion of orders distributing in different areas of Chengdu city.
5. Predict the possible destination based on the given data, estimate the time and fee according to the origin and destination given by the customer.
6. Show the time length of orders in certain days using a bar chart.

The ideas of using these charts and functions is that I want the data features to be visualized, where users can get useful information. Line charts are used to show the change of the number of orders by time. Users can know which day and when is an order peak to avoid getting stuck in the traffic congestion. Line charts of Orders & Grids can display which grid is the busiest, which can help the driver to select a less jammed route. The pie chart can display the proportion of orders in each grid. The Prediction part can help the passenger to estimate the time and cost of taking a ride. The time length distribution can help the company to get a overall view of orders in different days.

III. Implementation Details

All codes are written in C++ on the Qt Creator application.

The Graphic User Interface is mainly a class called *mainLayout*, which is inherited from *QWidget*. The class and layout relations are displayed below:



All the small items are inherited from *QWidget* Class.

Data Importing: During the data importing part, I created 15 new threads to read the data. I created a new class called *OrderInformation*, which stores the starting and ending time of each order, longitude and latitude, starting and ending grids and order fees. Each .csv file is imported into a QVector, and the pointers of all the 75 QVectors are combined into a QVector called *DataSet*. A *DataSet* is all the data we need. After a chart is totally read, a signal is emitted to the main thread, causing the progress bar to go forward, the counting variable will minus by 1. After all the files are loaded, the progress bar will disappear.

The First 3 Charts related to orders: the parameters can be set by the Spinboxes and LineEdits, and the “Plot” button can ask the program to display the chart according to the data. After the button is clicked, the program will start to traverse the DataSet, select out the qualified orders, calculate the number of each step and the sum of the fees to get the revenue. The data is then put into the chart, and interpolation is carried out to make the line looks smooth. In fact, the “Plot” button is asking the program to re-filter and display the newly generated data.

Prediction: the Origin and Destination Grid has to be input into the LineEdit box. The user can push the “Predict” button to ask the computer to display the result. The program will traverse the whole data, select out the orders whose origin is matched to the input and add the destination array element by 1. After the traverse is finished, the program will sort the copy of the array, the first-3 frequent destination will be displayed. According to the input Origin and Destination, during the traverse, qualified orders will be selected out. The program will calculate the average time and fee of all the qualified orders and display is. Therefore, circumstance that did not happen in the DataSet will not be displayed.

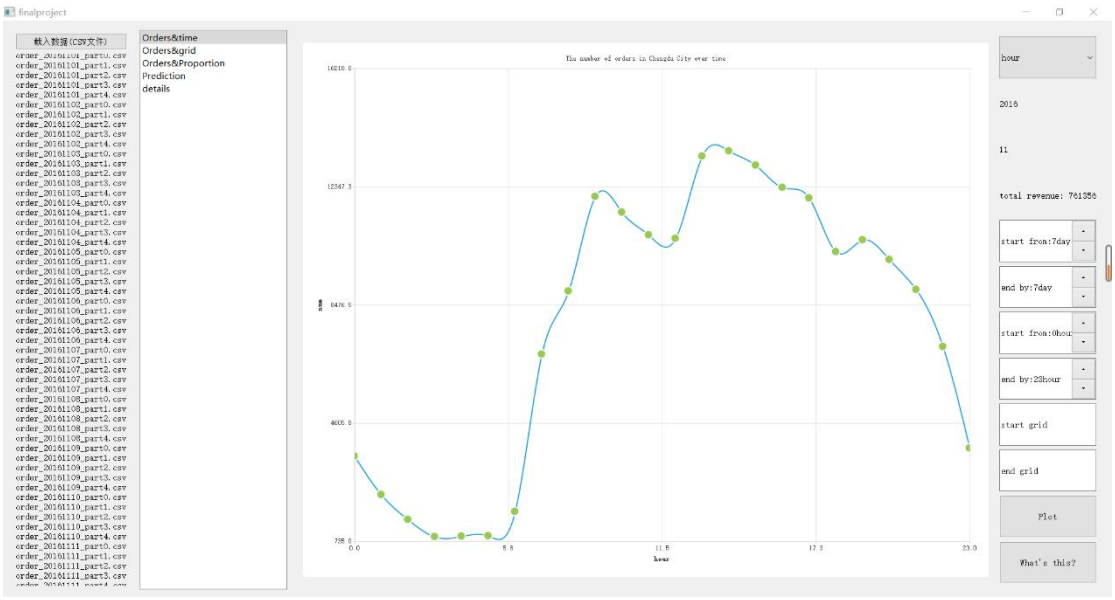
Travelling Details: Since the time is mainly distributed in 0 to 10000 seconds, the time step is 200 seconds. Orders in each day will be traversed, and sorted in to time ranges. There are 5 data sets in each chart, all 3 charts combine together to make up the situation of all 15 days. The bar chart includes the number of orders in each

time range.

IV. Results

Click the load button to load all 75 files, the loading process takes roughly 1 minute 40 seconds (tested on my own computer). The file names can be displayed in the left column. The first chart: set the mode to “day” the day range is 1 to 15, the chart can be displayed. Set the mode to “hour”, set the start day and end day to the same day, set the hour range to users’ interest, click the Plot button and the whole graph can be seen. As for the grid chart part, the time range can be selected, and user have to input the grid range I the two LineEdit boxes, click the Plot button, the chart can be viewed. The operation of pie chart part is the same as that of the Order&Grid part, the difference is when your mouse is hovered on one of the pie slices, it will show the corresponding grid number. The last part is the travelling time distribution part, user can get a view of the travelling time length of all the orders, whose time step is 200 seconds.

The graphical interface and chart example is listed below:



V. Discussions

The loading progress takes about 100 seconds, I tried to change the form of data storage from saving copies of the array to saving the pointer of it and did some other

optimization, but the time just reduced by about 40 seconds.

The order number trend in a day changes a lot as the time passes. There are few orders in the before dawn time (from 0 to 6 o'clock), there is a sharp increase in 7 o'clock, and the number of orders in daytime remains high. The number decrease a little bit in the noon, it will increase to the peak in 1 o'clock in the noon. It will smoothly go down in afternoon and night. I went to Chengdu for touring this summer, the time schedule is greatly different from that in my hometown Shandong. The people there sleep late at night and wake up much later than people in Shandong, as a result, I woke up in the morning a bit early but failed to find a opening breakfast store. The number of orders reach a maximal value in 9 o'clock in the morning, which is quite fit to my own experience in Chengdu.

Based on the pie chart of the program, the grid number 55, 56, 54, 45, 46, 65, 64 take up a great part of the total orders. Check the map and you will find that these grids are in the central part of Chengdu. No wonder these grids are extremely busy. What is interesting is that grid 84 covers a great ratio compared with the grids around it, which is far from the city center. I looked through the map, to find that the Chengdu Shuangliu International Airport is around that area, so the riddle is solved now.

VI. Thanks

Thanks to the lecturer Dr. Yuye Ling, who is an excellent model to all the students to students of Computer Science Department, Dr. Haiming Jin, who offers a rough view of algorithms and machine learning to us. Thanks to teaching assistant Jiahui Sun and Zhaoxing Yang, who spend a lot of time and energy on creating and reviewing our homework. Thanks to my classmate Jianxing Qin, who lead me to proficiently using the Qt creator and provide great assistance during the process of working on the final project, Zhuoren Liang, who also give me many inspirations and assistance during the study of the whole lesson.