KINETIK, Vol. 3, No. 4, November 2018, Pp. 305-310
ISSN : 2503-2259
E-ISSN : 2503-2267

DBscan Algorithm and Decision Tree to Automate Trip Purpose Detection

Lailatul Hidayah[*1], Catur Wulandari[2]
[1,2]Universitas Internasional Semen Indonesia
lailatul.hidayah@uisi.ac.id[*1], catur.wulandari@uisi.ac.id[2]

Abstract

One of transportation research topic is detecting trip purpose. Given a collection of GPS mobility records, researchers endeavored to infer useful information such as trip, travel mode, and trip purpose. Obtaining these attributes will help researcher in transportation modelling. This work proposed an approach in defining a trip or a trip segmentation which is a part of trip purpose problem as well as inferring the trip purpose. By Utilizing Dbscan clustering algorithm, decision tree, and some useful features, we are able to detect the trips and their purposes as well as building the model to automate the trip derivation.

Keywords: *Trip Purpose, Dbscan Algorithm, Decision Tree, Transportation Modelling, Travel Behavior*

1. Introduction

Trip purpose detection is one from many other research topics in transportation where researchers aim to extract meaningful information from GPS data such as travel mode, trip detection, travel behavior, etc. This information is analyzed and then utilized in many applications such as traffic analysis in transportation system planning [1]. In the future, it can also be helpful in designing land-use distribution, predicting travel demand [2] that can lead to solving traffic congestion, etc. Researcher can take advantage of the information of certain area usage combined with traffic demand from time to time to design a new transportation link for a better traffic management.

Various methods have been conducted to collect transport data to be analyzed. Traditional survey methods include face-to-face interview using paper and pencil, mail-out/ mail-back survey [3], and computer-assisted survey which consists of three types namely computer-assisted telephone interview (CATI), computer-assisted personal interview, and computer-assisted self interview(CASI)[4]. These approaches have some disadvantages such as travel time overestimating [5], misreporting [3], confusion of trip purpose [6], and non-response [7].

Overcoming these limitations, Global Positioning System (GPS) has been exploited in many countries for transport research [8]. Utilizing GPS, survey's respondents are given logging device or smartphone to track their mobility. This leads to significant reduction in users' active participation[9]. Hence besides accuracy gain, more data can be collected since more participants are eager to join the survey. Tenacious work in conventional survey has inhibited participation in studies [10].

Although GPS data provides more accurate information, there are several attributes that can't be extracted directly and need further post processing such as travel mode, trip purpose, and start and end of trip. Machine learning methods and hybrid methods have been utilized for computation in GPS data post processing to extract travel mode [8]. Similarly, several recent studies have investigated methods to automatically derive trip purpose. Wolf built a model to automatically derive trip purpose with the help of a database of land use [2]. Griffin and Huang utilized Decision Tree to train the GPS data for automatic trip purpose derivation [10]. Wang and Fu et.al. detected trip purpose by means of mixture of Hawkess processes [9]. In this study, we propose another method to derive trip purpose from GPS data.

2. Research Method

Each record in our data consists of timestamp when it is recorded, latitude, and longitude. We call each record as point afterwards. There are four main steps that have been applied in this work, namely filtering, clustering, trip reconstruction, and trip detection. The idea is to group

Hidayah, L., & Wulandari, C. (2018). DBscan Algorithm and Decision Tree to Automate Trip Purpose Detection. Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 3(4). doi:http://dx.doi.org/10.22219/kinetik.v3i4.672
Receive July 31, 2018; Revise August 03, 2018; Accepted September 20, 2018

points into several clusters and detect connection between those clusters from the GPS record data which will be defined as a trip. In order to make the clustering results reliable, we need to filter the raw data beforehand.

2.1 Filtering

The data that we have contain locations information recorded every 10 minutes in average. It tells us movement of the observed user, regardless whether the user is actually commuting to certain location of interest or staying in their current area.

If the user is commuting to a certain location, the points will form a line when we plot it in Google Earth while it will be a rectangle cluster of adjacent points if the user is actually staying in his area (e.g. office, home, town square, etc.)

In this step we want to distinguish these two types of points and keep only those in which the user is not travelling. We want to remove the other points since it will not be relevant for the next step (e.g. Clustering). We called these points as trip points.

We define a point to be a trip point if at the current timestamp the user's speed is more than a certain value. Intuitively, we want to remove points where the user is speeding.

2.2 Clustering

Having the points filtered, in this step we cluster the remaining points based on their geodistance. Borrowing the idea of Griffin[10], Dbscan algorithm is used [11]. Dbscan algorithm is a clustering algorithm for arbitrary shape clusters. Since in this case the volunteers' movements inside the houses and industrial buildings can form any shape, dbscan is chosen. Points in each user's record are clustered individually. Hence the resulting cluster is specific to a particular user. This approach could be further modified considering all points for all observed users in order to have more general cluster such as residential area, work area, instead of personal houses and office buildings.

After the clustering process is completed, we label the clusters using a set of rules. We use three labels namely home, work, and other. First rule is that a work area will have stay time less than equal eight hours in a day while the second rule is that home cluster will be the densest cluster. The second rule is motivated by an assumption that people tend to stay longer at their house compare to other places and meanwhile they don't move a lot around the house. Hence the home cluster will be very dense.

2.3 Trip Reconstruction

With several clusters formed, we consult the original data (unfiltered data) to reconstruct trips based on its timestamp order. Intuitively, this process should be done individually for each user in order not to confuse it with other users' sequence of points. If there is n number of clusters found, then there will be as many as 2 permutation of n types of trip purposes. Figure 1 summarizes the described procedures.

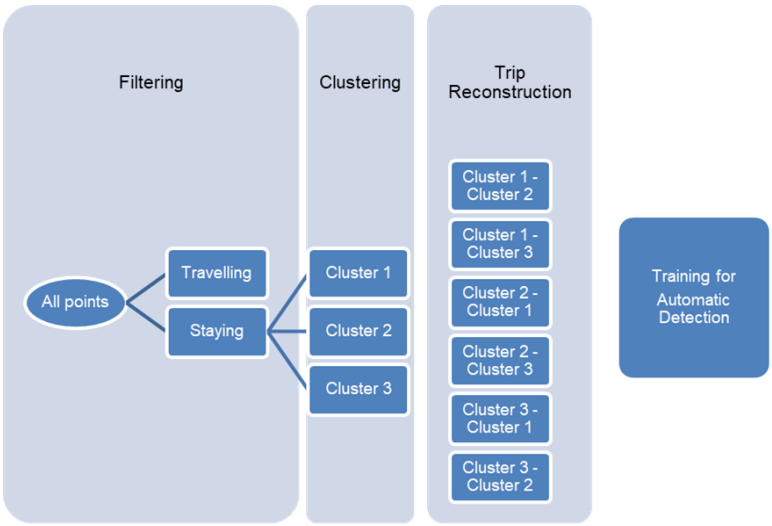

Figure 1. Trip Purpose Detection Workflow

2.4 Trip Detection

The purpose of this step is to build a model for an automatic categorization of trip purpose. The result from the previous step will be used as input data. Decision tree will be utilized for the training. For further information on decision tree used in this work, it is recommended to study [12]. A set of features is defined beforehand, as follow:

1. Duration of stay in the area
2. Start time
3. Home location (average values from Home cluster)
4. Distance of start point to Home cluster
5. Distance of end point to Home cluster
6. Distance of start point to Work cluster
7. Distance of end point to Work cluster

3. Result and Discussion

The GPS data for this work is obtained from a total of 123 volunteers who have carried the GPS logger everywhere they move. Duration of successfully recorded movement for each user varies from 1 day to 3 months. The data which are in .txt format are

3.1 Filtering

In this stage we removed points where the user is moving fast or with vehicle as we aim to keep only points where the user is not travelling. The speed is calculated from division of distances of two concurrent GPS point by difference of the timestamp.

Implemented in Java [13], in addition to the filtering procedure, a specific function is created to generate the result in CSV format. These CSV files were then converted into KML format by using an online csv-to-kml converter to be able to view it in Google Earth [14]. There are many online csv-to-kml converters such as [15]. Google Earth was preferred for its ability to read input from files as well as many other useful features which will potentially be useful for next step of the research. Table 1 shows number of points before and after the filtering process.

Table 1. Percentages of Filtered Points

User ID	Before	After	Percentage
78	7237	5311	0.733868
49	3386	1538	0.454223
109	6529	4780	0.732118
118	5849	4473	0.764746
88	4655	3218	0.6913

From the five random sampled user, it is shown that the percentage of filtered points over all initial points vary from 45 to 76 percent. Figure 2 and Figure 3 illustrates the filtering result for one of the users.

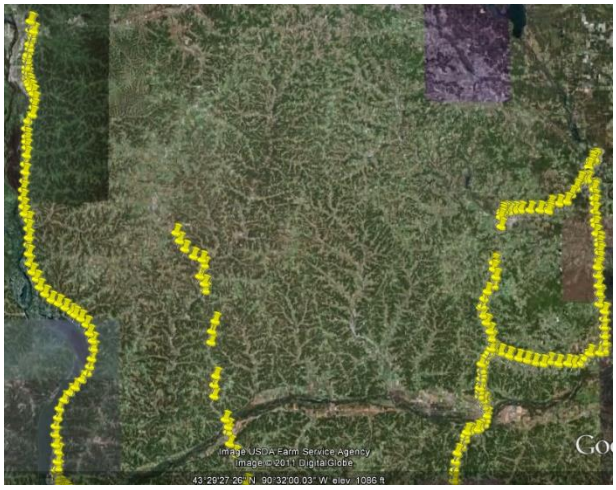

Figure 2. Google Earth View of User Movements Before Filtering

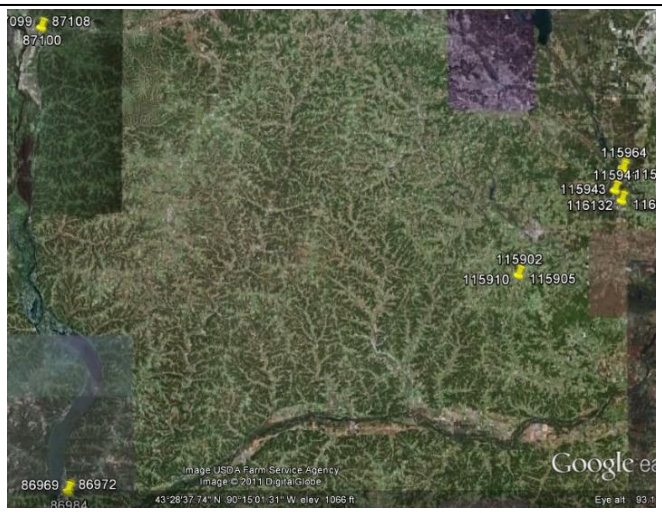

Figure 3. Filtered User Movements Points

3.2 Clustering

Dbscan algorithm is implemented for the clustering stage. MinEps= 0.0015 and MinPts= 10 are applied by default for all users. MinEps is minimum distance or radius for the cluster while MinPts is the minimum number of points that need to be present in the radius to deem it as cluster. Several values of Eps have been applied before. This variation will produce different number of groups or clusters. If we set it too large, it will tend to produce large size cluster. This might cause problem in the labelling step. We may lose some important information as one cluster will be labelled as one single type of cluster only. Thus, we have to make sure that we don't merge two or more area types in one cluster. This clustering and the following stages were implemented in Matlab [16].

Figure 4 depicts the difference obtained by two different values of MinEps. The red cluster in the centre of the left figure is broken down into several smaller clusters, shown by the figure next to it. Next step is to label the resulted clusters. Using the given rules, clusters are identified as either home, work, or other.

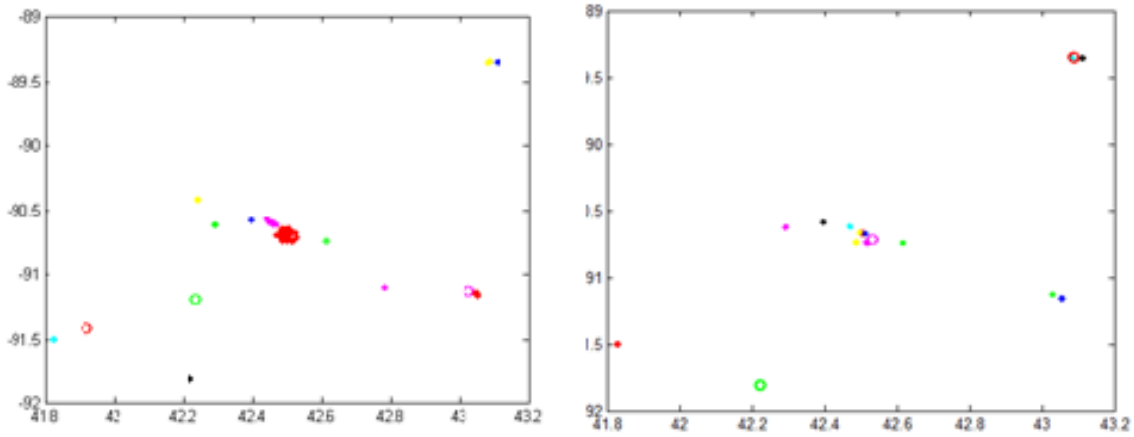

Figure 4. Results of Difference MinEps Value

3.3 Trip Reconstruction

Given several clusters with their labels obtained from previous step, we can construct trips by consulting the original data before we filter it. Having each point in filtered data labelled, we apply the following algorithm to construct the trip.

The function GetPoints will return all points in original data between these two labelled points (there might be some points in between the two points that have been removed by the filtering). Figure 5 shows one example of a reconstructed trip. Line 1 to 5 is a trip of type 3 which is a trip from work to home. Given this information, it is trivial to infer the trip purpose.

```
For i ← 1 to number of each point in filtered data
If i=1
        Skip
Else
        If label[i] != label[i-1]
                GetPoints(start,end)
```

```
1308689507679.000000 42.506898 -90.668105 21-Jun-2011 15:51:47
1308690112430.000000 42.483404 -90.661430 21-Jun-2011 16:01:52
1308690176069.000000 42.471038 -90.668207 21-Jun-2011 16:02:56
1308690238760.000000 42.460857 -90.674142 21-Jun-2011 16:03:58
1308690302377.000000 42.453340 -90.677023 21-Jun-2011 16:05:02
===============
1308745418364.000000 42.453945 -90.674927 22-Jun-2011 07:23:38
1308746023149.000000 42.498214 -90.698156 22-Jun-2011 07:33:43
===============
1308746087599.000000 42.496871 -90.697782 22-Jun-2011 07:34:47
1308746212206.000000 42.501300 -90.697925 22-Jun-2011 07:36:52
1308746276799.000000 42.498796 -90.688157 22-Jun-2011 07:37:56
1308746340482.000000 42.504287 -90.675969 22-Jun-2011 07:39:00
1308746404971.000000 42.502955 -90.668498 22-Jun-2011 07:40:04
1308746467615.000000 42.505352 -90.670214 22-Jun-2011 07:41:07
===============
1308776983536.000000 42.502950 -90.666803 22-Jun-2011 16:09:43
1308777046245.000000 42.494867 -90.662067 22-Jun-2011 16:10:46
1308777109884.000000 42.484949 -90.661874 22-Jun-2011 16:11:49
1308777174554.000000 42.471912 -90.668176 22-Jun-2011 16:12:54
1308777237115.000000 42.459460 -90.671871 22-Jun-2011 16:13:57
===============
1308777301781.000000 42.458752 -90.671206 22-Jun-2011 16:15:01
1308777364408.000000 42.454034 -90.677425 22-Jun-2011 16:16:04
===============
1308779028158.000000 42.451989 -90.674260 22-Jun-2011 16:43:48
1308779652235.000000 42.405611 -90.588873 22-Jun-2011 16:54:12
```

Figure 5. Reconstructed Trips Shown as Sets of Points

Using three types of labels namely work, school, and other, we will have 6 kinds of trips as follow:

1. Home-Work
2. Home-Other
3. Work-Home
4. Work-Other
5. Other-Home
6. Other-Work

3.4 Training

Done with the complete procedures of trip purpose detection, we now aim for a model to automate it. We treat the obtained label and the trip as our training data for a classification algorithm. Here decision tree is used with the help of a function called classregtree from Matlab. Considering the sample size and computational power, k=3 was chosen in the k-fold cross validation. A test performed using 3-fold cross validation was conducted by a self-implemented function where the accuracy was also calculated by computing the percentage of the true prediction compared with the number of tested points. It yielded an accuracy of 75%.

4. Conclusion

A step by step method has been described in section 3.1 to 3.4. This is another approach for automatic derivation of trip purpose from GPS record data. Utilizing the power of Dbscan clustering and some useful features as well as a simple algorithm, we can construct a trip and infer the trip purpose.

There is a large space to be filled for improvement. In the current work, a specific cluster for each user is built. It is very interesting to know how the result will be if we generate the cluster

generally, for all users. Hence, we might also have a general model for the automate trip detection given the cluster is generally defined.

Furthermore, instead of DbScan that can only incorporate spatial feature, we might want to apply other algorithm that will be able to incorporate more features besides spatial features.

In terms of the automate trip detection, we can try to use different classification algorithm. Lastly, since here the training data comes from a processed trip, the next challenge would be how to build model that would allow us to use raw data and automatically generate the trip purpose.

Acknowledgement

This work is supervised by Profesor Xiangliang Zhang from King Abdullah University of Science and Technology (KAUST), Kingdom of Saudi Arabia.

References

[1] B. Wang, L. Gao, and Z. Juan, *"A Trip Detection Model for Individual Smartphone-Based GPS Records with a Novel Evaluation Method,"* Advances in Mechanical Engineering, Vol. 9, No. 6, 2017.

[2] J. Wolf, R. Guensler, and W. Bachman, *"Elimination of the Travel Diary: Experiment to Derive Trip Purpose from Global Positioning System Travel Data,"* in Transportation Research Record: Journal of the Transportation Research Board, Vol. 1768, Pp. 125–134, 2001.

[3] J. Wolf, *"Using GPS Data Loggers To Replace Travel Diaries In the Collection of Travel Data by,"* Analysis, No. July, Pp. 225, 2000.

[4] P. Stopher, C. FitzGerald, and J. Zhang, "*Search for a Global Positioning System Device to Measure Person Travel,"* Transportation Research Part C: Emerging Technologies, Vol. 16, No. 3, Pp. 350–369, 2008.

[5] P. R. Stopher, *"Use of an Activity-Based Diary to Collect Household Travel Data,"* Transportation (Amst)., Vol. 19, No. 2, Pp. 159–176, 1992.

[6] P. McGowen and M. McNally, *"Evaluating the Potential To Predict Activity Types from GPS and GIS Data,"* in Transportation Research Board 86th Annual Meeting, 2007.

[7] M. Zimowski, R. Tourangeau, R. Ghadialy, and S. Pedlow, *"Nonresponse in Household Travel Surveys,"* Report to the Federal Highway Administration, 1997.

[8] L. Wu, B. Yang, and P. Jing, *"Travel Mode Detection Based on GPS Raw Data Collected by Smartphones: A Systematic Review of the Existing Methodologies,"* 2016.

[9] P. Wang, Y. Fu, G. Liu, W. Hu, and C. Aggarwal, *"Human Mobility Synchronization and Trip Purpose Detection with Mixture of Hawkes Processes,"* Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '17, Pp. 495–503, 2017.

[10] T. Griffin and P. O. B. Denton, *"A Decision Tree Classification Model to Automate Trip Purpose Derivation,"* Proceedings of CAINE, Pp. 44–49, 2005.

[11] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, *"A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,"* in Kdd, Vol. 96, No. 34, Pp. 226–231, 1996.

[12] L. Breiman, *"Classification and Regression Tree,"* Pp. 1–33, 2001.

[13] R. S. Newton, *"What Is Java,"* Technology Collection in Microstation Manager, 1998.

[14] Dep of State Geographer, *"Google Earth,"* 2017.

[15] "CSV To KML Converter," 2018.

[16] M. MathWorks, *"What is Matlab,"* 2017.