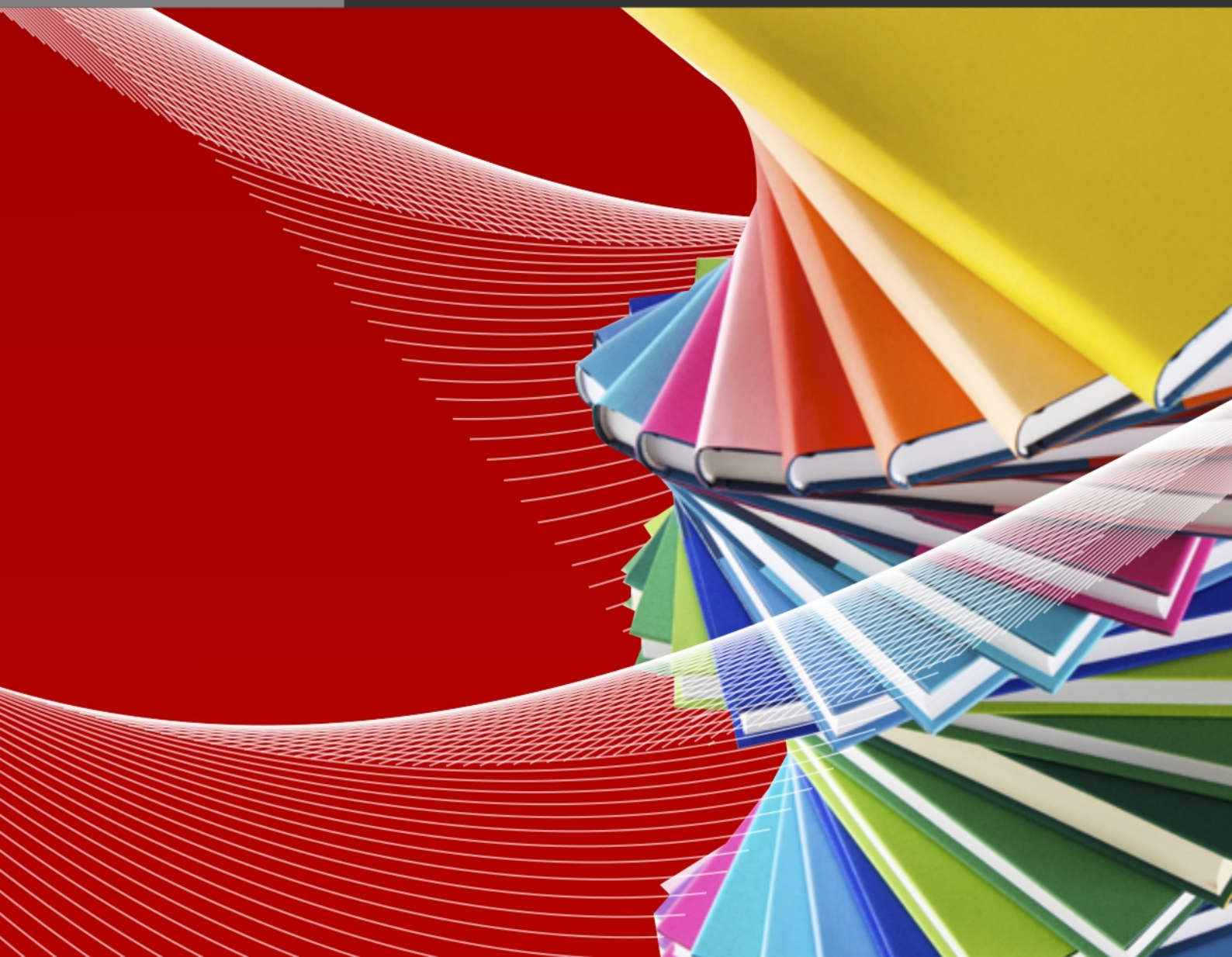
澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Outstanding Academic Papers by Students
學生優秀作品



University of Macau

Faculty of Science and Technology

澳門大學

UNIVERSIDADE DE MACAU

UNIVERSITY OF MACAU

Travel Route Recommendation System

By **Brian, Pun Chi Lap Student No: D-B1-2804-6**

Final Project Report submitted in partial fulfilment
of the requirements of the Degree of
Bachelor of Science in Computer Science

Project Supervisor

Dr. Leong Hou U

21 May 2015

DECLARATION

I sincerely declare that:

1. I and my teammates are the sole authors of this report,
2. All the information contained in this report is certain and correct to the best of my knowledge,
3. I declare that the thesis here submitted is original except for the source materials explicitly acknowledged and that this thesis or parts of this thesis have not been previously submitted for the same degree or for a different degree, and
4. I also acknowledge that I am aware of the Rules on Handling Student Academic Dishonesty and the Regulations of the Student Discipline of the University of Macau.

Signature : _____

Name : Brian, Pun Chi Lap

Student No. : D-B1-2804-6

Date : 21 May 2015

ACKNOWLEDGEMENTS

As a partial fulfilment of the requirement for the degree of Bachelor of Science, we are utmost gratitude to UM for providing the opportunity to carry out a project. We would like to thank Dr. Leong Hou U for providing us this topic for the final year project, and giving lots of opinions, suggestions and comments for this project.

This project first got started in the summer of 2014, as the topic for the final year project of the last year fresh graduates. The whole redesigning and reconstruction process is painful but we finally walked through all the difficulties under his help. Without his efforts this project definitely cannot reach the goals.

At last, I am thankful, from the bottom of my heart, to my group mate, who had been spending almost the whole year with me in this large project. On the discussion, designing, implementation, testing and refining stages, we had been working hard to walk through these stuffs. We argued a lot when facing difficulties and different opinions. With the encouragement to each other, we finally worked the problems out and be able to complete this project, and to share the achievement together.

ABSTRACT

The geotagged photos enable people to share their personal experiences as tourists at specific locations and time. Assuming that the collection of each user's geotagged photos is a sequence of visited locations, photo-sharing sites are important sources to collect the footprint of tourists. By carefully analysing their footprint movements, our objective is to extract representative and diverse travel routes.

In this paper, we propose a travel route recommendation method that makes use of (1) the travel experiences extracted from Flickr geo-tagged photos and (2) the landmark data from Open Street Map. 639,680 geotagged photos are included in Flickr and 9,227 locations are included in OpenStreetMap.

We first propose a mapping algorithm to map a geo-tagged photo is to a landmark based on their longitude and latitude. This mapping enables us to extract important landmarks from our photo collection. Then we use these data to estimate the relationship of the landmarks based on association rule. Lastly, we apply a genetic algorithm to generate tourist routes based on different criterion to satisfy user needs.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

1.1 Motivation

Travelling is one of the most critical things of our life. We can broaden our views and explore our world through travel. Furthermore, many people find it as an efficient way to relief stress and purify soul. Therefore, a well-planned travel is very important.

Nowadays, many people prefer independent travel rather than group tour because the schedule of independent travel is more flexible and personalized that you can choose your dates of travel as well as decide your own travel itinerary. Although there are lots of benefits on independent travel, the preparation is really difficult and annoying since there are some factors need to be concerned such as budget, travel route, transportation and accommodation. In real cases, most travellers enjoy the trip but not on the planning time because they find that planning itinerary is really hard and time-wasting and many of them get trouble on it. Most travellers want to visit all famous places and attractions in a limited time but they do not know the way to begin planning. The traditional method to solve the planning problem is buying travelling books, searching travelling blogs on the internet or asking friends' recommendation. However, those methods are not efficient and objective.

We are in the age of advanced information technology today, computer is commonly used and browsing internet is the thing which we almost do every day. Because of the popularity of using computer, we want to establish a system which can make our daily life more convenience. As the above mentioned, planning travel route is inevitable in independent travel and it is not an easy work. Therefore, we want to build a tool in order to help traveller to plan their travel route more easily. Since taking photos to record travel is the thing that most people would do, we use it as a source to establish the Travel Route Recommendation System. We want to summarize the experiences of travellers, analysed the locations where most people would go and when would they go to provide users a best travel route. We hope that we can create a scientific system to decrease the trouble which travellers may meet and saving their time in order to make them fully enjoy the trip and have a nice travel.

Figure 1: Generous concept of Travel Route Recommendation

Concept: User select their interest location and we provide the travel route to user

1.2 Challenge

Existing route recommendation systems have three main weaknesses. They usually recommend the route using some official website information so that they cannot cover some special point that the official website does not provide or the point that only the resident in that place known. They do not consider the time parameter in the point of interest. For instance, suppose there a supermarket that the opening time is nine o'clock and the closing time twenty-one o'clock. There should be recommending the user to that supermarket within the opening time. However, current existing recommendation system always ignores this point. In addition, the period of the traveller staying at the point of interest should also be a distribution not just the fixed value. Existing route recommended systems usually consider the distance between two points of interest. Actually it is not enough.

To address these problems, we develop a new route recommendation system. Aim to providing a route that can depend on tourists experience in the past instead of some official data on the internet. However, it is not an easy job to collect tourist route information. It is because tourists usually planning their trip with their own computer or just writing a blog. First, it is hard to find the source since we are hard to define which one should be included. Second, it is impossible to parse the data that we fetched on the internet due to the format is not the same. Third, each article provided different information so that it is difficult to do the analysis. As a result, we cannot provide useful information.

With the growth of social network site and global positioning system widely use, we find out one of possible solution in here. Nowadays, many people like posting photos on social network site. Thus, we are able to collect the geotagged photos on the social network site, then according to the user who posted the photos and the date, who posted, we can easily to figure out that user's route in the past. Furthermore, this route has detail information about the time that the user visited and also the coordinate.

Despite of we come up with the solution of the source data. The task to fetch source data is much more difficult than imagine. There are some challenges for us to collect the users' data. It is because the users always setting the limitation that who can see the photos and only friend can view is the common setting nowadays. The social network site will have limitation also. In addition of this, geotagged photo is much less than the photo without geotagged relatively.

In analyse part, we choose association rule as our main analyse method. It is the most common and popular method that widely use to discovering the interesting relations between variable in large databases. As a result, we occur a problem in our analyse module. The data is really discrete due to the data is fetched on the social network site so that the point of interest's support is really low. Therefore, we need to find out a method that can increase the support, in the meanwhile make sure the data is used completely so that we can create a trusty weight table.

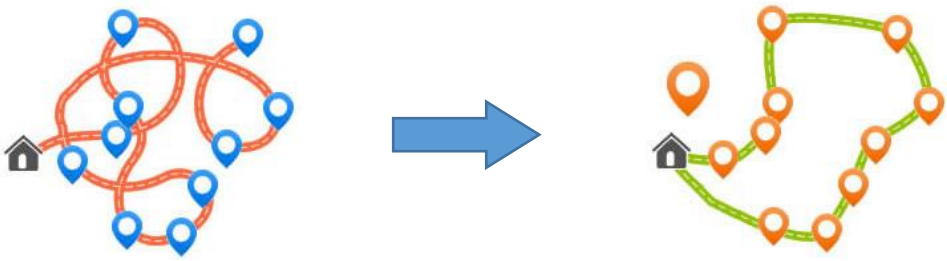


Figure 2: Generous concept of Travel Route Recommendation

1.3 System overview

With the popularity of smart phones and digital cameras that embed GPS technology, numerous geotagged photos were produced in these recent years. As a result, huge amounts of travellers recorded their journeys with photos and all photos are reflecting their travel behaviours and experiences. Therefore, there is a lot of useful travellers' knowledge accumulated on the internet. Those data are important and valuable for find the travellers' interest place and their travellers' route which can help other travellers as reference experiences or government to planning the public transportation.

The task of Travel route recommendation system is going to provide a custom route in city base on other travellers' record. And our work is to mine out the travel route in a city from huge amount of photos form Flickr. We have to find out the most useful data from the photo. When the huge amount photos goes together and we have to discover the relation between photos or users. It may be the user interest, sequence of photos or spending time. First of all, we choose Taipei as our testing city because which language is using Chinese and we have some travel experiences in there. All photos data is capture from Flickr legally.

Every single photo's data basically includes some information about the photo taken time, taken location's latitude and longitude. But all the locations of photos are dispersed and difficult to identify the taken place. We use Open Street Map's point of interest (POI) database and filter out other POI except viewpoint. After POI is ready, we used MPR (Minimum Bounding Rectangle) method to group the photos and it can easier to identify this area's photos is for which place. Also using the POI can classify the photos location to a place. Each place has a group of photos which are identified. Therefore, the traveller can be distinguished in three types (shopping, visit, food) according to the POI types of place. And we can measure the most popular visit time in a place by those photos taken time in one day. Similarly, the staying time in that place also can be analysed by traveller first photo and last photo taken in that place. The advanced way is to combine a traveller's photos in a group, and then we process many groups' photos to figure out the location sequences of all users.

Furthermore, all the traveller routes, place type, location, visit time and distances will process by the algorithm which included three method Genetic Algorithm, distance between the place and visit time. It calculate the weight for each place between other which is to evaluate the most reasonable route by those method, system will assemble all places rate, visiting time, traveling type and users to consider things like starting traveling times, visiting place types and starting places etc.

Finally, the Travel route recommendation system which will provide a recommend travel routes for user base-on many previous experiences from other travellers.

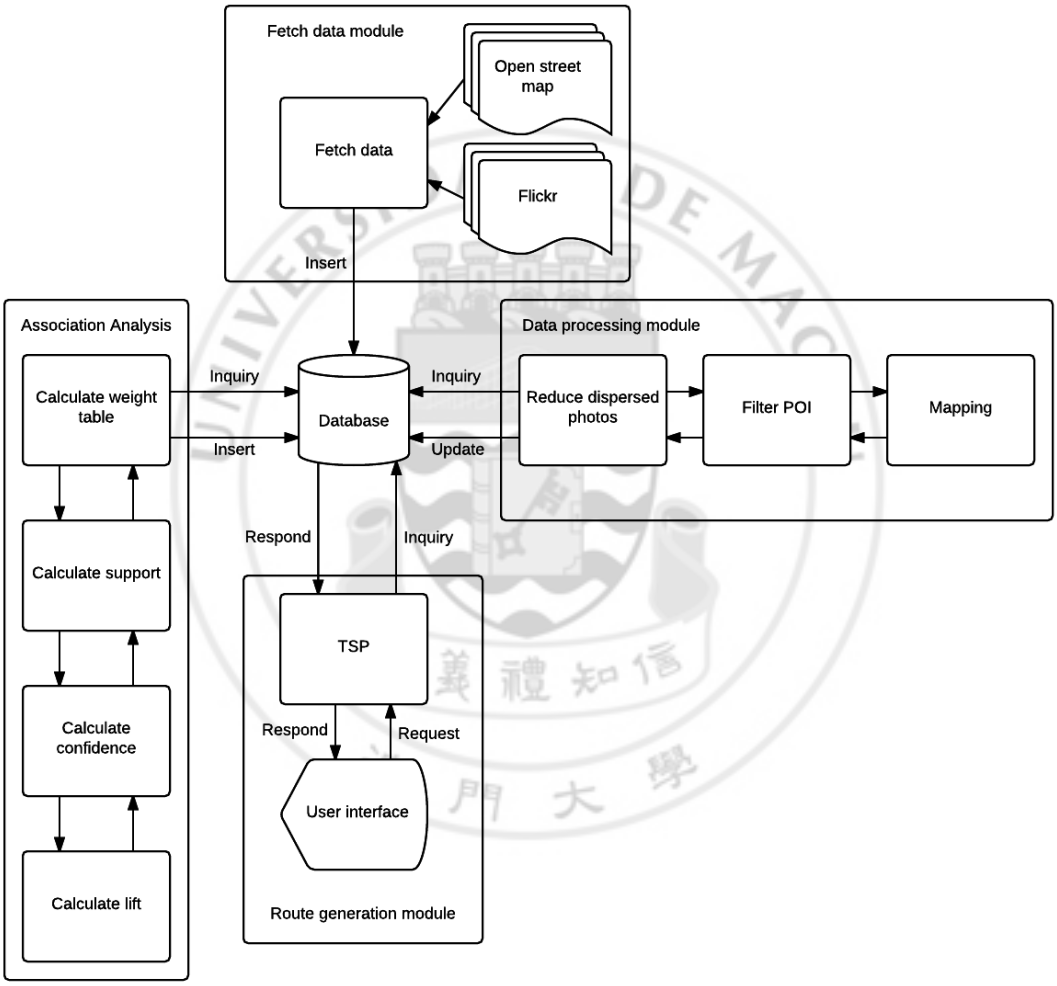


Figure 3: System workflow

CHAPTER 2. Literature survey and related work

Our concept of the system is user select their interest place and system will generate a route ordered by other place food print. We have some research for this topic that most travel/Journey planning website or apps are separate into two categories. One is provide two points transportation in a city (See Figure 4 and Figure 5). This journey planning system only provide you to select two point in a city and they will suggest three route to go there but the traveller most force on the viewpoint but most travel have no idea where is the popular place so that the information is not comprehensive.

Reference Website

- http://transportnsw.info/
- http://www.theaa.com/
- http://reittiopas.fi/

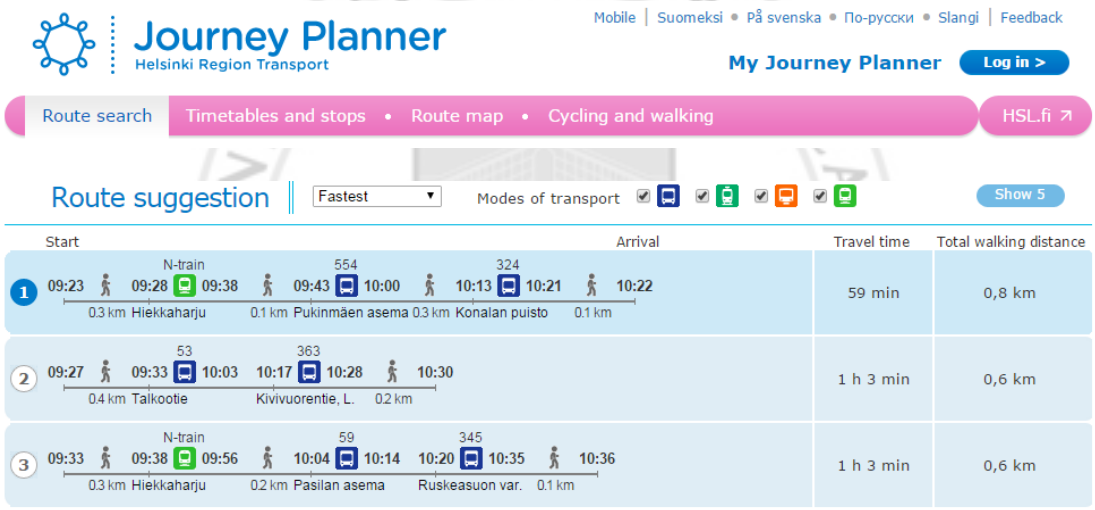


Figure 4: http://reittiopas.fi/ Sample test.

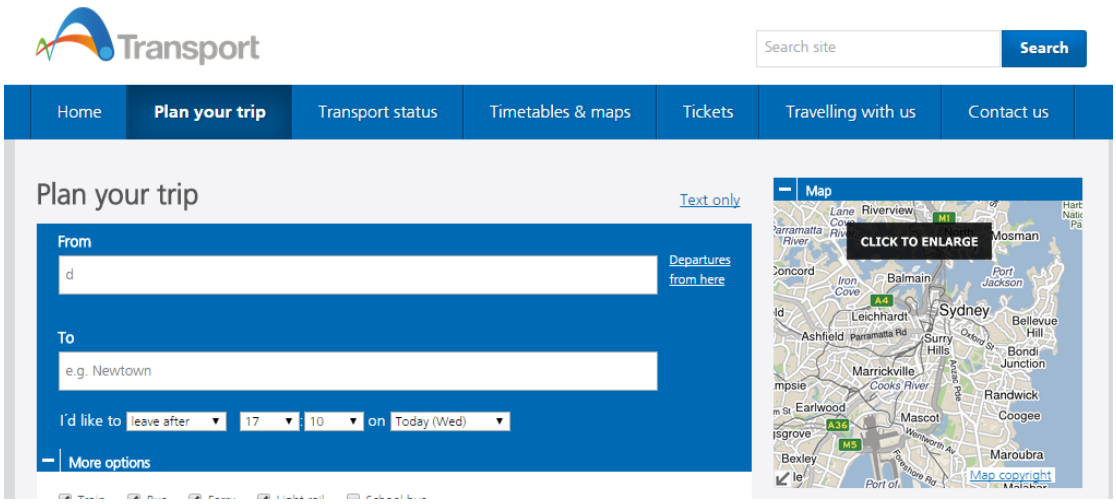


Figure 5: http://transportnsw.info/ Screen Caputer

Another type website is provide other users travel planning. Both of them are inadequate, in first category that. They just allow user to select some popular place in the city base on other user's pervious travel planning. We also find that some website have provided travel route optimization but only focus on the distance and not independent location opening and closing time or the transportation. For example in qyer.com (Figure 6) which is a screen capture in route optimization but the result is not acceptable because they arranged our route such as museum to the last of our route. Actually, if I follow their route and when I arrive the museum will be in evening. In my initial guess their system is only using one method to calculate the distance of each location and find out a shortest way for the suggestion. But this is very not enough for a route recommendation because the transportation and place visit time are also importance for user. If you suggest a route is not reasonable that this result will let user confused. So that our system are necessary and useful for those user to have a better planning.

Reference Website

- http://www.vztaiwan.com/
- http://plan.qyer.com/

Figure 6: http://qyer.com/ Route optimization

For our route generation module, we use the generic algorithm to solve problem. About generic algorithm, we reference article "Genetic Algorithms: Concepts and Applications"[1], this article has detail explanation of generic algorithm, it explains how generic algorithm work and also the limitation of it so that I get the idea of how to build our route generation module.

For the purpose of the efficiency in genetic algorithms, Korea's article "An efficient genetic algorithm for the traveling salesman problem with precedence constraints"[2] provide us a well solution to deal with it so that we can reduce the running time on the calculate of genetic algorithms.

For the number of generation in generic algorithm, we read the article "An efficient genetic algorithm for the traveling salesman problem with precedence constraints"[3], this article use some well-known instance of generic algorithm to test the performance deteriorates with populations significantly smaller or larger than the estimate recommends. Finally this article figure out that the max number of generation should be *2log2(N).*

For our data analyse module, we use the association rule to reach the goal. Having read the article "Association Analysis: Basic Concepts and Algorithm"[4], we get the basic idea of the association rule.

In order to further understand association rule, we read the article "Support vs Confidence in Association Rule Algorithms"[5], therefore, we understand the pros and cons of the support and confidence so that we can apply association rule in our system. Also we read the article "Standardising the Lift of an Association Rule"[6], this article introduction the idea of lift. As a result, we can have better design.

CHAPTER 3. SYSTEM OVERVIEW

3.1 Environment

- Program: PHP + Java+ JQuery
- Server: Windows
- Database: MySQL

3.2 Data Source

We have two main data source Flickr and OpenStreetMap and designed two data collect module to both source. Flickr is more easy because which provide an API directly to access the data. But also have some limitation. And OpenStreetMap provided a very large data package and we have to build a program to read all useful data and copy out.

3.3 Data Pre-processor

- Data Processor

  After we got all data from the source which is a basic data. The system need more useful data and we designed some module.

  - MPR (Minimum Bounding Rectangle) – use minimum rectangle to group photos into group as different place.
  - Find the popular visit time
  - Find the visiting time in a place

- Location Weight Processor
  - TSP
  - Distance
  - Time

3.4 Server

- MySQL
  - Store the processed data, POI data and the places rate.
- PHP

- Sending the user input value and receiver the return data to Java using socket commutation.
- Process the return data and get data from database.
- Access OpenStreetMap API to get the drive route
- Map generation (Use the return result and OSM drive route result then draw the route in image)

- Java
  - Receiver the user input and calculate the places rate using those method to return a reasonable route to PHP.

3.5 Interface

- Using web interface
  - Left side is the Taipei city map and user can select the location who want to visit and the system will return the result to user on the map as a drive route.
  - Right side is the control panel for user.
    - Numble for place (how many place user want to go)
    - Start time (starting time of visit day)
    - Type for place (which type of place you prefer first)
    - Method (distance, time, weight)
    - Numble for Generation (About the algorithm running times)
    - Fix Start Point
    - Looping (User will go back to start point or end on the last place)

Figure 7 : Inserting the Cross Reference of a Figure.

3.6 Overview

This system is provided an interface for users to select the point of interest that they want to visit, and then generate a route in map to the user with point of interest information. In addition, we also provide several features to our user. For example, Supplementary function is a convenience function. It allow user just select the type of point of interest that they want to go and how many places they want to go, and then we also can provide a recommended route to users.

Basically our system workflow is use website to collect user request, once submit button was clicked, the request will sent to our backend program through socket. The backend program will do the calculation according to the request. After that the backend program will sent back the result to the frontend program, final the frontend program will update the website and display the result.

In addition, our frontend program, the data collect module and data processing module are using Php and JQuery, on the other hand, the backend program, such as analyse module and route generation module are using Java.

CHAPTER 4. APPLICATION SYSTEM SOFTWARE DESIGN

4.1 System database design

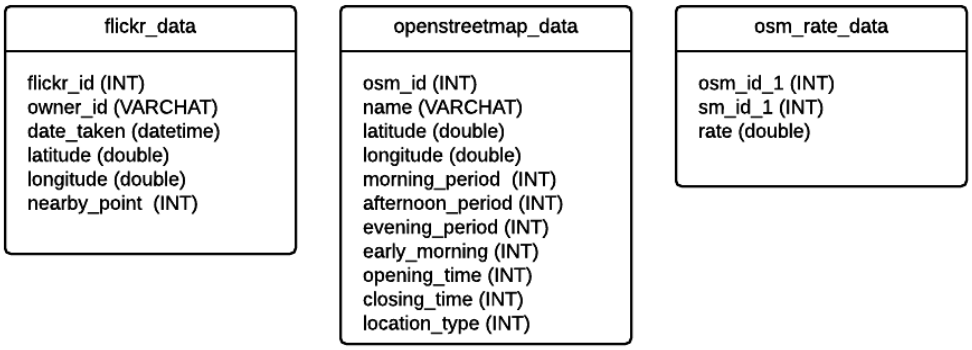


Figure 8: Database diagram

4.2 Data collect module design and target

Travel route recommendation system used two data source for photos data processing, route analyse and viewpoint located. In this project, we focus on Taipei city which is because of the language is Chinese and we have experience to visit this city then easier to evaluate the result of this system. Therefore, our target data is all Taipei photos in Flickr in past 4 years and OpenStreetMap POI (Point Of Interest) in Taipei city.

4.3 Data processing module design

After integrated all data on target. We have to process and optimizing the photos data to centralize for matching the POI location to the photos.

A. Reduce Dispersed Photos (4.3.1)
B. POIs Filtering
C. Mapping photos info POI
D. Popular visit time in a place (4.3.2)

4.3.1 Reduce noise in photo data

The distribution of photo is dispersed. In order to deal with this problem, we have following steps. First, reduce dispersed photo. Second, filter point of interest. At last, map photo with point of interest

Figure 9: Original distribution of photo

Figure 10: Distribution photo after processing

4.3.2 Time distribution and smoothing

The photo taken time is useful information also. We have to calculate 4 years photo data for each location and order by the taken time to find out the peak visiting hours.

Figure 11: Calculate 4 years photo data in SNM and order by the taken time

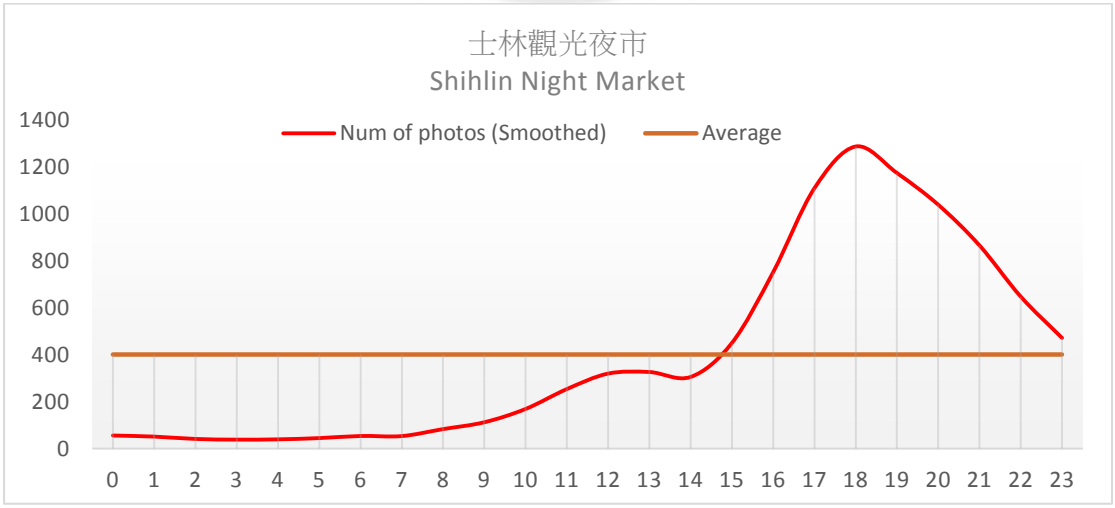


Figure 12: Calculate 4 years photo data in SNM and order by the taken time (With smoothing)

CHAPTER 5. Application System Methodology

5.1 Data collecting module

Flickr and OpenStreetMap is our main data source for photos data and POI (Point Of Interest) in Taipei city. We designed a data collection module for appropriate the official API limit and store all data into database for the following process.

5.1.1 Flickr data source

Flickr is a popular image hosting website which hosted over 5 billion photos and provide an API (*Application Programming Interface*) for non-commercial use. Developers allow to access all public photo and photo related data such as *user_id*, *taken_date*, *image_id*, *geo_tags*. But there are some limitation for free access that is each API access only responses 500 latest record and not allow to access over 5000 times per hour. So that we have to design a data mining module acceptable for Flickr API. The data start point is from 2014-09-01 to 2010-01-01. And we defined the end point is 2010 because of the photos data not appropriate and most photos lost geography tag.

Used API Methods

1. flickr.photos.search – to find out all the photos taken in Taipei and included the *user_id*, *image_id* but not included the *geo_tag* for photos. And because of the responses limit and we cannot access and record all data same time. It search the photo 1704 days hourly in target and total access the API 40896 times. Finally we got 639,680 photos data from Flickr.

```
<?pseudocode
function connect_flickr_api($val1,$val2){
        $data = file_get_contents (post_val_to_flickr_url);
        Return $data;
}
connect_mysql_db();
$start_date = "2014-09-01";
while(date($processing_date) <= date("2010-01-01")){
        for ($i = 1, $i<=21,$i++){
                connect_flickr_api();
        }
        write_to_db;
}
?>
```

2. flickr.photos.getInfo – for the advanced photo data such as *taken_datetime*, *latitude*, *longitude*, *location_id*. Scan every photos for detail data and this process used over a month to collect it.

5.1.2 OpenStreetMap data source

OpenStreetMap is a free editable map of the world and open data for public. Offical will pubish the data package every month but which is a very large data over 20GB. There are some difficulty to disassemble the data package because of the format is taiwan.osm.pbf. We found a converter to converter the data to csv format for follow proccess. Finally we got 14,431 POIs from OpenStreetMap in Taipei. This POI package include all POI type and we have to filter out useless POI such as road, hospital, gas station etc.

```
1.   read_file
2.
3.     while( read_file !== FALSE){
4.             change data format to CSV
5.             change CSV into array;
6.            if ($longitude >= 121.669281005 && $longitude<=121.357322692)
7.             if ($latitude>= 25.1402015686 && $ latitude<= 24.9430255889)
8.             write into db;
9.     }
10.  ?>
```

5.1.3 Distance and traffic route method

In the system, we also provide the distance, traffic time and traffic route between two locations which is using OpenStreetMap API to access the two location traffic data. We choosing which is using driving traffic data because OpenStreetMap public transportation data in Taipei is not enough and some location very close but they suggest to take bus and using double time to go the next location. We can try to using Google Maps API to access the traffic data but which provide a very limit API, if using over a number and which will block this. If we want a better result of the traffic and distance data, we have to pay for this service.

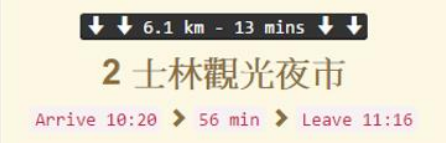


Figure 13: Display distance and traffic time

5.2 Data processing module design

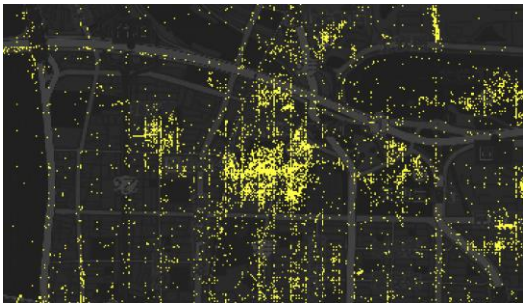
For all photo data from Flickr are dispersed (see Figure 14). So we have to build some module to group an areas photos in a place.

Figure 14: Plot all photos' location into maps

(Each yellow pixel equal one photo)

5.2.1 Reduce Dispersed Photos

Reduce dispersed photos in the map. Firstly we have to compare which photos is not useful for generating the route or get information. (see Figure 15 and Figure 16) In Figure 15 the photo is sparse then figure 16. In figure 16 can find a group of photo is much more close together and which mean many people taken photo this area so it may be a viewpoint in this city. So that we have to enhance and highlights the group photos and deduce the mess point.



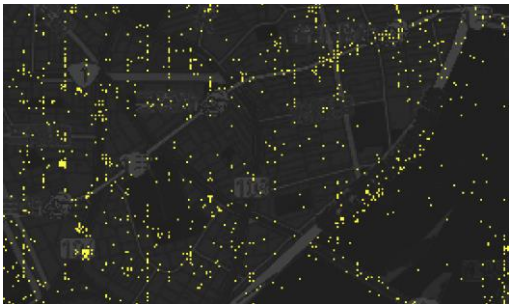


Figure 16: Dispersed photos (Sample B)

Algorithm

A. Suppose all photos have 100 more neighbours photo

B. Let D as distance <= 0.5 km

C. Let T as target photo 100 (100 photos in 0.5km radius)

D. Calculated distance (D) between other points using below formula.

Distance = D
Current Point = (Lan_1, Lon_1)
Selected Point = (Lan_2, Lon_2)

$$D = 3956 \times 2 \times \sin^{-1}\sqrt{\sin^2\left(\frac{(lat_1 - |lat_2|) \times \pi}{180 \times 2}\right) + \cos\left(\frac{lat_1 \times \pi}{180}\right) \times \cos\left(\frac{|lat_2| \times \pi}{180}\right) \times \sin^2\left(\frac{(lon_1 - |lon_2|) \times \pi}{180 \times 2}\right)}$$

E. Count number of points (N) in (D) radius for all point (n) (See Figure 17 and 18)

F. If $N(n_1) <= T$, then mark this point is not useful

G. If $N(n_1) <= T$, then mark this point is useful and using for the follow process

The follow program will find every point and check in 0.5km whether over 100 photos have in there. (See Figure 17 and 18)

```
1.  <?pseudocode

2.  Select all point in db;
3.  While (all_point) {
4.
5.  SELECT *, 3956 * 2 * ASIN(SQRT( POWER(SIN((latitude] -
     abs(latitude)) * pi()/180 / 2), 2) + COS(latitude] * pi()/180 ) * CO
    S(abs(latitude) * pi()/180) * POWER(SIN((longitude] -
     longitude) * pi()/180 / 2), 2) )) as distance
6.  FROM flickr_data dest
7.  having distance < 0.5
8.  ORDER BY distance limit 100
9.  "
10.
11. $result = mysql($sql);
12. If count $result >= 100;

13. Mark this photos have over 100 relate photos
14. }
15. ?>
```

Figure 17: Process dispersed photos (A)

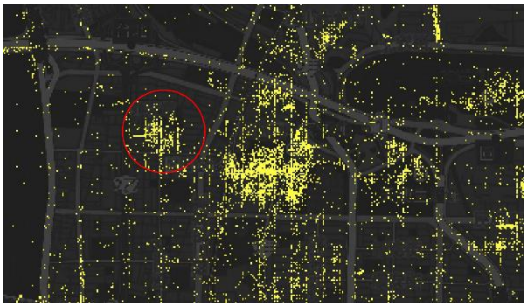


Figure 18: Process dispersed photos (B)

Processed 1 times point reduce and the result as figure 19 and figure 20. They are distinctly reduce dispersed points. More points are group together.

Figure 19: Processed dispersed photos (A)

Figure 20: Processed dispersed photos (B)

5.2.2 POIs Filtering

OpenStreetMap's POI (point of interest) is a feature on a map that occupies a particular point. This point classified in different type.

Some example of types of POI:

- Churches, schools, town halls, distinctive buildings market
- Post offices, shops, post-boxes, telephone boxes, mall
- Tourist attractions, Museum

Most POI types is useless for a travel place like post offices and telephone boxes. Now we have over 644,431 POI data in Taipei. Firstly following the POI filtering rate table (see Table 1) to filter out *Rate=0* irrelevant POIs (`LIBRARY, POLICE, COLLEGE, FASTFOOD, ROAD, TERMINAL, BUSSTOP etc…`).

Table 1: POI filtering rate table

Rate = 0	Rate = 1	Rate = 2
ACCOMMO_ALPINEHUT,1 ACCOMMO_CAMPING,2 ACCOMMO_CARAVAN,3 ACCOMMO_CHALET,4 ACCOMMO_HOSTEL,5 ACCOMMO_HOTEL,6 ACCOMMO_MOTEL,7 AMENITY_COURT,8 AMENITY_FIRESTATION,9 AMENITY_LIBRARY,11 AMENITY_PLAYGROUND,12 AMENITY_POLICE,13 EDUCATION_COLLEGE,19 EDUCATION_NURSERY,20 EDUCATION_SCHOOL,21 EDUCATION_UNIVERSITY,22 HEALTH_DENTIST,30 HEALTH_DOCTORS,31 HEALTH_HOSPITALEMERGENCY,32 HEALTH_HOSPITAL,33 HEALTH_PHARMACY,34 HEALTH_VETERINARY,35 LANDUSE_SWAMP,45 MONEY_BANK,46 MONEY_EXCHANGE,47 ...	FOOD_BAR,23 FOOD_BIERGARTEN,24 FOOD_CAFE,25 FOOD_FASTFOOD,26 FOOD_ICECREAM,27 FOOD_PUB,28 FOOD_RESTAURANT,29 SHOP_ALCOHOL,73 SHOP_BAKERY,74 SHOP_BICYCLE,75 SHOP_BOOK,76 SHOP_BUTCHER,77 SHOP_CARREPAIR,78 SHOP_CAR,79 SHOP_CLOTHES,80 SHOP_COMPUTER,81 SHOP_CONFECTIONERY,82 SHOP_CONVENIENCE,83 SHOP_COPYSHOP,84 SHOP_DEPARTMENTSTORE,85 SHOP_DIY,86 SHOP_FISH,87 SHOP_FLORIST,88 SHOP_LAUNDRETTE,97 SHOP_MARKETPLACE,98 SHOP_PHONE,99 SHOP_MOTORCYCLE,100 SHOP_MUSIC,101 ...	TOURIST_ARCHAELOGICAL,137 TOURIST_ART,138 TOURIST_ATTRACTION,139 TOURIST_BATTLEFIELD,140 TOURIST_BEACH,141 TOURIST_CASTLE,142 TOURIST_CASTLE2,143 TOURIST_CINEMA,144 TOURIST_FOUNTAIN,145 TOURIST_INFORMATION,146 TOURIST_MEMORIAL,147 TOURIST_MONUMENT,148 TOURIST_MUSEUM,149 TOURIST_RUINS,151 TOURIST_THEATRE,152 TOURIST_THEMEPARK,153 TOURIST_WINDMILL,156 TOURIST_WRECK,157 TOURIST_ZOO,158

After the first filtering which reduce 80% POI and remain 389 POI are useful. But we find that some POI is very close together (see Figure 21) and it is not good for route suggestion because the point is too close together.

Figure 21: Nearby POI points

The system supposed each place's distance between other not less than 0.5km. It find out the distance between others. When the point too close then it will choose one point that type is higher (see Table 1).

Distance = D

Current Point = (Lan_1, Lon_1)
Selected Point = (Lan_2, Lon_2)

$$D = 3956 \times 2 \times \sin^{-1}\sqrt{\sin^2\left(\frac{(lat_1 - |lat_2|) \times \pi}{180 \times 2}\right) + \cos\left(\frac{lat_1 \times \pi}{180}\right) \times \cos\left(\frac{|lat_2| \times \pi}{180}\right) \times \sin^2\left(\frac{(lon_1 - |lon_2|) \times \pi}{180 \times 2}\right)}$$

Finally, the system filter out 173 points for the following match photos process.

5.2.3 Mapping photos into POI

5.2.3.1 Introduction of Minimum bounding rectangle

The minimum bounding rectangle (MBR) is using a min rectangle to build a bounding and cover the target area. (See Figure 22)

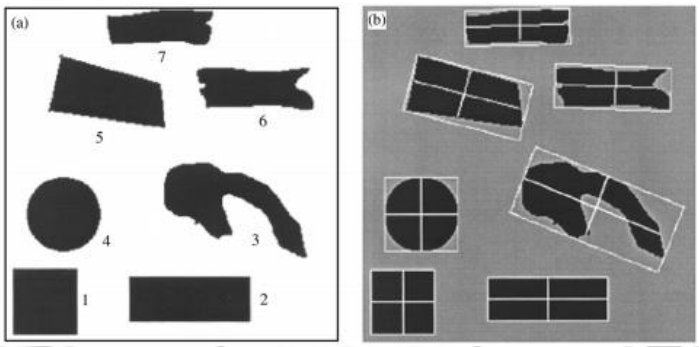


Figure 22: Using MPR to bounding rectangles of each object

After reduce dispersed photos we can see the photos data are group together (See Figure 22) and we have to marking each group of photos using MBR (Minimum bounding rectangle) and using this group to mapping with those filtered POI.

We the photo filtering part cannot filter all point in there, so that when we using MPR to process the data and we find that the bounding rectangle area is smaller than other 20% then the program will skip this small part. It is because this small part mean the data is not enough if we keep this part to the follow processing and we will find that the visiting time result is not acceptable.

After we mapping the grouped area into the filtered POI data. We can have the follow result (Figure 23). In this result, some POI cannot map into the area and some area cannot map into POI. It mean that some location OpenStreetMap data source is not fully complete and some POI is a viewpoint but less people visiting and take photo in that place.

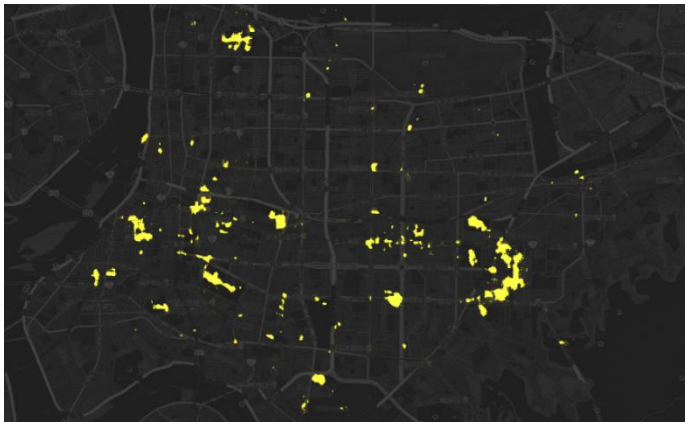


Figure 23: Reduce dispersed photos

Figure 24: An area using MPR sample.

Figure 25: Mapping the grouped area into the POI

Finally, we can have this POI map which is the result of filtering module and photos mapping module processing. And this is the final version using in route processing with three module.

Figure 26: POI plotting after mapping

5.2.4 Popular visit time in a place

For the route recommendation, time control also the important part of it. But it is difficult to find all the viewpoint opening hour one by one and impractical. Now we have huge amount of photos and it can mining normal visiting time from other travellers. For this module, it is using 4 years data to statistics and figure out the most popular visit time in a place. We statistics each place in 4 years total photo around 5000-8000. This module separate 24 parts for each hour and calculate the photos order by the taken time in that place. Also calculate the average taken photos in each hours. We have to prepare the number of photo in 24 parts then find out the time characteristic. *National Taiwan Museum* and *Shihlin Night Market* are chosen for an example once is because those place have different characteristic in the taken time order. (See Figure 27 and 28).

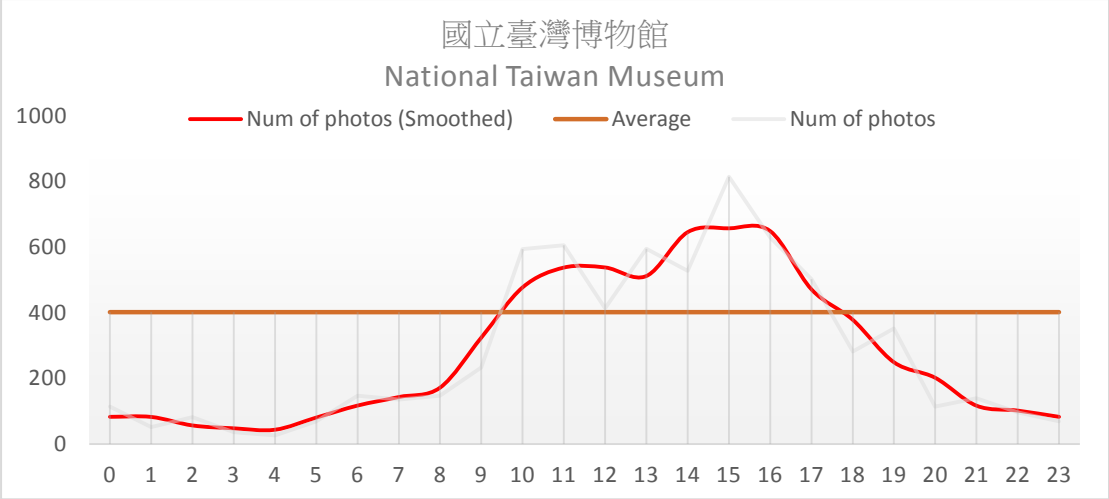


Figure 27: Calculate 4 years photo data in NTM and order by the taken time (smoothed)

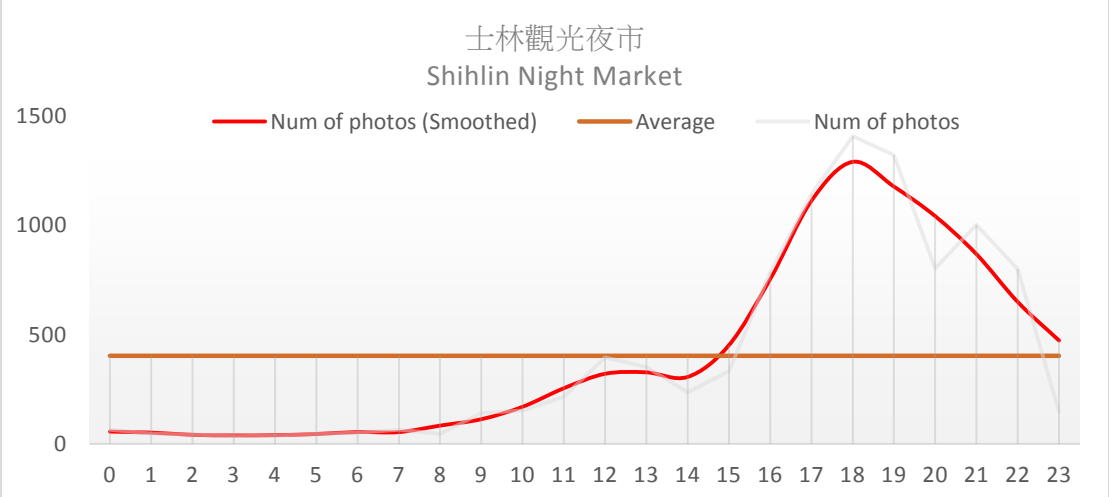


Figure 28: Calculate 4 years photo data in SNM and order by the taken time (smoothed)

For the verification of the module, we searched the National Taiwan Museum official opening time is 9:00-18:00. We suppose this module can calculate the popular visiting time in NTM. In this module result (Figure 27) show that the peak hour of taken photo in NTM is around 10:00 to 19:00. The result and official opening hour is similar. So the result is reasonable be the visiting time.

One more place for testing using this module is Shihlin Night Market where is no official opening time but most restaurant and shop are opening at evening still mid-night. This place estimate visiting time is latest than the NTM and in the graph (Figure 28) show that the peak hour of taken photo is around 16:00 to 23:00 which also reasonable because the opening hour is evening. So that we know this place visiting hour is around 16:00-23:00.

Calculate the popular visiting hour

A. Let P as all photos in a place

B. H_1 to H_{24} as number of each hours

C. $\bar{P} = \frac{P}{24}$ Calculate the average taken photos in that place.

D. Find the first and last $H_{1\text{-}24} >= \bar{P}$ as populate visiting time.

5.2.5 Calculate the stay time

For this module, we have to find out each location and calculate the staying time in different time. We separate the time in 4 parts and analyse the average stay time by user first and last taken photo in that place.

L as Location, $\boldsymbol{U_x}$ as all user, $\boldsymbol{T_{1\text{-}4}}$ as 4 time parts.

Algorithm

```
1.   while L
2.          for U1 to Ux
3.                  switch T1, T2, T3, T4 {
4.                         Select first_taken_time and last_taken_time;
5.                         Stay_time = last_taken_time - first_taken_time
6.                  }
7.          }
8.   }
```

Table 2: Time separation for calculate the staying time

Morning	Afternoon	Evening	Early morning
6:00-11:59	12:00-17:59	18:00-23:59	00:00-05:59

Table 3: Calculate each location staying time in different time

國立臺灣博物館 National Taiwan Museum				
	Morning (min)	Afternoon (min)	Evening (min)	Early morning (min)
User A	238	121	53	4
User B	223	199	52	5
User C	190	193	56	1
User (A,B,C,D,E,F,G…..) All user visited in that place …				
Average	217	171	54	3

5.3 Route processing and generation

5.3.1 Association Rule

Association rule is usually used in market-basket analyse. The purpose is to figure out the trends from the data. The data we analyse always is large, discrete and also heterogeneity. There are two important parts in association rule, one is support and another one is confidence.

5.3.2 Rebuild route

We have rebuild the user's travel route in the past. Therefore our input is the collection of record in database and the output should be the collection of route.

5.3.3 Lift calculation

Genetic algorithm is suitable routinely used to generate useful solutions to optimization and complex search problems. So what does this algorithm doing? "Survival of the fittest" is the core idea of genetic algorithm. This algorithm is using some operation to figure out the fittest gene under specific condition in the population. There are a few important operations in genetic algorithm. They are selection, crossover and mutation.

5.3.4 Fitness functions

The fitness functions are distance, weight and time. Distance fitness function is using distance between two points of interest as measurement, the shorter distance get the higher score. Weight fitness function is using the weight between two points of interest as measurement, and the weight between them are calculated by association rule, the confidence between two points of interest will have a positive relation with the score. Time fitness function is using the time information of point of interest as measurement. The time information of point of interest for measurement is opening time, closing time. The time that the user wants to visit is within opening time that will get the higher score. In addition to this, we also have their combinational fitness functions: distance with time, distance with weight and also distance with time and weight.

CHAPTER 6. System evaluation

6.1 Different method evaluation

6.1.1 Distance method evaluation

For the distance method is compare the total distance of the route and find out the shortest route. But there are many unacceptable order such as it arranged to visit night market (see Figure 29 *士林觀光夜市*) at morning and arranged to visit temple (see Figure 29 *東和禪寺*) at night. Actually, night market have to arrange at night or nightfall time and temple should be at morning or afternoon because different location have different opening time or recommend sightseeing time. If the system suggest in incorrect time for visitor then they will feel confused. So that only consider the distance between the locations is very not enough for the router suggestion. And it should be use some more method to optimize the travel route.

Figure 29: Distance method evaluation

6.1.2 Weight method evaluation

The weight method is to calculate each location relational rank and compute a highest related route which mean all location are related and many other traveller using this route before. But we can see the result of this method which is easy to find out the defect in the route. The route is very disorder and difficult to follow this route because the distance between each location is very high. For example, after the first location *新光三越百貨公司* then the system suggested a farthest location to user which is the second location *士林觀光夜市* (see Figure 30), then go to third location *台北火車站*. But we can find that the first location and third location are most near. Actually, after the first location we should go to third location *台北火車站* before second location *士林觀光夜市*. Also the second location should not suggest to visiting at morning because for the opening time. Beside of this problem, the distance of the route is not good because some we can see some location that user have to go retreat.

Figure 30: Weight method evaluation

6.1.3 Time method evaluation

The time method is to calculate the acceptable visiting time to user by other traveller's experience. User follow this order can go all location on the opening and popular visiting time. But there are some problem is about the total distance and relation between each location. Some location also have to go retreat to the following location that is not a good route for user. User may have to spend more traffic time between locations. And it should be improve.

Figure 31: Time method evaluation

6.1.4 Combine method evaluation

This method is combined three method together which method are distance, weight and time method. We try to test this method using same POIs as the period evaluation. So the result can compare with other result. In this testing result, we find that the route is the best, all location can visit on popular visit time and also can stay more time in every location. For the total distance, it is a direct route and there are no situation that user have to go retreat to visit the following location. User can save more time in traffic and spend more time in the viewpoint. The route is shorter that using weight and time method. The time control is very good. Combine all method and which can balance the method advantage and reduce the weak point in the route. So that this result is acceptable and able to suggest user in good travel route.

Figure 32: Combine distance, weight and time method evaluation

6.2 Supplement method evaluation

6.2.1 Visiting time evaluation

Visiting time is to calculate the location popular visiting time, if the route cannot arrange a suitable time to this location using the time method, the distance and weight method will arrange a shortest way and related to the suggest route. For example (see figure 33), the user select the starting visiting time is very late (starting time: 16:00) but want to visit 8 location. Then the system will generate a route with 8 points but some location may be over we will try to find out the most location you can visit. When some location the visiting time is over then we will show that for you (see figure 34).

Figure 33: Visiting time evaluation

Figure 34: Visiting time over

6.2.2 Staying time evaluation

The staying time calculation method are separate 4 part per day. Each part of time will analysis the period traveller staying time. So that in different arrive time will show you different staying time. (see Figure 35) When the route suggest you go to *士林觀光夜市* at morning then the suggest staying time is ***56min***. If suggest you at night then the staying time of this location will be ***98min.***

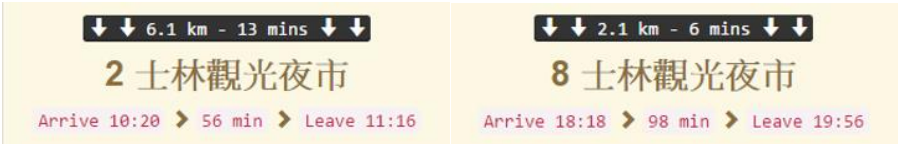


Figure 35: Staying time evaluation

CHAPTER 7. Ethics and professional conduct

7.1 Privacy

We use our fetch data module in our route recommendation system fetched a great number of photos from Flicker. The photos we fetched are public to outside. Anyone can access them. Therefore, we use these photos as our source data. We untie the information in photo. We use these information for analyse.

However, the action we did is invasion of privacy?

The user posted his photo in public on Flicker. Therefore, we have right to access the photos and we an able to download it for analyse use. However, we must follow the rule 1.2. Make sure that our system only uses their photo as analyse but not use their photo as some illegal purpose so that harm the photos owner.

1.2 Avoid harm to others

7.2 Unauthorized Access

In this project we are using Flickr and OpenStreetMap as the main data source. The both of them are legal for all users to access their public data for free. If one day, we using this project to found a profit company. One day, Flickr announce that all data are close the access for public and only able to using for studies and non-profit use. It means that if our company continues using Flickr data and we have to pay. But you discover that your id is using student level and the access permission is allow have you to access their data in free.

According rule 1.3, 2.3 and 2.6, we have to be honest if we collect the data using student identity but on my profit project which is broken the rule with Flickr. It is because their data is only for studies and non-profit use. So that I cannot use illegal way to collect the data, we have to pay for that.

1.3 Be honest and trustworthy

2.3 Know and respect existing laws pertaining to professional work

2.6 Honour contracts, agreements, and assigned responsibilities

CHAPTER 8. Conclusion

Throughout the software development phases of this project, we understand that in order to complete the task well and on time, commutation with each other is essential. A good communication could bring along a good atmosphere, and increase the efficiency. We divided the task into two parts based on the functional difference. Although most of the job is separated and can be done independently, but we always come together to discuss the solution and idea of the problems we meet, and the way of our developing method. We know that the opinions from the group mates are important, so we choose to hold meetings constantly to have the effect like brain-storming.

In the programming aspect, it is our first experience to fetch such great number of data from the internet and deal with it. We met several problems that the program does not come out with the result we expect. Since our source data social network site has limitation of the photo. We only can fetch 5,000 photos per hour and also the photos will have opportunity that gets the photo that we already have. We have to care about the execution flow of program statements very carefully. In the system structure, we are using Model-View-Controller framework to construct our web-based information system, now we have a general idea of MVC, but we still can improve the structure of our web pages. Frameworks components can be separated into smaller pieces then take the advantage on separating system functions and easier maintenance.

From cooperation we get lots of useful information and idea, more communications and discussions opened our eyes, enriching our knowledge and helped to make the system to be born in a better way. As the final and biggest project in the university life, we think this project is not only testing about out design and development power, but also tests us the cooperation ability, communication skill and team spirit. We believe that teamwork is very important in software development, as most of the IT people are working in the form of team and group. For the sake of future career, it is a good opportunity for us to be working in a team.

We are very thankful to our supervisor who has given his greatest support to our team. Without his experience in system design and user experience, we could be able to finish our system successfully. Almost for the whole year he is showing his patience, professional skills, experience on software, and his heart in education. We are very content to have him as our supervisors.

This project gives a chance for us to unify the things we have learnt throughout this major in the university life. At the first two years of university life, we have learnt many theories and techniques, but actually there are not much big projects for us to fuse all the knowledge learnt from each course. Now this final year project has given us a good chance to realise what is ‘software’, from zero to all, from ideas to real product, to accomplish this project, we realized what we have learnt, in order to put them all together to get the work done.

To conclude, this project is a good experience to our team. It gives our team a chance to communicate with the others, to try to understand the user requirements well so that less conflict will come out later. We had used some programming languages that we have not used in school projects before in the development stage. It had great improved our programming skill and knowledge on different type of programming language. Though there are many improvements can be done from the system in this project. We hope that later version of this system prototype could appear in the society and contributes as much as possible in the environmental protection.

REFERENCES

1. https://www.flickr.com/services/api/
2. http://planet.openstreetmap.org/
3. Ref: http://wiki.openstreetmap.org/wiki/Points_of_interest
4. Ref: https://github.com/MorbZ/OsmPoisPbf/blob/master/poi_types.csv
5. http://en.wikipedia.org/wiki/Minimum_bounding_rectangle
6. http://transportnsw.info/
7. http://www.theaa.com/
8. http://reittiopas.fi/
9. http://www.vztaiwan.com/
10. http://plan.qyer.com/
11. A simple method for fitting of bounding rectangle to closed regions (D. Chaudhuria, A. Samalb) Pages 1983-1984, 2007. http://www.researchgate.net/profile/D_Chaudhuri/publication/222676553_A_simple_method_for_fitting_of_bounding_rectangle_to_closed_regions/links/53fd9c900cf2364ccc08dc86.pdf