Categorization of Users Based on Trajectory Data

AKANSHA AGARWAL

Roll No. 14IT60R14

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

West Bengal, India

April 2016

Categorization of Users Based on Trajectory Data

A Thesis submitted in partial fulfilment of the

requirements for the degree of

Master of Technology

in

Information Technology

by

AKANSHA AGARWAL

Roll No. 14IT60R14

under the supervision of

Dr. Soumya Kanti Ghosh

Department of Computer Science and Engineering

Indian Institute of Technology, Kharagpur

West Bengal, India

April 2016

DECLARATION

I, **AKANSHA AGARWAL**, Roll no **14IT60R14**, registered as a student of M.Tech. program in the Department of Computer Science and Technology, Indian Institute of Technology, Kharagpur, India (hereinafter referred to as the 'Institute') do hereby submit my thesis, title: **Categorization of Users Based on Trajectory Data** (hereinafter referred to as 'my thesis') in a printed as well as in an electronic version for holding in the library of record of the Institute.

I hereby declare that:

1. The electronic version of my thesis submitted herewith on CDROM is in PDF format.

2. My thesis is my original work of which the copyright vests in me and my thesis do not infringe or violate the rights of anyone else.

3. The contents of the electronic version of my thesis submitted herewith are the same as that submitted as final hard copy of my thesis after my viva voice and adjudication of my thesis on 02-05-2016.

4. I agree to abide by the terms and conditions of the Institute Policy on Intellectual Property (hereinafter 'Policy') currently in effect, as approved by the competent authority of Institute.

5. I agree to allow the Institute to make available the abstract of my thesis in both hard copy (printed) and electronic form.

6. For the Institute's own, non-commercial, academic use I grant to the Institute the non-exclusive license to make limited copies of my thesis in whole or in part and to loan such copies at the Institute's discretion to academic persons and bodies approved of from time to time by the Institute for non-commercial academic use. All usage under this clause will be governed by the relevant fair use provisions in the Policy and by the Indian Copyright Act in force at the time of submission of the thesis.

7. Furthermore

(a) I agree to allow the Institute to place such copies of the electronic version of my thesis on the private Intra-net maintained by the Institute for its own academic community.

(b) I agree to allow the Institute to publish such copies of the electronic version of my thesis on a public access website of the Internet should it so desire.

8. That in keeping with the said Policy of the Institute I agree to assign to the Institute (or its Designee/s) according to the following categories all rights in inventions, discoveries or rights of patent and/or similar property rights derived from my thesis where my thesis has been completed:

a. with use of Institute-supported resources as defined by the Policy and revisions thereof,

b. with support, in part or whole, from a sponsored project or program, vide clause 6(m) of the Policy.

I further recognize that:

c. All rights in intellectual property described in my thesis where my work does not qualify under sub-clauses 8(a) and/or 8(b) remain with me.

9. The Institute will evaluate my thesis under clause 6(b1) of the Policy. If intellectual property described in my thesis qualifies under clause 6(b1) (ii) as Institute-owned intellectual property, the Institute will proceed for commercialization of the property under clause 6(b4) of the Policy. I agree to maintain confidentiality as per clause 6(b4) of Policy.

10. If the Institute does not wish to file a patent based on my thesis, and it is my opinion that my thesis describes patentable intellectual property to which I wish to restrict access, I agree to notify the Institute to that effect. In such a case no part of my thesis may be disclosed by the Institute to any person(s) without my written authorization for one year after the date of submission of the thesis or the period necessary for sealing the patent, whichever is earlier.

Akansha Agarwal
Department of Computer Science and
Engineering,
Indian Institute of Technology, Kharagpur.

CERTIFICATE

This is to certify that this thesis entitled **Categorization of Users Based on Trajectory Data**, submitted by **AKANSHA AGARWAL** to Indian Institute of Technology, Kharagpur, is a record of bonafide research work carried under my supervision and I consider it worthy of consideration for award of the degree of Master of Technology of the Institute.

Dated:

Dr. Soumya Kanti Ghosh
Department of Computer Science and
Engineering,
Indian Institute of Technology, Kharagpur.

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude and sincere thanks to my advisor, ***Dr. Soumya Kanti Ghosh*** for his inspiration, encouragement and able guidance all throughout the course of my M.Tech Project. It is because of his valuable advice from time to time and constant support that today I have been able to give shape to my work. It is indeed an honour and a great privilege for me to have worked under his guidance which has made my research experience productive. I learned an approach of humanity, perseverance and patience from him.

I sincerely acknowledge my deepest gratitude towards all faculty members of School of Information Technology for providing in-depth knowledge on various subjects over the past two years. It was a pleasure to learn and work with their co-operation.

Lastly and most importantly, I am grateful to my beloved father ***Sanjeev Kumar Agarwal*** and my beloved uncle ***Rajeev Kumar Agarwal*** for their unconditional love and constant encouragement which has given me the strength to complete this thesis.

Dated:

AKANSHA AGARWAL
Department of Computer Science and
Engineering,
Indian Institute of Technology, Kharagpur.

ABSTRACT

The advancement in the GPS related devices enable people to log the location histories they visited with spatial temporal data. Also with this the interests of people have shifted from analysing the raw trajectories to the study of trajectories at the semantic level according to the application purpose. This study can be used for various purposes like examining the human periodic behaviour, frequently visited regions, pattern followed by the users. This report discusses the basic terminologies related to the trajectories and movement related data, analysing the trajectory data by different points of view like Distribution of trajectories by distance, Data collection time etc. and the discussion of various algorithm and techniques.

For Experimentation purposes data is first cleaned to remove the outliers. Further the trajectories are segmented to find the stop points and the segmented trajectories are enriched semantically. Using various approaches for analyzing semantic trajectories and extracting the knowledge about their characteristics, frequent visited places are find and frequency of visits of various users for these frequently visited places is calculated in order to cluster the users on the basis of the start and stop points of the trajectories. In order to verify the results of clustering, another approach for clustering is used that is clustering the users based on whole trajectories as one atomic unit by applying Dynamic Time Warping (DTW).

Key words: Geo-tagging, Semantic trajectory, Clustering, Dynamic Time Warping

Contents

Declaration iii

Certificate by the Supervisor v

Acknowledgments vi

Abstract vii

List of Figures x

1 Introduction 3

2 Review of Literature 7

3 Preliminary Concepts and Terminologies 11

3.1 Data Model 12

4 Data Pre-processing 15

4.1 Cleaning of trajectory data /Noise Filtering 16

4.1.1 Mean Filter 16

4.1.2 Modification of Data 17

4.2 Trajectory Segmentation and Stop point Identification 18

4.3 Geo-tagging of Trajectory 20

4.4 Normalizing Geo-tagged data 20

5 Trajectory Analysis **22**

5.1 Finding frequently visited Places 24

5.2 Ranking of locations using HITS analysis 26

5.3 User Similarity and Trajectory Clustering 31

5.3.1 On Basis of Start and End Points 31

5.3.2 On the Basis of Whole Trajectory as an Atomic Unit 34

6 Conclusion and Future Scope **40**

List of Figures

1.1 Geographic trajectory example 4

1.2 Semantic trajectory example 5

2.1 Geographic trajectory example 9

3.1 Distribution of Trajectories 12

4.1 Data Pre-processing 15

4.2 GPS Trajectory and Erroneous Points 16

4.3 Normalization for Semantic Tagging 21

5.1 Geographic Level Analysis 22

5.2 Application Level Analysis with establishment as one of the POI's . . . 23

5.3 Application Level Analysis without considering establishment as POI's 24

5.4 Hierarchical Graph Modeling User Location 27

5.5 Cluster Representation 28

5.6 Location History of User 0 28

5.7 Two Layered Graph of Users and Locations 29

5.8 Hierarchical Clustering for Start and Stop Locations 32

5.9 Optimal Number of Clusters 33

5.10 User 1 Travelling Experience 33

5.11 User 2 Travelling Experience 33

5.12 Comparison of DTW with other similarity measure 35

5.13 DTW Concepts 36

5.14 Hierarchical Clustering on the basis of DTW 38

5.15 Optimal Number of Clusters 39

Categorization of Users Based on Trajectory Data

Chapter 1

Introduction

On one side when location-procurement technologies such as GPS, GSM network, sensors etc., lead to the collection of large spatial temporal datasets and increase in penetration of these datasets gives the chance of discovering valuable knowledge about movement behaviour and patterns. Other side the increased competition and the rapid growth in the services provided by social networks leads to the development of various recommendation systems. For example the several social networking sites such as Facebook, Twitter , ecommerce sites like Amazon, Paytm , not only provide various friend search option but help us in finding people who have similar interests and share the same locations. In last few years many location based social networking sites not only consider users locations but also show interest in the trajectories of the users.

Nowadays, it has been observed that there is a large adoption of smart phones and GPS enabled devices so it has become expedient to collect and analyse the movement related data of an object in geographical space. Movement related data basically deals with the trajectories followed by the moving object such as cars, buses, pedestrians etc. The data collected from various devices is raw and needs to be processed further for the purpose of combining with the contextual data to deal with Map networks.

Analysing movement data for security applications by capturing the foot prints of the users through security cameras, following the path followed by users through sensors can be useful to detect anomalies among the various trajectories for intrusion detection. Collecting trajectories of vehicles can be used for traffic optimization and management like rerouting the traffic from highly congested areas to lesser congested areas. Trajectory analysis can also be used for better service and quality e-business with tracking of goods for shipment.

Trajectory analysis research has enabled us to develop tools and techniques for various fields. Some of the areas of noticeable research include

- **Clustering of Trajectories** Based on similarity of trajectories, grouping the moving objects into various clusters and finding the common trajectories.
- **Classification of Trajectories** Predicting the class labels of the clusters formed by clustering of trajectories based on different features.
- **Anomaly Detection** Detecting the intruder from significant set of moving objects by identifying the abnormal or the suspicious behaviour.
- **Semantic Annotation/tagging** Tagging the trajectories with the landmarks passed by so as to enrich the trajectories with the semantic data.
- **Frequently Visited Locations** Finding the locations which are frequently visited by the users and have high ranking as compared to others.
- **Analysing Periodic behaviour** Finding the repetitive behaviour and activity at any location at particular time interval
- **Predicting Locations** Predicting the locations to be visited by the users in future.

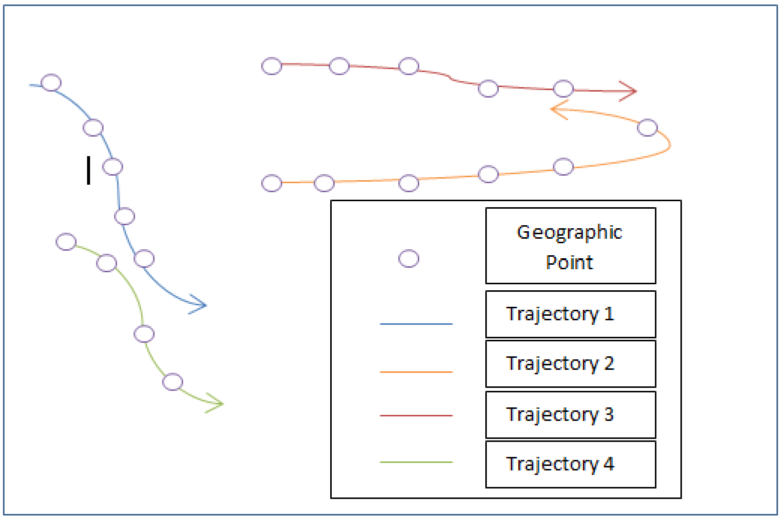


Figure 1.1: Geographic trajectory example

Most of the studies relate to trajectories define trajectory as a path taken by the user to visit different places. It is geographical capturing of data which depicts a user's physical moving behaviour in real world. When any trajectory is seen at the abstract level it is just the sequence of multiple latitudes and longitudes collected at the particular timestamp i.e. Considering only spatial-temporal features as shown in Figure 1.1 but when analysed at the concrete level can be mapped onto the various semantic features along with the consideration of the various geographical and application domains as shown in Figure 1.2.

Earlier studies discuss mobile users similarity only in terms of their trajectories so here most of the discussion focuses on analysing the geographical features of user trajectories[1][2][3]. As mentioned before, a geographic trajectory is typically a sequence of geographic points captured at a particular time value (represented as <latitude, longitude, timestamp>). So measuring the user similarity only on the basis of latitude and longitude basically means persuading it by the geographic properties of the trajectory data i.e. defining the similarity only on basis of the geographic distance between the trajectories which is not at all a sufficient feature. For example in Figure 1.1, Trajectory 1 and Trajectory 4 are closer and Trajectory 2 and Trajectory 3 are closer thus Trajectory 1 and Trajectory 4 are similar and Trajectory 2 and Trajectory 3 are similar. But when dealt with sematic trajectories (i.e. the sequence of locations with semantic tagging such as Restaurant, School, Park, etc. and mapping of geographic points to the landmarks passed by) shown in Figure 1.2 it is found that the results are quite different. It is observed that both Trajectory 1 and Trajectory 2 represent the sequence <Dormitory, Restaurant, University> while Trajectory 4 represent the sequence <Home , Bank, Shop>. The semantic behaviour of Trajectory 1 and Trajectory 2 is quite the same and of Trajectory 1 and Trajectory 4 is quite different thus we can conclude that Trajectory 1 and Trajectory 2 are more similar to each other than to Trajectory 4. So we observe

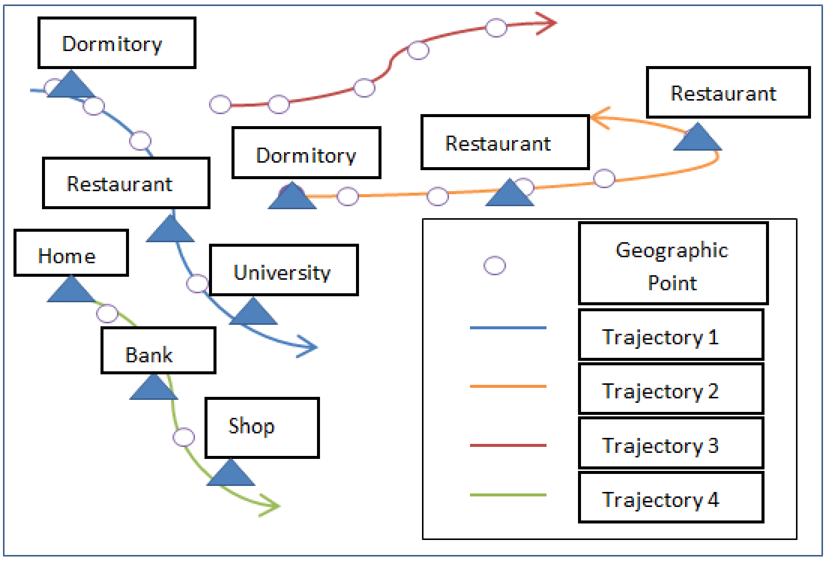


Figure 1.2: Semantic trajectory example

the trajectories by two insights:

- **Geometric overlaps** People following the same path in the geographic locations might have same interests for example in figure 1.1 Trajectory 1 and Trajectory 4 users.
- **Semantic Overlaps** Some users might not follow the same path but may have the same interest if they follow same movement pattern in the semantic locations For example in figure 1.2 Trajectory 1 and Trajectory 2 users.

In Chapter 2, we include a survey of related literature. In Chapter 3, we present the analysis of our dataset, that for how what period data is collected, how much area is covered by the trajectories collected etc. In Chapter 4, we present all the steps required for preprocessing of the data like cleaning, modification etc. Chapter 5 contains our main problem challenges i.e. how this trajectory data can be used for solving different problems. Finally, we conclude the work in Chapter 6 and provide directions for future research.

Chapter 2

Review of Literature

Many researches in past had discussed about the Trajectory similarity measurment [4] and user similarity measurement [1][2][3]. In [4], a Partition-and Group algorithm is proposed by Lee et al in which similarity between two trajectories is calculated. Partition-and Group algorithm consist of two phases Partitioning Phase and grouping Phase. First the characteristic points of each trajectory are found and the trajectory is divided into line segments. Once we have line segments three different distance measures, i.e., perpendicular distance, parallel distance, and angular distance is calculated on these segments so to group the trajectories into clusters. But since the distance measures can only be applied for the geographic information, concept of sematic tagging cannot be applied here for finding the user similarity. The main motive of distance based measure is to analyse the movement behaviour of the $users'$ and find the user similarity on the basis of the path followed by them.

In [3], a friend and location recommendation system is proposed by Zheng et al. To work on location and user similarity, this recommendation system considers $users'$ movement patterns in various locations and calculates the TF-IDF values for all the locations. The Geographic regions from the trajectories where the user stays for more than threshold period of time are discovered and are denoted a stay points. Later density-based clustering algorithm is used to develop a hierarchical framework for organizing these stay points into clusters. These clusters are known as stay locations/regions. These regions are used to measure the $users'$ similarities by discovering the common sequences of stay regions at each level of clustering hierarchy. For each stay location the TF-IDF value is calculated, where TF is the minimum frequency of the $users'$ who have accessed this region while IDF value represents the number of $users'$ who visited this stay location. Finally for deriving the similarity between two $users'$ the summation of TF-IDF is calculated for all stay regions.

In [1], to calculate the similarity of two mobile $users'$ an LBS (Location Based Service) Alignment method was proposed. The LBS-Alignment method uses longest common sequence between the sequential patterns followed by two $users'$ and calculates two $users'$ similarities. The Mobile Sequential Patterns common part ratio$'$ s is taken as the similarity by analysing the longest common sequence. But Similar to Partition and Group algorithm this approach considered temporal information and location history; but did not take into account the semantic tagging of locations.

After many researches based on geographic information, in recent few years, a number of studies on Semantics of Trajectories have appeared in the literature[5][6].

In [5], Alvares et al came with an idea that the geographic semantic information is explored to mine Semantic Pattern of mobile $users'$ location histories out of geographic trajectories. Firstly, stops are discovered similar to the stay points in [3] of each trajectory and these stops are mapped to semantic landmarks. Then, sequential pattern mining algorithm is applied on the sequence dataset to obtain frequent semantic trajectory patterns. These frequent semantic trajectory patterns represent the semantic behaviours of mobile $users'$.

In previous researches geographic semantic information and concept of stop points were used without considering the fact that whether these stay points point to interesting landmark or not. In earlier cases some of the stop points were unknown and hence sometimes the geographic semantic information could not discover interesting patterns. For example, as shown in Figure 3, stop1C and stop2B are not associated with any semantic landmark. Hence, Trajectory 1 is transformed as the sequence <School, Restaurant, Unknown, Park >. From the figure, it is clearly predictable that stop1C is near the Park A. Thus, by taking geometric distribution of these stops into picture, stop1C and stop1D are grouped together as Park A such that the Trajectory 1 is transformed as the sequence <School, Restaurant, Park>. Similarly in Trajectory 2 stop2B and stop2C are grouped together as Park B such that the Trajectory 2 is transformed as the sequence <School, Park, Restaurant>Hence In [6], Bogorny et al hierarchical geographic semantic information was considered in order to discover more interesting patterns.

The work[7][8][9][10] includes detecting important locations visited by the user by predicting the $users'$ movement pattern among these locations. For each location, it recognizes user-specific activities. Instead of recognizing user-customized results aim is to consider multiple $users'$ location histories. So Fosca et al. [11] came with an idea of extension to the sequential pattern mining model that analyses the trajectories of multiple moving objects. These multiple moving object shares the property of visiting to same sequence of locations at the same time period. This analysis of mining trajec-

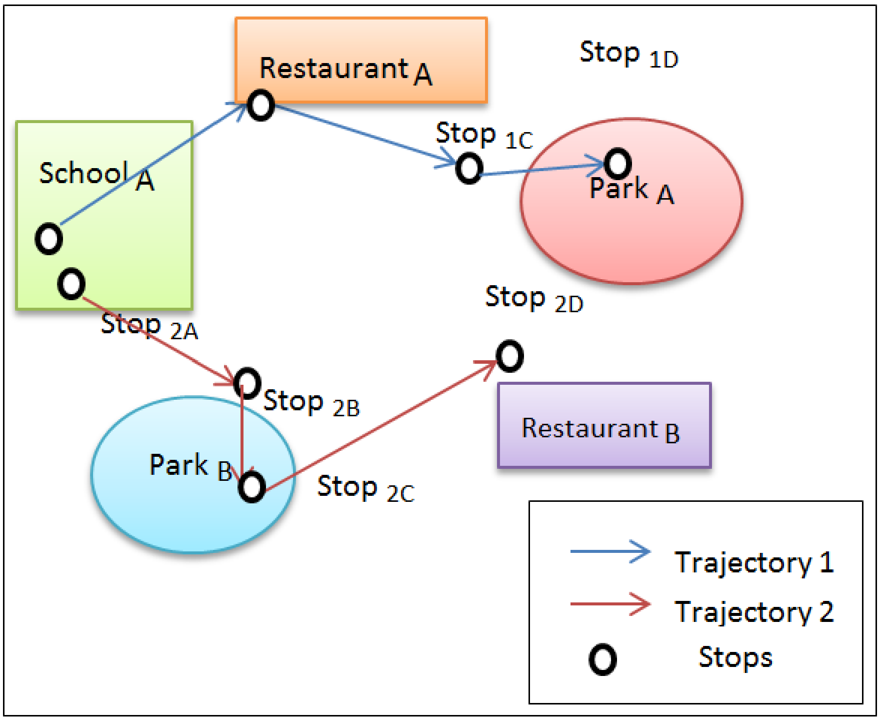


Figure 2.1: Geographic trajectory example

tories of multiple *users*′ was further used for researches. From multiple user 's GPS trajectories, history of a driver's destination and their driving behaviour was extracted by MSMLS[12] so to as to predict the next location to be visited by the driver as the trip progresses. Adding to these Zheng et al.[13] aims to conclude *users*′ transportation mode, such as walking, driving, travelling by bus etc., based on the GPS trajectories of multiple individuals. Huge amount of spatial temporal data in the form of user trajectories can be generated from the Social interactions of the users. And this data can be roughly categorized into single user trajectories and multiple *users*′ trajectories. The single user trajectories relates to *users*′ who generate the trajectories for their individual interest over a certain time period, while the multiple *users*′ trajectories concentrates on group of *users*′ who interact socially with their friends, family members, colleagues and generate the trajectories. In both cases whether it is single user trajectory or multiple *users*′ trajectories the amount of data produced is huge and therefore challenging for the analysts to interpret. For such analysis though many techniques exist in the literature, however clustering techniques are found to be the most worthy.

Clustering is a data-mining technique to partition set of data into similar and dissimilar groups called clusters while Aggregation is mining process in which the collected data is presented in the summarized form for further analysis. Cluster analysis is most common unsupervised learning which finds hidden patterns and groupings in the data and takes into account high intra cluster similarity and low inter cluster similarity. Overall objective of all types of clustering is same whether it is partitioning, hierarchical, density-based, grid-based [14], they differ only on basis of how additional parameters such as outliers, noise etc. are analysed and how different dimension dataset is studied. One of the studies evaluated different clustering techniques with focus on trajectory clustering and described each technique with their merits and demerits.

Our scenario of clustering the trajectories requires focusing on hierarchical clustering. All $users'$ stay points are collected and this dataset is hierarchically clustered into several clusters in a divisive manner. So the similar stay points from multiple $users'$ will be assigned to same clusters on different levels of clustering. Input that plays an important role to clustering algorithm is an appropriate distance metric. An evaluation was performed by Morris and Trivedi [15] and they discussed different distance similarity such as Dynamic Time Warping (DTW), Longest Common Subsequence (LCS) and Modified Hausdroff (MH).Our distance matrix uses Haversine distance and this distance is calculated between various stop points.

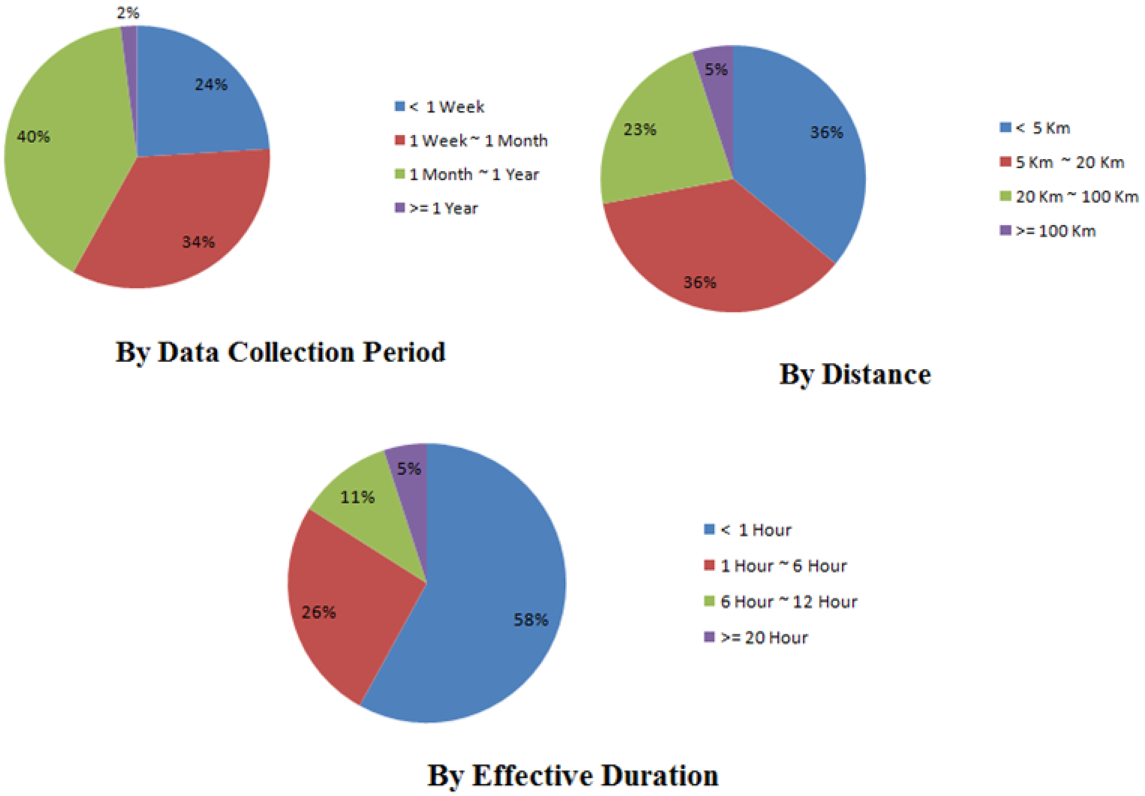
Chapter 3

Preliminary Concepts and Terminologies

For the analysis purpose Geolife GPS Trajectory Dataset is used. This Dataset is for 182 users with a total of 17,621 trajectories collected over a period of 3 years (from April 2007 to August 2010) mostly around Beijing. The format of the data analysed is [Latitude, Longitude, Altitude, Timestamp] where Latitude and Longitude are in degrees and Altitude is in feet. From the data we can easily glimpse that data is collected for every 1 to 5 seconds or every 5 to 10 meters per point.
Different types of analysis on the data leads to following observations (as shown in figure 3.1):

- Distribution of Trajectories by Distance means the area covered by the trajectories.
- Distribution of Trajectories by Data Collection Period means for how long the data is collected for particular user.
- Distribution of Trajectories by Effective Duration means the duration of each trajectory i.e. time elapsed between two stop points.

Data collected from the GPS enabled devices is called RAW DATA. The terminology better to be used here for this data is MOVEMENT PATTERN rather than trajectory. It basically consists of the sequence of points in the spatial domain taking into consideration its temporal aspects.



By Data Collection Period

By Distance

By Effective Duration

Figure 3.1: Distribution of Trajectories

3.1 Data Model

Data Collected from GPS enabled devices are further used for proposed framework. The mode of transport is not taken into consideration hence we represent the movement as user movement despite of its transport mode for example walk, by Bus, by car etc. The sampling frequency of the data is one record per 5 seconds. For some reasons if the data is found missing then it is tried to interpolate the data at the same frequency.

- **GPS record** GPS record G is a five tuple data denoted as <user_id, Latitude, Longitude, Altitude, Timestamp> where user_id is the id of the user for which the data is recorded, Latitude and Longitude are the Euclidean coordinates, Altitude is the altitude of the landmark to be considered and timestamp is the Time value/instance at which data is recorded. An Example of GPS record is <User_0, 98.23234, 116.6237, 231.763, 2009-07-14 11:22:35>.

  Latitude and Longitude coordinates are given in UTM (Universal Transverse Mercator) coordinate system. When these GPS records are ordered according to the time stamp value then we get the representation of the trajectory of any particular user.

- **Trajectory** A trajectory T of any user is defined as ordered sequence of GPS records of the user, T = G1→ G2 → G3→ .... →Gi → Gn where G1,G2 .....Gn represent the GPS record and these records are ordered by timestamp value.
  A Trajectory basically defines the movement pattern of the user. The raw data consist of set of data for all trajectories for all users but the all data points/Coordinates collected by every 5 seconds may not be point of interest (POI). From the raw data the sequence of locations/landmarks visited by the user along with the time of the visit are computed. Our interest is to find the locations/regions where the user has stayed for longer than some predefined time.

- **Stay point** A stay point P for any user is defined as a pair of <Geographic coordinate, time of visit> where Geographic coordinates represent the pair of latitude and longitude and time of visit is the particular time at which user has visited the location.
  Every time whenever user stays at a point for more than predefined time a stay point is extracted. Each Location may be visited multiple number of times by different users hence for every location multiple stay points may be obtained. Also the Geographic coordinates of the stay points at same location may not be same every time in real world but they are close to each other and within a certain a certain range.
  Once set of stay points is obtained challenge is to find the locations consolidated from this collection of stay points and also the count of visit of each consolidated location so as to rank the locations by the number of visits. Each consolidated location point to a different location at the application level for example may it is restaurant or mall or a University.

- **Semantic Location** For each stay point obtained, there are multiple locations which are pointing to the stay point. All these locations basically have a different geographic Coordinates hence to obtain the Semantic Location the centroid of these geographic coordinates is computed.
  A Semantic Location SL is Represented as a cluster of stay points and is denoted by four tuple <X, Y, S, SP> where X and Y are the centroids of the cluster , S is the Semantic Location / Semantic tagged location and SP is the set of all the geographic coordinates pointing to the particular location. For example <98.23384, 116.6287, Tsinghua University, 98.23234, 116.6237, 98.23534, 116.6337>.Once

we have got the semantic locations we are in a position to define the Location history

- **Location History** For any user location history LH is defined as a sequence LH = SL1 → SL2 →......→SLi→....... SLn of semantic locations SL. Given the raw data i.e. GPS trajectories, we can build the location histories for all users.

Chapter 4

Data Pre-processing

This section introduces four basic but important techniques for processing the trajectory data that we need to do before starting mining clustering and categorizing of data, Four basic Techniques includes Cleaning of trajectory data, segmentation of trajectory and stay point detection, Geo-tagging of trajectory and Normalizing data for Geo-tagging. This is represented diagrammatically in figure 4.1.

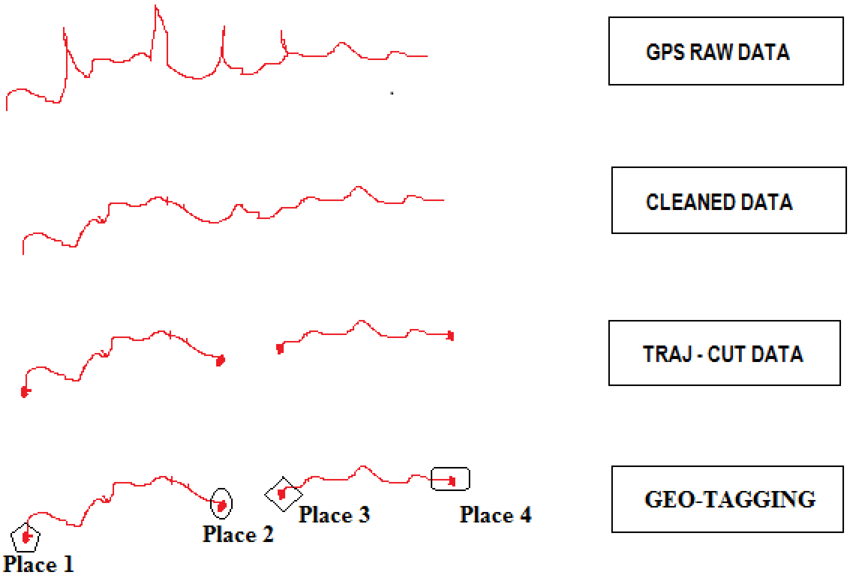


Figure 4.1: Data Pre-processing

4.1 Cleaning of trajectory data /Noise Filtering

Spatial Raw data collected from GPS devices is not reliable and perfectly accurate. It may contain noise because of sensors and other factors like poor signals at some places. Sometimes the error in the GPS trajectories is very small and is acceptable as these are very little distortions from the true value.

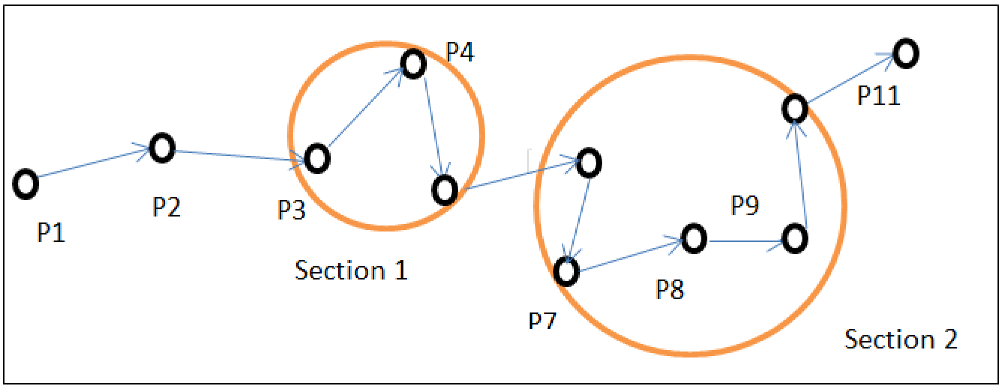


Figure 4.2: GPS Trajectory and Erroneous Points

For e.g. some of the captured GPS points of vehicle may lay out of the road where actually the vehicle was driven. These types of errors can be fixed by various smoothing methods and map-matching algorithms, which is basically a process to map a sequence of raw/collected latitudes and longitudes onto a sequence of road segments.
In other situations where the error is extremely high is not negligible. For e.g. in figure 4.2 where the point P4 is several hundred meters away from its true location and the error is too big that if we try to calculate the travel speed or derive any other useful information the result may not be correct .So we need to filter out such points before proceeding onto the further tasks. Though this problem cannot be completely solved but it is tried to minimize the noise to maximum.

4.1.1 Mean Filter

For any point to be measured, the estimated value is to be calculated by the mean of the point to be estimated and its n-1 predecessors. Basically it can be thought of a sliding window of size n and the estimated value is the mean of all these n values covered by the sliding window. The value of n needs to be decided by hit and trial method by which error can be reduced maximally. This filter is practical and gives correct result when used for individual error points like P4 as shown in section 1 of figure 4.2 but when there are multiple consecutive error points as shown in section 2 of figure 4.2, the correct value of the point to be measured can be predicted only with the help of large size sliding window .The large sized sliding window results in large deviation of the predicted value from the true value.

Also Mean filter is not at all good choice when the sampling rate of the trajectories is too low i.e. the distance between the consecutive points sampled is several hundred meters. So for the removal of outliers, mean filter was used but in a different way.
For each trajectory mean value was calculated considering all the points in the trajectory. Once we have the mean value now the mean error needs to be calculated by taking the average of all the errors of all points of trajectories (i.e. sum of deviation of the point from the mean value for all the points of trajectories). By using this mean error value the erroneous points like p4 and other can be detected and neglected by checking the error value for each point. If the error value for any point is greater than the mean error than that point is neglected. By this method even the multiple consecutive points can be detected and hence we can get the error free trajectories.

4.1.2 Modification of Data

Filtering of trajectory data includes modification also. Data is modified such that all the timestamps for which the data is collected are different and received data is sequence of increasing timestamp values. And when required interpolation of trajectories is also done.
Though when the similarity measure is applied for the trajectories, Euclidean, Manhattan or any other distance measure which aligns the i-th point on one time trajectory with the j-th point on the other, are not used. Instead DTW (Dynamic Time Warping) similarity measure is used which allows similar shapes to match even if they are not in phase in the given time zone, Still Interpolation of trajectories need to be done for the users for whom data is too less to be mapped.
Interpolation is a method of inserting new data points within the discrete set of known given data points. The new data points are inserted such that they fit with the old points. Though the data is collected for every 5 seconds but it may happen that for some users, some days, the data is too less to be mapped so we need to insert the additional sample points to make the data well sampled.

4.2 Trajectory Segmentation and Stop point Identification

Once we have reproduced the data, we have clean movement track data in hand. This data since is collected for 24/7 may contain repeated entries for same location. So next step involves cutting or segmenting the trajectories into small pieces considering various factors like GPS-gap, weekdays/weekends, distance covered, velocity etc. and especially on basis of the places where user have stayed for more than a threshold period. The data we get after cutting the trajectories is called traj-cut data and we call this algorithm as traj-cut algorithm. We call GPS-Gap in data if there is no data for few hours for any of the user because of the GPS Unavailability of data because of signal absence.

Algorithm 1 Identification of Stop Points

```
Input: GPS raw data T, distance and time span threshold as $D_T$ and $T_T$ resp.
Output: A set of Stay Points $S_P$ = {SP}
while i < NoOfPoints , do
    Set j := i + 1
    while i < NoOfPoints , do
        Distance = Distance$(Pi, Pj)$; //Calculating the distance between two points Pi and Pj
        if Dist < $D_T$ then
            ΔT = $T(Pi) - T(Pj)$; // Calculating the time span between two points
            if ΔT > $T_T$ then
                SP.(latitude,longitude) = ComputeMeanCoord( $P_k$ ; i <= K <= j)
                $S_P$ .add(SP)
                i := j ; break
            end if
            j := j + 1
        end if
    end while
end while
Return SP
```

We also have traj-cut if there is no significant movement of the user in a particular time interval or the velocity is zero for a given time interval. Threshold needs to be decided for this movement area and the time interval to be used.

By cutting the trajectories[16]we try to find the stop points. Stop points are defined as the points where the person has stopped for a period of time or we can say the place

where the person has spent some time doing his/her regular activity. These stop points can vary from application to application and from dataset to dataset. Like the dataset we have used is for university of Tsinghua so the stop points basically involve period of time in Dormitories, restaurants, schools, and some entertainment zones on weekends. So it forms a flip-flop of moves and stops. Moves involves the period for which the user travels while stops involves stop-points.
There are various methods to find stop points like centroid based method, speed based method, Duration based method and hybrid method[17] each with its pros and cons.

- Centroid based method is very similar to K-means clustering and is applied to get the locations that are important for the subject. The trajectory points are divided into k-clusters with centroid of each cluster as the stay point. But in this method the number of stay points i.e. the value of k has to be known beforehand which is nearly impossible as one can never predicts the number of stop points just from the trajectory data.

- Speed based method makes use of the speed and the locations where the speed approaches to zero are considered as stop points. Advantage of this method is that even if the Speed is not directly available from the GPS data, it can be easily calculated for the given geographic co-ordinated and time stamp value. Some limitations still arise in this method as something the speed approaches to zero even when the location is not of interest like in parking lot or in traffic condition or in case of bad weather.

- Duration based method is the most commonly used method for identifying the stop locations. In Duration based method distance between the points is calculated and at the same time duration /span is also calculated. If span between two points is more than the threshold time, those points are identified as stop points. One problem of duration-based methods is how to decide the threshold value for the method as the result is very sensitive to this threshold value.

- Hybrid method as the name suggests is the hybrid of speed based and Duration based method in which the elapsed time between the points is compared to the threshold distance and the speed is referred to approaching to zero.

The algorithm shown in algorithm 1 first checks the distance between any point and its successor point in the trajectory, if the distance is less than the distance threshold (e.g. 100 m), it then checks the time span between these two points and the velocity for the user. If the time span is larger than a given threshold, and the velocity is less than the minimum value, a stay point is characterized. So after applying traj-cut algorithm result is GPS trajectory database T = {T1, T2, T3...... Tm} for each user where Ti = {<x1, y1, t1>, <x2, y2, t2>, <x3, y3, t3>... <xk, yk, tk>}. Here < x1 , y1 , t1 > is the start point of the trajectory and < xk , yk , tk > is the end point of the trajectory.

With these stay points basically we turn the series of time stamped spatial points into the sequence of meaningful trajectories.

T = t1→ t2 → t3 → t4..... → tn S = s1→s2 → s3 → s4..... → sn

This work facilitate us to use the data for various applications like places which are frequently visited by users, For each user what are the frequent places, categorizing the users based on their trajectories.

4.3 Geo-tagging of Trajectory

Once we have traj-cut data, to prepare the data for further mining what we need to apply is the semantics on these trajectories. We do that by using the concepts of stops and moves. We extract the points of our interest i.e. the start and the stop points of all trajectories or we can the stop points which we figured by traj-cut algorithm and map these points to the application domain data or the semantic level.

The information related to the types of buildings/places from where the user starts or where he ends is extracted and the particular latitude/longitude is mapped to the most appropriate places nearby. The common points of interest are university, schools, Dormitories, Restaurants where most of the people living around the area visit.

4.4 Normalizing Geo-tagged data

While enhancing the data to the semantic level it may happen that multiple points may map to the same domain so we need to find the sparsity of the data for the particular place. Other way we can say that we need to find the threshold of the error or the error introduced in data while moving onto the semantics. For a particular place to find the error content, standard deviation of all the latitudes and longitudes pointing to that place is calculated. The value for standard deviation seems to be very small which shows that all the points of interest are represented by very small area or the sparsity of data is very less.

Since the data is not that much sparse any of the application domains can be repre-

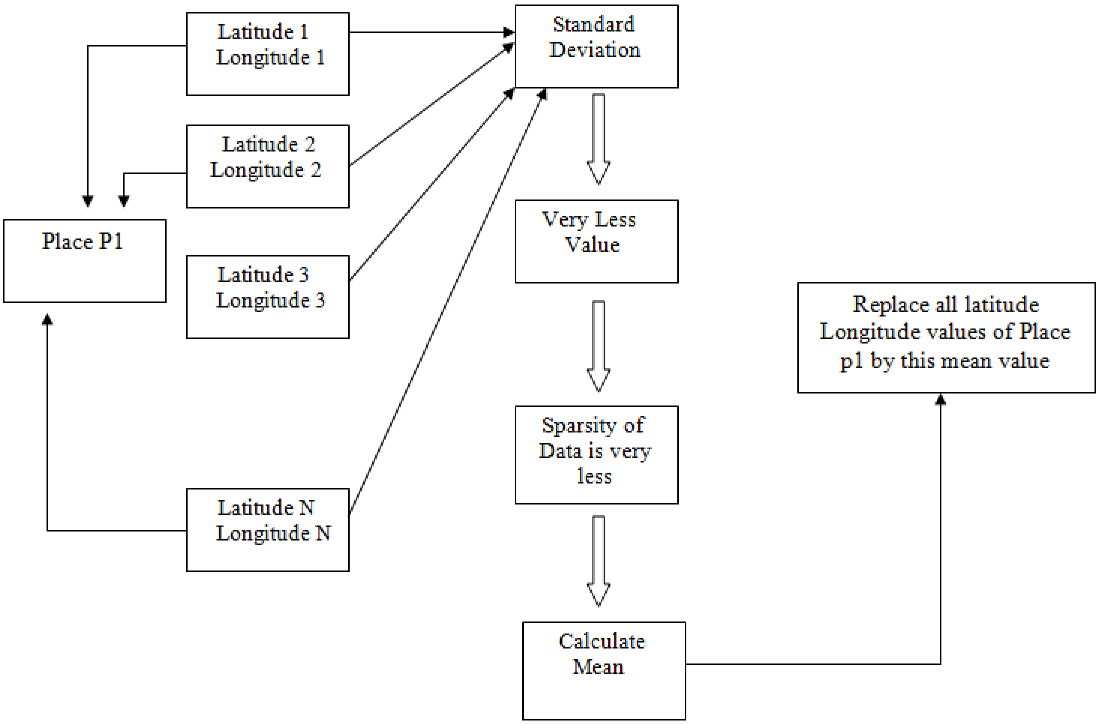


Figure 4.3: Normalization for Semantic Tagging

sented by the mean of the multiple values used for the representation of this application domain. Whole process of this normalization is represented by the diagram in figure 4.3.

Chapter 5

Trajectory Analysis

Aim of the thesis lies in finding the frequently visited places by the users for market value analysis, clustering the trajectories in order to find the anomalies in patterns followed by the users and the categorization of the users on the basis their trajectories.

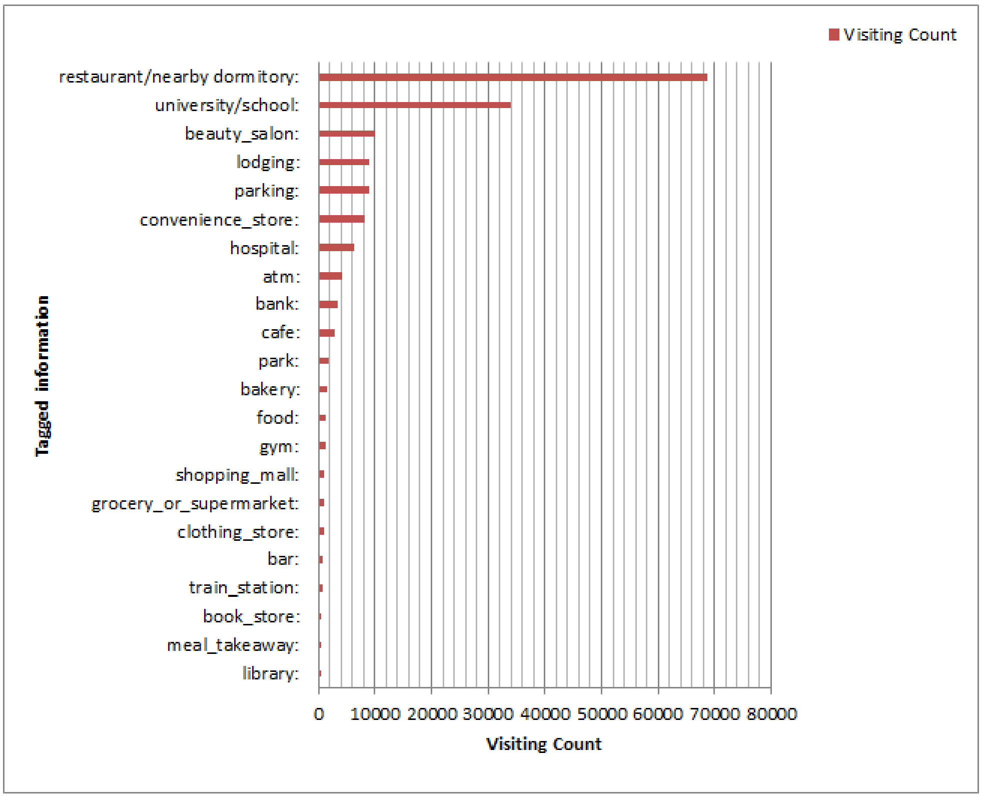


Figure 5.1: Geographic Level Analysis

For finding the frequently visited places or to find the significant places various indicators like the number of visits, the duration of the visits can be used. HITS link analysis algorithm is also used to analyse the importance of the places

These indicators of the significance of locations may be used in various ways by using the location histories of the users. Applying the concept of web search engines, the various rules can be applied further for ranking the locations like the locations that are visited along with the significant locations may also be assigned higher ranking. Similarly the users who visit this significant location may be given higher supremacy, which may then further be used for increasing the significance of the locations visited by these users.

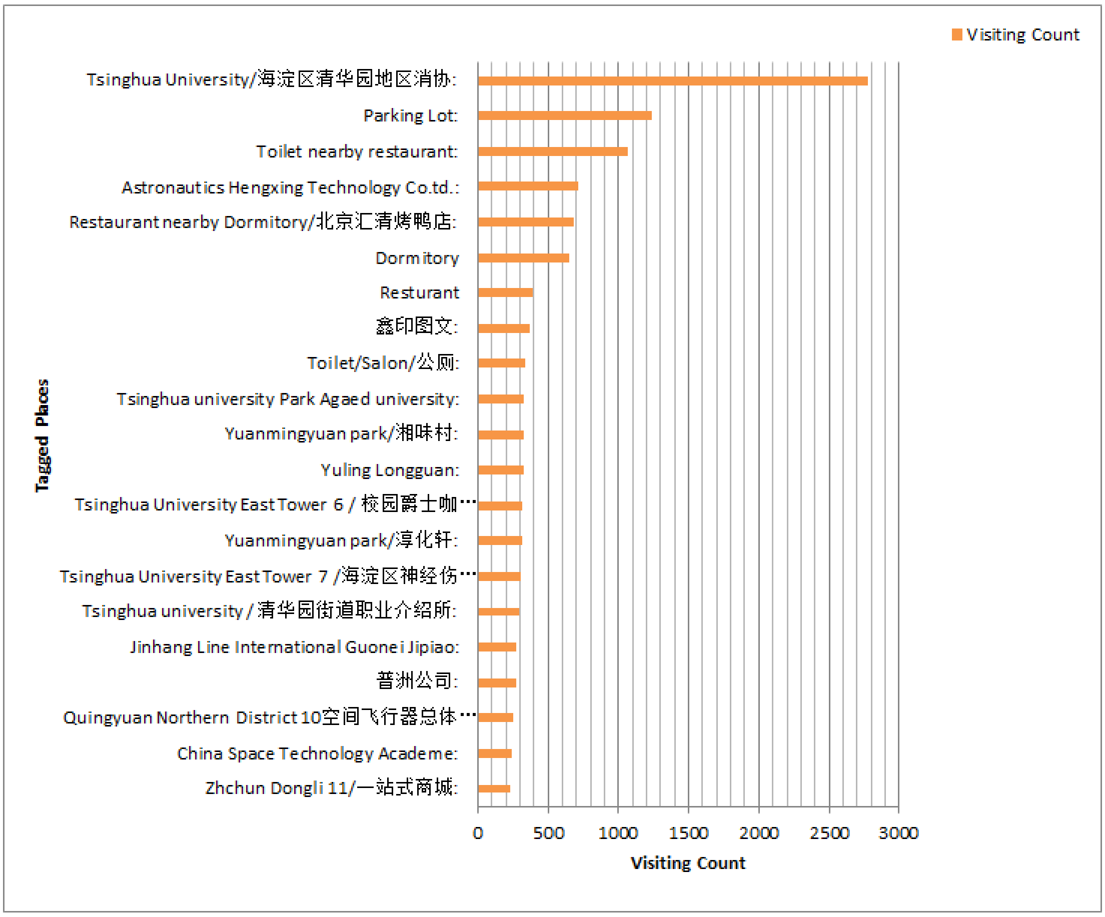


Figure 5.2: Application Level Analysis with establishment as one of the POI's

5.1 Finding frequently visited Places

For finding the significant places by this method it is assumed that more the number of times a location is visited more is the significance of the location. Ranking of the locations is done according to the number of visits by all the users. If any tie is seen, the next concept of ranking by duration is taken into consideration and tie is broken by the sum of duration for both the tied locations.

From the normalized dataset all the geo-tagged information is extracted and all the visited places are taken into account to find the top K frequent visited places.

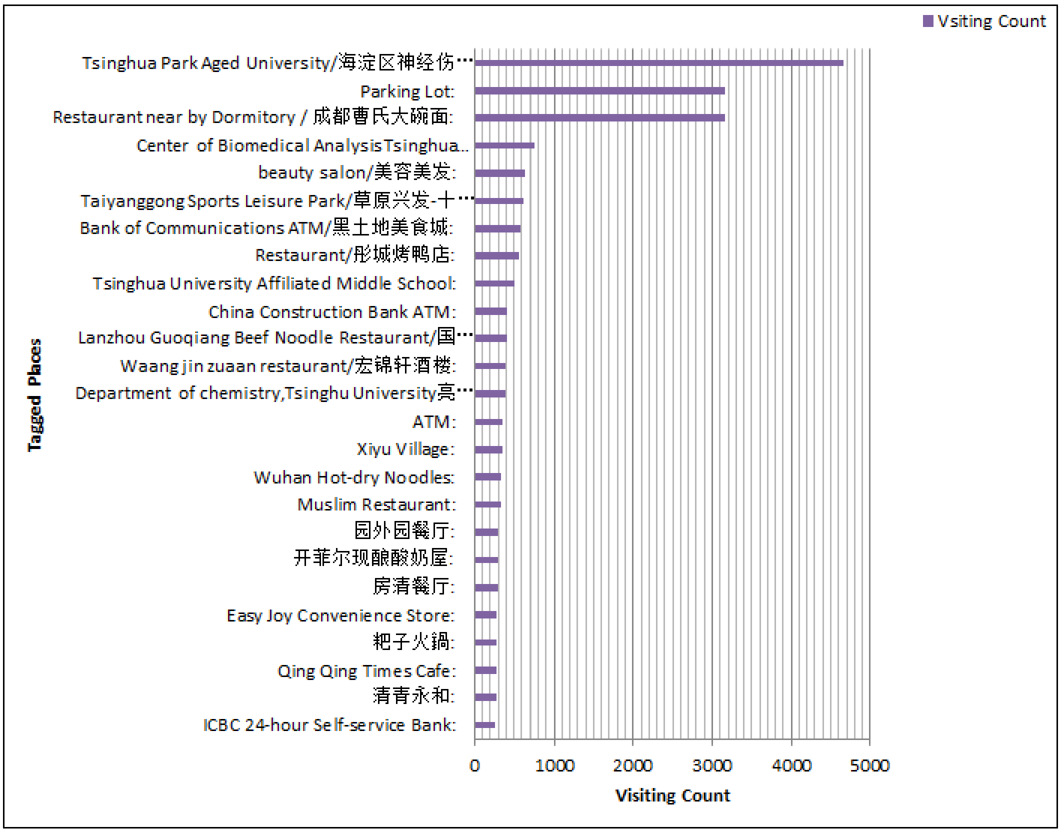


Figure 5.3: Application Level Analysis without considering establishment as POI's

Concept of Semantic ontology is followed in finding the frequent regions. At initial stage the foot prints are calculated for the abstract level geographical domains like university, schools, ATM, restaurants etc. and further the result is processed to get the foot prints for the particular locations like Chongqing Home Cooking restaurant, China Construction Bank, ATM etc. In other sense we can say analysis is done at the application level.

Geographical level analysis results are shown in figure 5.1.For this case frequently visited places are analysed at the abstract level so as to know the interests of the people.

The semantic tagging of the location varies along with the POI's (Points of interests) considered. Likewise in our case if "establishment"is considered as one of the POI's then the results vary a lot in comparison to when it is not. The main difference comes in terms of Dormitory, since dormitory is covered under establishments.

Application level analysis results represent that which all places are visited more frequently when we go at the concrete level, like which if university is visited maximum number of times then actually which university we are talking about here. When we analyse at the lower level we see that some locations have extremely large number of foot prints as compared to other locations.
Figure 5.2 shows the result of application level analysis with taking "establishment"as one of the POI's and figure 5.3 represents the same analysis results without considering "establishment ". Some of the other POI's which are considered are university, schools, restaurants, cafe, hospital, library, food and lodging.

Algorithm 2 Extracting Frequent places

```
Input: Normalized Geo-tagged data set T, Set of Stay Points classified per user
Output: Set of Stay Points with frequency of visits $S_F$[][2]
while i < NoOfUsers , do
    while j < NoOfStaypoints , do
        if StayPoint $S_i$ not in $S_F$ then
            Add $S_i$ in $S_F$
            Add $S_F$[i][1] = 1;
        else
            $S_F$[i][1] = $S_F$[i][1] + 1;
        end if
        j = j + 1
    end while
    i = i + 1
end while
Return $S_F$
```

The algorithm used for extracting frequent places is shown in algorithm 2. This method of finding the locations visited more frequently is effective in the estimation of the population at different places and their interests.

We see that the most frequently visited places are either academic related buildings since the data is related to university or some cafeteria/park ATM etc. located in the university. Basically we process each and every stop point for each and every user and thus calculate the count for every place visited.

These frequent visited places can also be used for the recommended systems to recommend the regions to various dealers on the basis of the number of foot prints of the users. These can be used for market value analysis like how many users are travelling a particular and in mostly what time interval.

5.2 Ranking of locations using HITS analysis

Extraction of stop points is covered in chapter 4 followed by semantic tagging. Next we aim to extract the meaningful locations from these stay points. Also many stay points can be mapped to a single location so instead of considering multiple stop points/ multiple latitude-longitude pair for a particular location, better option is to cluster the similar stop points and get the meaningful location out of this.
Hierarchical clustering is applied for clustering the stay points. Each stay point start with its own cluster and further clusters are merged as we move upwards on to the hierarchy. Clusters are merged such that they may represent same semantic location. For the merging of the clusters a measure of dissimilarity between sets of clusters is required. Since here cluster represent the stay points which are formed by latitude and longitude, the measure of metrics is “Haversine distance”. Haversine formula is an important equation for finding the distance between two points on the sphere from their latitudes and longitudes. Given two latitudes and longitudes (lat1,long1) and (lat2,long2) Haversine distance calculation is given as:
Radius of Earth = 6371000; // Meters
Lat1 = lat1.toRadians();
Lat2 = lat2.toRadians();
latDiff = Lat2 - Lat1;
Long1 = long1.toRadians();
Long2 = long2.toRadians();
longDiff = Long2 - Long1;
a = sin(latDiff/2) * sin(latDiff/2) + cos(Lat1) * cos(Lat2) * sin(longDiff/2) * sin(longDiff/2);
c = 2 * $atan2$ ($\sqrt{a}$, $\sqrt{1-a}$);

d = R * c; // in Meters

For deciding the number of clusters to be formed, at each level of hierarchy reverse geocoding is performed. “Reverse Geocoding”is a process of reverse coding of a location represented by latitude and longitude onto a readable address or place name. The level at which it is found that every cluster corresponds to a same semantic location, is responsible for deciding the no of locations to be formed. A final cluster formed at the top of the hierarchy is expected to represent a single, unique semantic location as shown in figure 5.4.

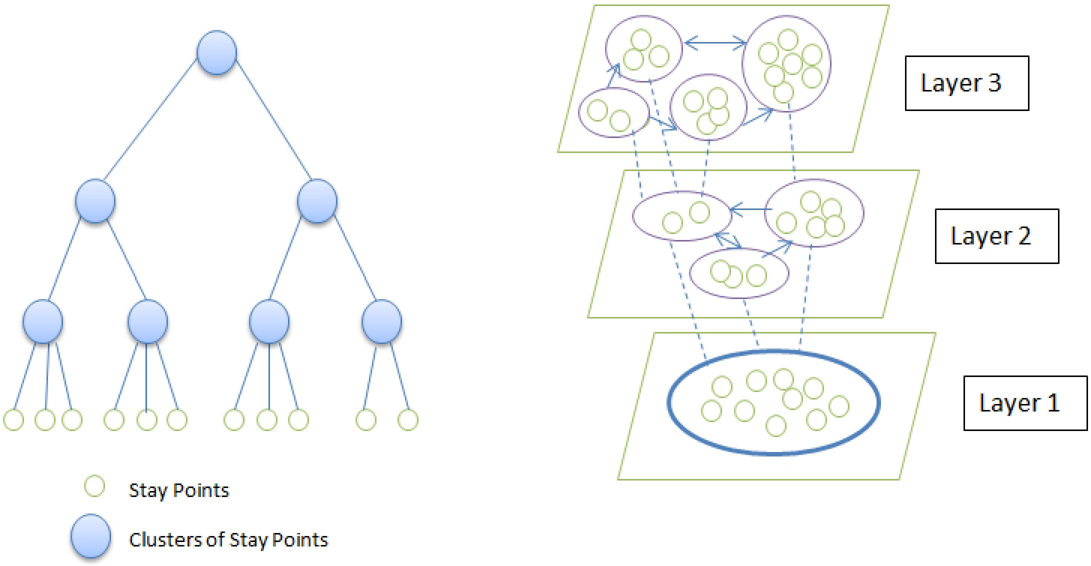
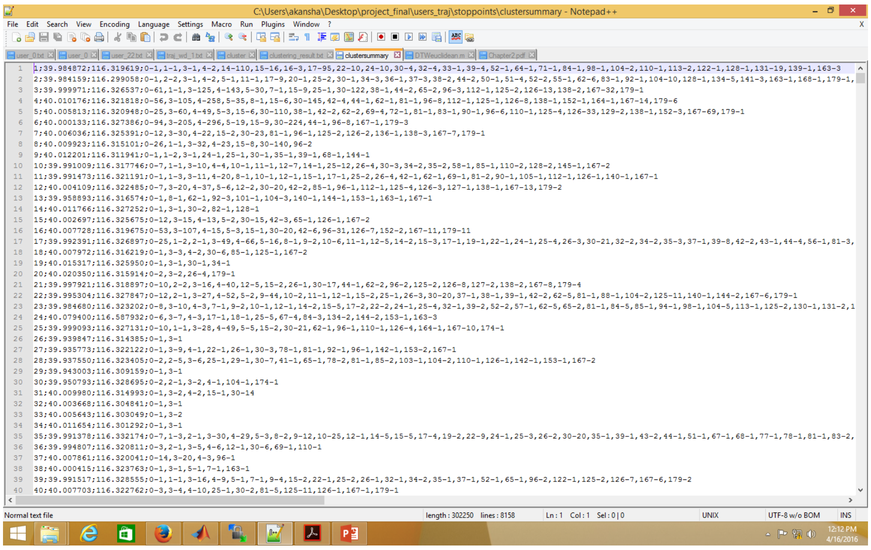


Figure 5.4: Hierarchical Graph Modeling User Location

For the Geolife dataset the number of clusters to be considered for further analysis was 2478. These clusters actually correspond to the meaningful locations. Each cluster is represented by three tuples (CID, Lat-Long, UID-Count) as shown in figure 5.5(snapshot of the data format) here CID is the cluster id, Lat-Long is a pair of latitude and longitude value basically it the mean value of the latitudes and longitudes of all the stop points covered within a cluster, UID Count is a pair of user id and the count value i.e. which all users have visited the particular cluster followed by the number of visits by each user. These four tuples are maintained at each level of the hierarchy.

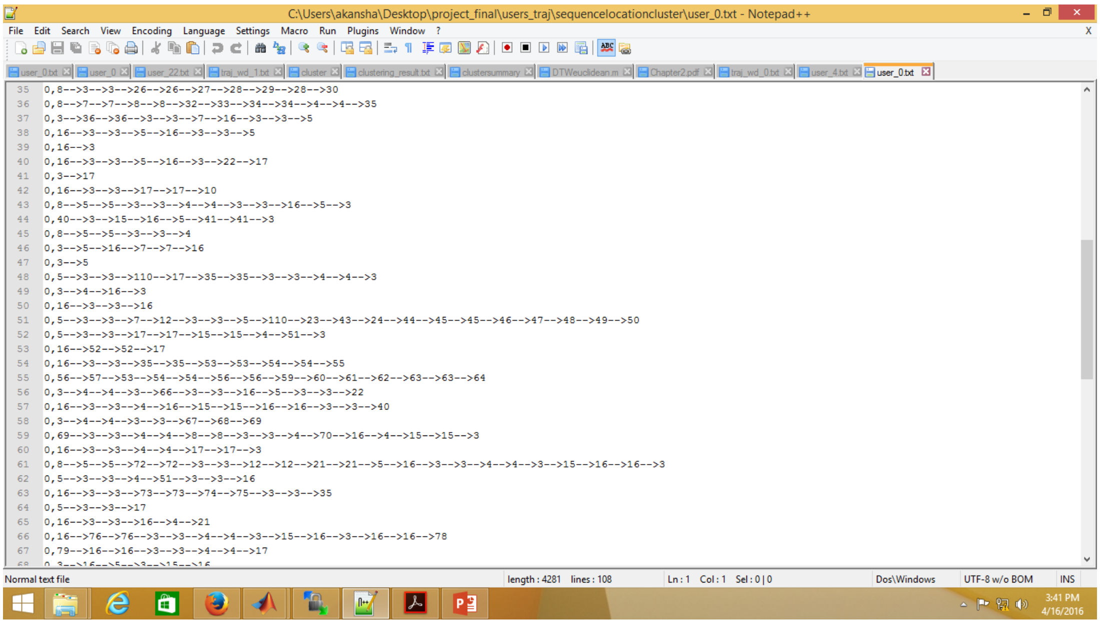

C:\Users\akansha\Desktop\project_final\users_traj\stoppoints\clustersummary - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
user_0.txt
user_0
user_22.txt
traj_wd_1.txt
cluster
clustering_result.txt
clustersummary
DTWeuclidean.m
Chapter2.pdf

```
1;39.984872;116.319619;0-1,1-1,3-1,4-2,14-110,15-16,16-3,17-95,22-10,24-10,30-4,32-4,33-1,39-4,52-1,64-1,71-1,84-1,98-1,104-2,110-1,113-2,122-1,128-1,131-19,139-1,163-3
2;39.984159;116.299058;0-1,2-2,3-1,4-2,5-1,11-1,17-9,20-1,25-2,30-1,34-3,36-1,37-3,38-2,44-2,50-1,51-4,52-2,55-1,62-6,83-1,92-1,104-10,128-1,134-5,141-3,163-1,168-1,179-1,
3;39.999971;116.326537;0-61,1-1,3-125,4-143,5-30,7-1,15-9,25-1,30-122,38-1,44-2,65-2,96-3,112-1,125-2,126-13,138-2,167-32,179-1
4;40.010176;116.321818;0-56,3-105,4-258,5-35,8-1,15-6,30-145,42-4,44-1,62-1,81-1,96-8,112-1,125-1,126-8,138-1,152-1,164-1,167-14,179-6
5;40.005813;116.320948;0-25,3-60,4-49,5-3,15-6,30-110,38-1,42-2,62-2,69-4,72-1,81-1,83-1,90-1,96-6,110-1,125-4,126-33,129-2,138-1,152-3,167-69,179-1
6;40.000133;116.327386;0-94,3-205,4-296,5-19,15-9,30-224,44-1,96-8,167-1,179-3
7;40.006036;116.325391;0-12,3-30,4-22,15-2,30-23,81-1,96-1,125-2,126-2,136-1,138-3,167-7,179-1
8;40.009923;116.315101;0-26,1-1,3-32,4-23,15-8,30-140,96-2
9;40.012201;116.311941;0-1,1-2,3-1,24-1,25-1,30-1,35-1,39-1,68-1,144-1
10;39.991009;116.317746;0-7,1-1,3-10,4-4,10-1,11-1,12-7,14-1,25-12,26-4,30-3,34-2,35-2,58-1,85-1,110-2,128-2,145-1,167-2
11;39.991473;116.321191;0-1,1-3,3-11,4-20,8-1,10-1,12-1,15-1,17-1,25-2,26-4,42-1,62-1,69-1,81-2,90-1,105-1,112-1,126-1,140-1,167-1
12;40.004109;116.322485;0-7,3-20,4-37,5-6,12-2,30-20,42-2,85-1,96-1,112-1,125-4,126-3,127-1,138-1,167-13,179-2
13;39.958893;116.316574;0-1,8-1,62-1,92-3,101-1,104-3,140-1,144-1,153-1,163-1,167-1
14;40.011766;116.327252;0-1,3-1,30-2,82-1,128-1
15;40.002697;116.325675;0-12,3-15,4-13,5-2,30-15,42-3,65-1,126-1,167-2
16;40.007728;116.319675;0-53,3-107,4-15,5-3,15-1,30-20,42-6,96-31,126-7,152-2,167-11,179-11
17;39.992391;116.326897;0-25,1-2,2-1,3-49,4-66,5-16,8-1,9-2,10-6,11-1,12-5,14-2,15-3,17-1,19-1,22-1,24-1,25-4,26-3,30-21,32-2,34-2,35-3,37-1,39-8,42-2,43-1,44-4,56-1,81-3,
18;40.007972;116.316219;0-1,3-3,4-2,30-6,85-1,125-1,167-2
19;40.015317;116.325950;0-1,3-1,30-1,34-1
20;40.020350;116.315914;0-2,3-2,26-4,179-1
21;39.997921;116.318897;0-10,2-2,3-16,4-40,12-5,15-2,26-1,30-17,44-1,62-2,96-2,125-2,126-8,127-2,138-2,167-8,179-4
22;39.995304;116.327847;0-12,2-1,3-27,4-52,5-2,9-44,10-2,11-1,12-1,15-2,25-1,26-3,30-20,37-1,38-1,39-1,42-2,62-5,81-1,88-1,104-2,125-11,140-1,144-2,167-6,179-1
23;39.984680;116.323202;0-8,3-10,4-3,7-1,9-2,10-1,12-1,14-2,15-5,17-2,22-2,24-1,25-4,32-1,39-2,52-2,57-1,62-5,65-2,81-1,84-5,85-1,94-1,98-1,104-5,113-1,125-2,130-1,131-2,1
24;40.079400;116.587932;0-6,3-7,4-3,17-1,18-1,25-5,67-4,84-3,134-2,144-2,153-1,163-3
25;39.999093;116.327131;0-10,1-1,3-28,4-49,5-5,15-2,30-21,62-1,96-1,110-1,126-4,164-1,167-10,174-1
26;39.939847;116.314385;0-1,3-1
27;39.935773;116.322122;0-1,3-9,4-1,22-1,26-1,30-3,78-1,81-1,92-1,96-1,142-1,153-2,167-1
28;39.937550;116.323405;0-2,2-5,3-6,25-1,29-1,30-7,41-1,65-1,78-2,81-1,85-2,103-1,104-2,110-1,126-1,142-1,153-1,167-2
29;39.943003;116.309159;0-1,3-1
30;39.950793;116.328695;0-2,2-1,3-2,4-1,104-1,174-1
31;40.009980;116.314993;0-1,3-2,4-2,15-1,30-14
32;40.003668;116.304841;0-1,3-1
33;40.005643;116.303049;0-1,3-2
34;40.011654;116.301292;0-1,3-1
35;39.991378;116.332174;0-7,1-3,2-1,3-30,4-29,5-3,8-2,9-12,10-25,12-1,14-5,15-5,17-4,19-2,22-9,24-1,25-3,26-2,30-20,35-1,39-1,43-2,44-1,51-1,67-1,68-1,77-1,78-1,81-1,83-2,
36;39.994807;116.320811;0-3,2-1,3-5,4-6,12-1,30-6,69-1,110-1
37;40.007861;116.320041;0-14,3-20,4-3,96-1
38;40.000415;116.323763;0-1,3-1,5-1,7-1,163-1
39;39.991517;116.328555;0-1,1-1,3-16,4-9,5-1,7-1,9-4,15-2,22-1,25-2,26-1,32-1,34-2,35-1,37-1,52-1,65-1,96-2,122-1,125-2,126-7,167-6,179-2
40;40.007703;116.322762;0-3,3-4,4-10,25-1,30-2,81-5,125-11,126-1,167-1,179-1
```

Normal text file length : 302250 lines : 8158 Ln : 1 Col : 1 Sel : 0 | 0 UNIX UTF-8 w/o BOM INS

Figure 5.5: Cluster Representation

Once all the locations are found, next step is to construct the location history. "Location History"is represented as a sequence of stay points/places a user has visited in the geographic space with the corresponding arrival and leaving times. Location history is constructed for each user and hence now there is an idea that which place is visited after/before which place as shown in figure 5.6.


C:\Users\akansha\Desktop\project_final\users_traj\sequencelocationcluster\user_0.txt - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
user_0.txt
user_0
user_22.txt
traj_wd_1.txt
cluster
clustering_result.txt
clustersummary
DTWeuclidean.m
Chapter2.pdf
traj_wd_0.txt
user_4.txt
user_0.txt

```
0,8-->3-->3-->26-->26-->27-->28-->29-->28-->30
0,8-->7-->7-->8-->8-->32-->33-->34-->34-->4-->4-->35
0,3-->36-->36-->3-->3-->7-->16-->3-->3-->5
0,16-->3-->3-->5-->16-->3-->3-->5
0,16-->3
0,16-->3-->3-->5-->16-->3-->22-->17
0,3-->17
0,16-->3-->3-->17-->17-->10
0,8-->5-->5-->3-->3-->4-->4-->3-->3-->16-->5-->3
0,40-->3-->15-->16-->5-->41-->41-->3
0,8-->5-->5-->3-->3-->4
0,3-->5-->16-->7-->7-->16
0,3-->5
0,5-->3-->3-->110-->17-->35-->35-->3-->3-->4-->4-->3
0,3-->4-->16-->3
0,16-->3-->3-->16
0,5-->3-->3-->7-->12-->3-->3-->5-->110-->23-->43-->24-->44-->45-->45-->46-->47-->48-->49-->50
0,5-->3-->3-->17-->17-->15-->15-->4-->51-->3
0,16-->52-->52-->17
0,16-->3-->3-->35-->35-->53-->53-->54-->54-->55
0,56-->57-->53-->54-->54-->56-->56-->59-->60-->61-->62-->63-->63-->64
0,3-->4-->4-->3-->66-->3-->3-->16-->5-->3-->3-->22
0,16-->3-->3-->4-->16-->15-->15-->16-->16-->3-->3-->40
0,3-->4-->4-->3-->3-->67-->68-->69
0,69-->3-->3-->4-->4-->8-->8-->3-->3-->4-->70-->16-->4-->15-->15-->3
0,16-->3-->3-->4-->4-->17-->17-->3
0,8-->5-->5-->72-->72-->3-->3-->12-->12-->21-->21-->5-->16-->3-->3-->4-->4-->3-->15-->16-->16-->3
0,5-->3-->3-->4-->51-->3-->3-->16
0,16-->3-->3-->73-->73-->74-->75-->3-->3-->35
0,5-->3-->3-->17
0,16-->3-->3-->16-->4-->21
0,16-->76-->76-->3-->3-->4-->4-->3-->15-->16-->3-->16-->16-->78
0,79-->16-->16-->3-->3-->4-->4-->17
0,3-->16-->5-->3-->15-->16
```

Normal text file length : 4281 lines : 108 Ln : 1 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS

Figure 5.6: Location History of User 0

We then use location histories to construct a two layer graph which models the connections between various locations, between locations and users and ultimately the connections between users and locations as shown in figure 5.7. One layer in the graph is the user layer while the other is the location layer. In user layer the nodes represent the various users while in location layer the nodes represent the various locations formed from the hierarchical clustering result.

An edge exists between the user and the location if the user has visited the particular location and the edge exists between the locations if these locations are visited by the user in sequence. This location sequence can be checked form the location histories.

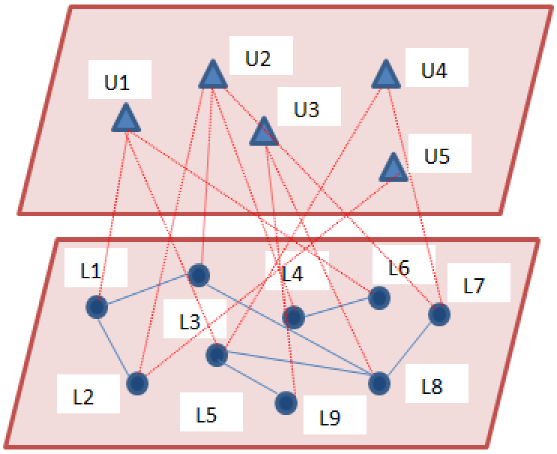


Figure 5.7: Two Layered Graph of Users and Locations

Location-Location edges and User-Location edges are weighted. Location-Location weight captures the number of trips between locations while User-Location weight captures the number of visits by the user to that particular location.

More formally, the two layer graph consists of two sub graphs, a User-Location sub graph and a Location- Location sub graph.

User-Location sub graph is a weighted graph GUL = (U, L, EUL, WUL) where U represent a set of nodes representing users, L represent a set of nodes representing Locations, EUL represent a set of edges that represent user visit to that particular location, WUL represent the weight of the edge describing the number of visits by the user to that location.

Location-Location sub graph is a weighted graph GLL = (L, ELL, WLL) where L represent a set of nodes representing Locations, ELL represent a set of edges between locations that represent the location sequence or the trips followed by the user, WLL represent the weight of the edge describing the number of transitions between locations. Using n users and m locations, an n*m adjacency matrix UL is built for the above mentioned weighted graph GUL .UL= [xij] : 0 < i < n , 0 < j < m; where xij represents how many times user i has visited the location j.

Using m locations, an m*m adjacency matrix L is built for the above mentioned weighted graph GUL .L= [yij] : 0 < i < m , 0 < j < m; where yij represents how many times user has driven between location i and location j.

For the given data set a part of UL and L adjacency matrix is shown below.

$$GUL = \begin{bmatrix} 1 & 1 & 61 & 56 & 25 & 94 & 12 & 26 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 125 & 105 & 60 & 205 & 30 & 32 \\ 2 & 2 & 143 & 258 & 49 & 296 & 22 & 23 \\ 0 & 1 & 30 & 35 & 3 & 19 & 0 & 0 \end{bmatrix}$$

$$GLL = \begin{bmatrix} 320 & 2 & 11 & 0 & 0 & 0 & 0 & 0 \\ 2 & 32 & 2 & 1 & 0 & 0 & 0 & 0 \\ 11 & 2 & 1616 & 621 & 261 & 7 & 58 & 142 \\ 0 & 1 & 621 & 518 & 20 & 5 & 22 & 41 \\ 0 & 0 & 261 & 20 & 300 & 1 & 13 & 95 \\ 0 & 0 & 7 & 5 & 1 & 0 & 0 & 0 \\ 0 & 0 & 58 & 22 & 13 & 0 & 58 & 26 \\ 0 & 0 & 142 & 41 & 95 & 0 & 26 & 128 \end{bmatrix}$$

The authority vector for above matrix comes out to be [0.003842, 0.0007279, 0.5690057, 0.250140, 0.0964342, 0.00273430,0.02178194, 0.05533313,] so the ranking of the locations is [3 , 4 , 5 , 8 , 7, 2 , 1, 6]

The algorithm used for computing authority and hub score is shown in algorithm 3.

Algorithm 3 Calculating Hub and Authority Vector

```
Input: A Graph G and an Adjacency matrix A corresponding to graph G
Output: Hub and Authority vector
n = length(A) //Total number of nodes
Au = dot(transpose(A),A) // dot product of transpose of A and A
Hu = dot(A1,transpose(A1)) // dot product of A and transpose of A
a = ones(n) //Unit matrix of size n
while j < 15, do //defines the no of iterations
    a = dot( Au , a )
    h = dot(h,Hu)
end while
Return a,h
```

5.3 User Similarity and Trajectory Clustering

Formally, clustering the users based on trajectory similarity means classifying or dividing a set of users into various clusters on the basis of their travelling experiences and the path followed by them. For clustering the users two approaches are taken into consideration 1) On the basis of start and end points 2) On the basis of whole trajectories as an atomic unit. Clustering on the basis of start and end points requires to find the start and end points of all the trajectories for all the users which has already been done in the previous section. Now considering the travelling pattern of the users and finding the places visited by them at the geographic level we try to cluster the users.
While clustering on the basis of whole trajectories require comparing all the trajectories of all the users with one another and on the basis of similarity and dissimilarity the users are clustered. The measure considered for finding the similarity between trajectories is DTW (Dynamic Time Warping)

5.3.1 On Basis of Start and End Points

Once we have the start and end points for all the users and also we have the places frequently visited by all the users, now we proceed by calculating the frequency of these places for all the users and try to cluster the users on the basis of the frequencies computed.

For clustering the users on the basis of start and end points, the same hierarchical graph which we got in section 5.2 is used. In Section 5.2 each cluster represents some location and also all the users who have visited that location. So by using this a distance matrix can be built consisting of all the users and the locations. Each entry of the matrix represent the count for which user has visited the location.

Next Normalization of the matrix is done since the total number of places visited by all the users differ. Now Hierarchical clustering is applied onto the matrix so as to seperate the users onto different clusters on the basis of the locations visited by them (Result of Hierarchical clustering as dendrogram shown in fig. 5.8).Once we have got the dendrogram we need to decide the number of clusters so as to cut the dendrogram at the right place.

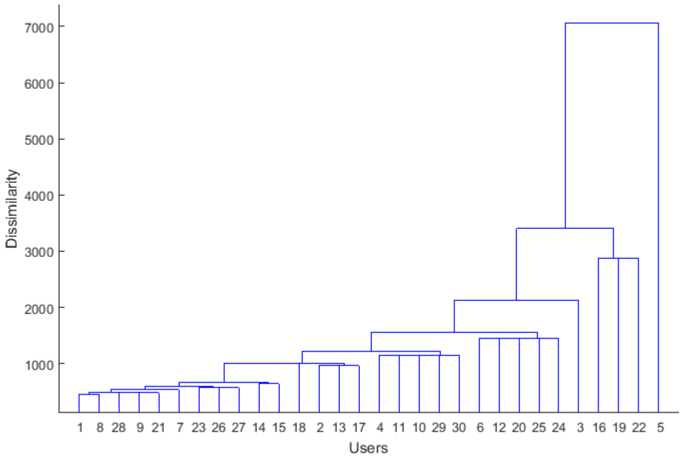


Figure 5.8: Hierarchical Clustering for Start and Stop Locations

For deciding the number of clusters elbow method is applied in which a graph is constructed for SSE (Sum-of-Squared error) vs No-of-Clusters. With each value for number of clusters SSE is computed and is plotted. The graph looks like an arm, and an “elbow”in the arm gives the optimal number of clusters. It is so because at an elbow minimum SSE value is there. Though SSE value decreases as the number of cluster value increases and it tends to 0 when the number of cluster is equal to 1 as then there is no error found, but for the optimum value the minimum SSE value is checked. Silhouette criterion for determining number of clusters is given by:
$S(i) = (b(i) - a(i))/max(b(i), a(i))$
where a(i) = Average dissimilarity of i with all other data within the same cluster
b(i) = Average dissimilarity of i to any other cluster, of which i is not a member

Optimal number of clusters for the above dendrogram shown in figure 5.8 comes out to be 4 as shown in figure 5.9.

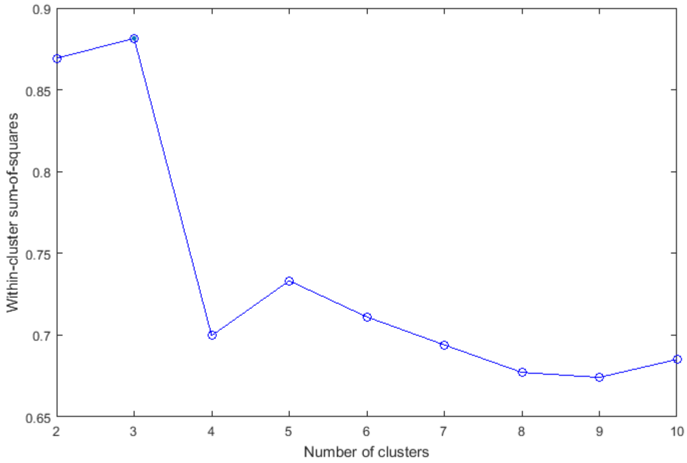


Figure 5.9: Optimal Number of Clusters

Same approach is also applied for the abstract locations visited by the users. From figure 5.1 it is found that restaurants, schools, bank, hospital etc. are most frequently visited places for this data so next we go on finding the frequency of these places for each user and we get the results for two different users user1 and user 2 as shown in figure 5.10 and figure 5.11 resp.

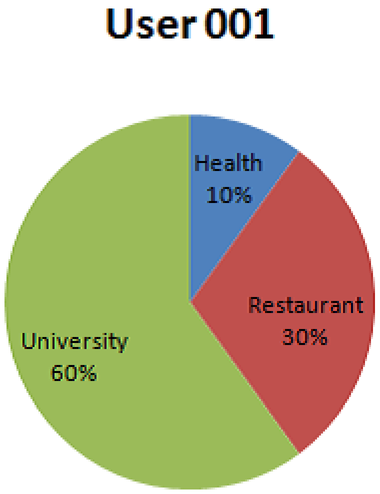


Figure 5.10: User 1 Travelling Experience

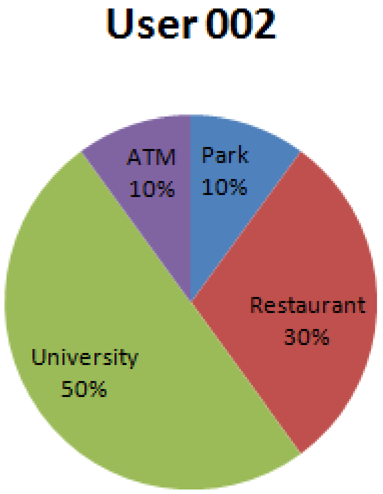


Figure 5.11: User 2 Travelling Experience

So on the basis of these frequencies as a similarity measure further we cluster the users. Also it is assumed that the periodic behaviour of the user depends on the frequently visited regions so by following the same approach as used in finding the frequently visited places we try to find all the places which are visited frequently by the particular user. In other words we can say that we find all the places where the particular user has his/her foot prints. Once we have all the places where the user has foot prints we try to find how frequently that place is visited by the user. Frequent places must depict some pattern of the trajectories followed by the user. As a student in the university the user must be visiting his/her school/university daily and too at fixed period of time so all this information can be used for learning the classifier at the classification stage.

5.3.2 On the Basis of Whole Trajectory as an Atomic Unit

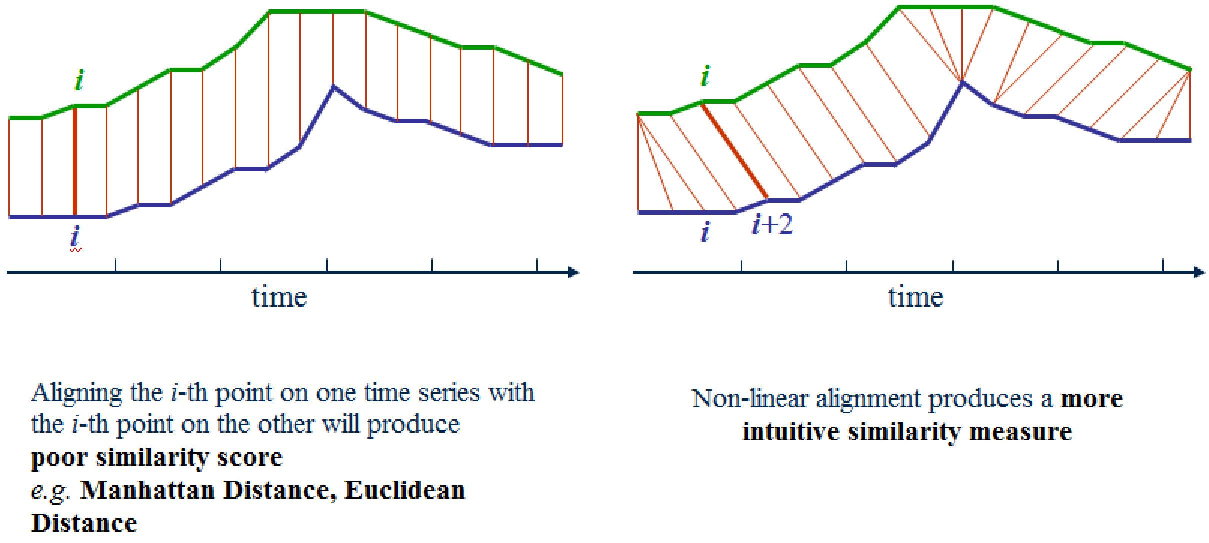
Next we try to cluster the users based on the whole trajectories as one atomic unit. So for this clustering we consider the Dynamic Time Warping (DTW) as Similarity measure.

Any distance whether it is Euclidean, Manhattan or anyone else which aligns the i-th point on one time series with the i-th point on the other will always produce a poor similarity score so we need some non-linear (elastic) alignment e.g. DTW which produces a more spontaneous similarity measure, allowing similar shapes to match even if they are out of phase in the time axis. Distances metrics like Manhattan and Euclidean also compares only the time series which are of same length and these doesn't handle the outliers and noise present in the data.

Dynamic Time Warping (DTW) is a similarity measuring algorithm basically used to find the similarity measure between the two temporal sequences which may be varying in terms of time and speed. Here the data on which we have applied DTW is taken for different users and different days so many different means of travel may be there but we need worry about all this as DTW itself take care of accelerations/decelerations ,change in speed etc. By DTW method even the time series here what is referred to trajectories can be compared, as instead of doing one to one comparison as used in Euclidean distance (shown in figure 5.12) it prefers doing the many-to-one and one-to-many comparison. It recognizes the similar shapes even when signal transformation like shifting or scaling is present.

Given two trajectories T = t1, t2, t3....... tn and S = s1, s2, s3......... sm of length n and m respectively. DTW method considers the distance between every point of one trajectory to every point of other trajectory. Hence a matrix where every entry shows the distance between the points of one trajectory to another.

distanceMat(i,j) refers to the distance from ith point of trajectory T to jth point of tra-



Aligning the *i*-th point on one time series with the *i*-th point on the other will produce **poor similarity score**
e.g. **Manhattan Distance, Euclidean Distance**

Non-linear alignment produces a **more intuitive similarity measure**

Figure 5.12: Comparison of DTW with other similarity measure

jectory S dist(Ti,Sj) where 1<=i<=n and 1<=j<=m.
Now objective of DTW is to find a warping path P = p1,p2,p3,....pi....pk where pi refers to dist(Ti,Sj) and since it is a continuous path max(n,m) < k < m+n-1 such that it minimizes the cost function

$$\text{DTW(T,S)} = \sum_{i=1}^{k} Pi$$

Since DTW considers every single point of one time series to another there is always a chance of exponential paths so to restrict these cases concept of Monotonicity, Continuity, Boundary conditions and Warping window is applied shown in figure 5.13.

Monotonicity guarantees that any of the feature will never be repeated in the alignment i.e. alignment can never go back in the time index. Continuity guarantees that the alignment should not omit important features i.e. alignment path can never jump in the time index. Boundary Conditions guarantee that the alignment does not consider partially one of the sequences while Warping Windows guarantees that the alignment does not try to skip different features and gets stuck at similar features.

The algorithm for DTW is shown below in algorithm 4

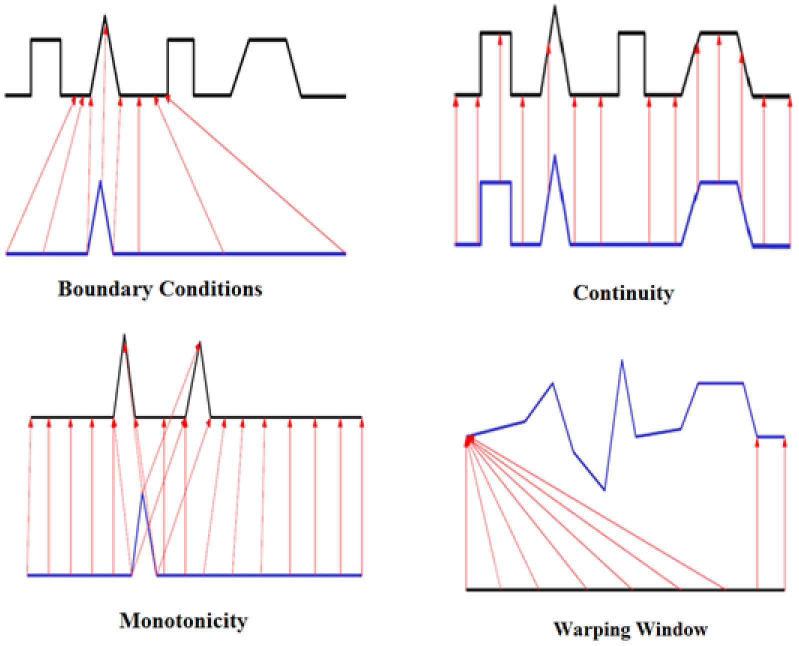


Figure 5.13: DTW Concepts

Algorithm 4 Finding DTW Similarity Measure

```
Input: Two trajectories T and S of length L1 and L2 resp.
Output: Similarity measure between two trajectories
Dist[L1][L2] = 0
Cost[L1][L2] = 0
i = 0, j = 0
while i < L1 , do
    while j < L2 , do
        dist[i][j] = Haversine Distance$(Pi, Pj)$ //Calculating the Haversine distance between two points Pi and Pj where Pi lies on trajectory T and Pj lies on trajectory
        j = j + 1
    end while
    i = i + 1
end while
```

Algorithm 3 Finding DTW Similarity Measure (continued)

```
Cost[0][0] = Dist[0][0]
i = 0,j = 0
while j < L2 , do
    Cost[0][j] = Dist[0][j] + Cost[0][j-1] //Moving in right direction along the first row
end while
while i < L1 , do
    Cost[i][0] = Dist[i][0] + Cost[i-1][0] //Moving in upward direction along the first column
end while
i = 0,j = 0
while i < L1 , do
    while  j < L2 , do
        Cost[i][j] = min(Cost[i-1][j-1] , Cost[i][j-1], Cost[i-1][j]) + Dist[i][j])
        j = j + 1
    end while
    i = i + 1
end while
Path = (L1-1,L2-1)
i = L1 - 1, j = L2 - 1
while i > 0 , do
    while j > 0 , do
        if i = 0 then
            j = j - 1
        else if j = 0 then
            i = i - 1
        else
            if Cost[i-1][j] = min($(Cost[i-1][j], Cost[i-1][j-1], Cost[i][j-1])$ then
                i = i - 1
            else if Cost[i][j-1] = min$(Cost[i][j-1], Cost[i-1][j-1], Cost[i-1][j])$ then
                j = j - 1
            else
                i = i - 1;
                j = j - 1;
```

Algorithm 2 Finding DTW Similarity Measure (continued)

```
            end if
        end if
        Path.append[i][j]
    end while
end while
for i, j in Path do
    accumulatedCost = accumulatedCost + Dist[i][j]
end for
Return Path and accumulatedCost
```

Since for some users it may happen that some of the paths are repeated for eg. a user daily travels from Dormitory to University so this path is repeated frequently, So instead of directly jumping onto finding the similarity between the uers, we first try to reduce the the number of paths per user. For doing this also, DTW is used.

At each stage whenever two trajectories are found similar, these are merged into one and are further represented by the "representative trajectory".

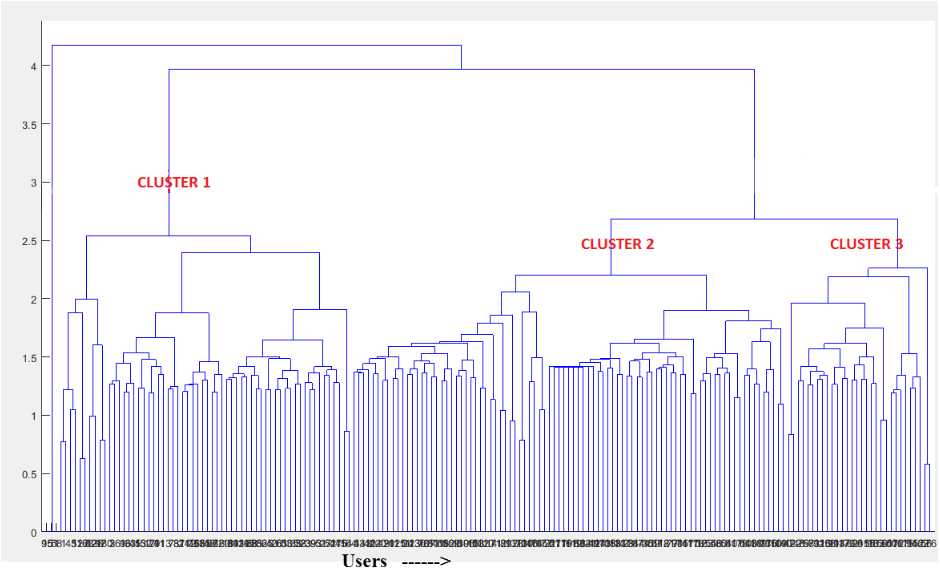


Figure 5.14: Hierarchical Clustering on the basis of DTW

For building the representative trajectory we took into consideration the backtracking part of the DTW algorithm in which the path to be followed in DTW is calculated. Since this path gives us the minimum cost, it means it basically finds the most similar parts of the trajectory so gives best result for the representative trajectory.

Here also to build distance matrix Haversine distance is used and deciding the no of clusters Elbow method is used.
Dendrogram obtained after applying the hierarchical clustering is shown in figure 5.14 and the optimal number of clusters come out to be 2 as shown in figure 5.15

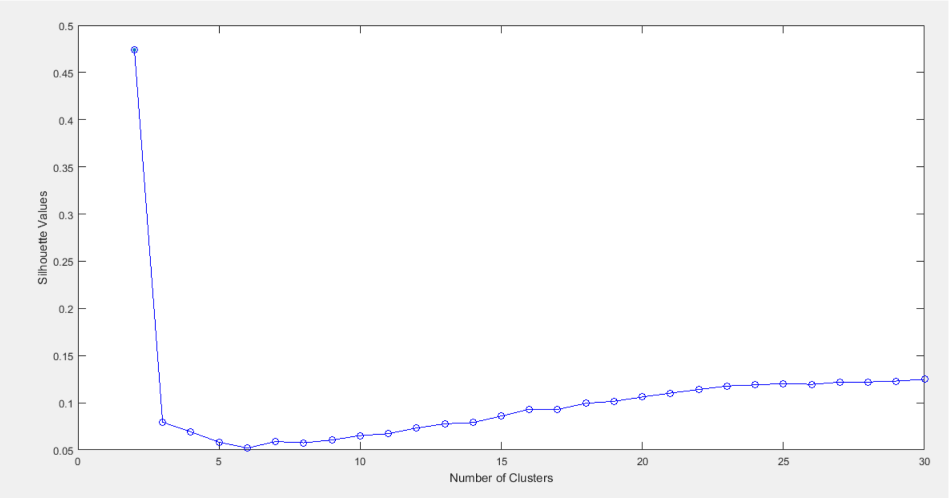


Figure 5.15: Optimal Number of Clusters

Once result is there for all the users, now we apply the same process between the users and try to cluster the users.

Chapter 6

Conclusion and Future Scope

In this paper a framework is proposed in order to cluster the users on the basis of the places they have visited i.e. the start and locations of the trajectories followed by them and also on the basis of the paths followed by the users. The core of work lies in semantic tagging of the trajectories followed by the users and similarity on the basis of semantically tagged trajectories.

The experimental results show that since this data is collected around Beijing university the most of times only university, school and Dormitory is visited so there is a huge difference in the frequency of the visiting these 3 places and other palces. Also for this data set there are very less number of clusters and most of the users fall in the same cluster. The same experiments when performed on some other data set with variety of users may give better results. Further for improving the accuracy in clustering of the users we can add other dimensions like timestamp. Likewise here we have considered only the start and end locations travelled/visited by the users but further we can also take into consideration the time, at which user is visiting these places. Once we have highly accurate clustering results work can be further done to categorize these clusters into different groups i.e. which cluster represent which type of users.

Also scaling needs to be further done for 180 users in all aspects. More will be the number of users higher will be the accuracy.

Work can further be extended by semantically tagging complete trajectories, Since for this we need to buy the API otherwise the number of requests per day for the API are fixed so it becomes difficult to do tagging for whole trajectories. Semantically tagging complete trajectories can also help in improving the results.

Work done in finding frequent places can be further used for building recommendation systems and market value analysis. Categorization work can be used for identifying the outliers especially helpful in places where security is a main concern.

In the future, the work can be extended in the following directions. First is to improve the clustering result by considering location histories and semantically tagged trajectories and second is to categorise the clusters formed by using associative rule mining.

Bibliography

[1] Quannan Li, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu, and Wei-Ying Ma. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, page 34. ACM, 2008.

[2] Eric Hsueh-Chan Lu and Vincent S Tseng. Mining cluster-based mobile sequential patterns in location-based service environments. In *Mobile Data Management: Systems, Services and Middleware, 2009. MDM'09. Tenth International Conference on*, pages 273–278. IEEE, 2009.

[3] Yu Zheng, Lizhu Zhang, Zhengxin Ma, Xing Xie, and Wei-Ying Ma. Recommending friends and locations based on individual location history. *ACM Transactions on the Web (TWEB)*, 5(1):5, 2011.

[4] Jae-Gil Lee, Jiawei Han, and Kyu-Young Whang. Trajectory clustering: a partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 593–604. ACM, 2007.

[5] Luis Otavio Alvares, Vania Bogorny, Bart Kuijpers, JAF de Macelo, B Moelans, and Andrey Tietbohl Palma. Towards semantic trajectory knowledge discovery. *Data Mining and Knowledge Discovery*, 2007.

[6] Vania Bogorny, Bart Kuijpers, and Luis Otavio Alvares. St-dmql: a semantic trajectory data mining query language. *International Journal of Geographical Information Science*, 23(10):1245–1276, 2009.

[7] John Krumm and Eric Horvitz. Predestination: Where do you want to go today? *Computer*, 40(4):105–107, 2007.

[8] Lin Liao, Donald J Patterson, Dieter Fox, and Henry Kautz. Building personal maps from gps data. *Annals of the New York Academy of Sciences*, 1093(1):249–265, 2006.

[9] Lin Liao, Donald J Patterson, Dieter Fox, and Henry Kautz. Learning and inferring transportation routines. *Artificial Intelligence*, 171(5):311–331, 2007.

[10] Donald J Patterson, Lin Liao, Dieter Fox, and Henry Kautz. Inferring high-level behavior from low-level sensors. In *UbiComp 2003: Ubiquitous Computing*, pages 73–89. Springer, 2003.

[11] Fosca Giannotti, Mirco Nanni, Fabio Pinelli, and Dino Pedreschi. Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 330–339. ACM, 2007.

[12] John Krumm and Eric Horvitz. Predestination: Inferring destinations from partial trajectories. In *UbiComp 2006: Ubiquitous Computing*, pages 243–260. Springer, 2006.

[13] Yu Zheng, Like Liu, Longhao Wang, and Xing Xie. Learning transportation mode from raw gps data for geographic applications on the web. In *Proceedings of the 17th international conference on World Wide Web*, pages 247–256. ACM, 2008.

[14] Yu Zheng. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):29, 2015.

[15] Brendan Morris and Mohan Trivedi. Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 312–319. IEEE, 2009.

[16] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th international conference on World wide web*, pages 791–800. ACM, 2009.

[17] Lei Gong, Hitomi Sato, Toshiyuki Yamamoto, Tomio Miwa, and Takayuki Morikawa. Identification of activity stop locations in gps trajectories by density-based clustering method combined with support vector machines. *Journal of Modern Transportation*, 23(3):202–213, 2015.