TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 18, No. 1, February 2020, pp. 124~132
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v18i1.13006

A developed GPS trajectories data management system for predicting tourists' POI

Rula Amjad Hamid[1], Muayad Sadik Croock[2]
[1]College of Business Informatics, University of Information Technology and Communications, Baghdad, Iraq
[2]Computer Engineering Department, University of Technology, Iraq

Article Info

Article history:

Received Apr 28, 2019
Revised Jun 25, 2019
Accepted Jul 12, 2019

Keywords:

IoT
Point of interest
Stay points
Tourism
Trajectories

ABSTRACT

One of the areas that have challenges in the use of internet of things (IoT) is the field of tourism and travel. The issue here is how to employ this technology to serve the tourism and managing the produced data. This work is focus on the use of tourists' trajectories that are collected from global positioning system (GPS) mobile sensors as a source of information. The aim of work is to predict preferred tourism places for tourists by tracking tourists' behavior to extract the tourism places that have been visited by such tourists. Density based clustering algorithm is mainly used to extract stay points and point of interest (POI). By projecting GPS location (for user and places) on the Google map, the type and name of places favored by the tourists are determined. K nearest neighbor (KNN) algorithm with haversine distance has been adopted to find the nearest places for tourists. The evaluation of the obtained results shows superior and satisfactory performance that can reach the objective behind this work.

Corresponding Author:

Rula Amjad Hamid,
University of Information Technology and Communications, Iraq,
College of Business Informatics Baghdad, Iraq.
Email: eng_rula_amjed@uoitc.edu.iq

1. INTRODUCTION

It is frequently thought that the local authorities and tourism agencies have a adequate understanding of tourist's preferences, needs, and how local people's interests can be integrated in tourism planning [1]. A chief challenge in managing tourism system using IoT is how to track user behaviors and preference acquisition [2]. There is a need to know the details information of precise locations visited by tourists, the attracted locations by tourist, personal reflections on tourists' experiences and future travel behavioral intentions [3].

Many studies employed tourist GPS trajectories to classify and forecast the behaviors of tourists that visit locations by collecting their movements, choices and needs. Trajectory is a location sequence (spatial–temporal) with travel times. The relation among the sequences depends on the neighborhood function and the time tolerance [4]. The study in [5] designed dataflow mining structure for user's mobile behavior trajectory depend on place services in mobile. The aim of the study of [5] was to get user path data that incorporates place information and social information. Another study in [6] proposed a heuristic method that combines dynamic time warping and the earth mover's distance, to accurately measure the similarity of tourist trajectories. The study of [7] expanded the application of tourist movements in the mobile Internet era, in which movement data (using GPS trajectories) could be collected more easily. This was done by proposing a method that improves prediction accuracy and trade-off between prediction accuracy and efficiency.

The study in [8] proposed method using tied ranking with ordinal logistic regression to predict the factors that can affect the tourism sector in Iraq and what are the factors influencing this field in order to focus on the development of this industry. The work in [9] extracted tourism POI from a vast quantity of applicant POIs based on tourist preferences. The research in [10] presented the use of location check-ins, which are available on mobile social media platforms, as an extra data supply to revise tourist behaviors. In general, most existing approaches are not capable to tackle the challenge in integrated and widespread manner [11].

The aim of this work is to extract GPS points from trajectories data, analyze behavior patterns of tourist paths, and predict the preferred places for tourists. This is done using the google map information to determine the favorite places by tourist using clustering algorithms. These algorithms depend mainly on the density information and the POI for each tourist. The collected information is used for building the dataset for system prediction.

2. DATA DESCRIPTION

The proposed system uses GeoLife Trajectories dataset. This GPS dataset was composed in (Microsoft Research Asia) by 182 users in a period of over five years (2007-2012). A trajectory of this dataset is denoted by points sequences. Each one has the information of (latitude, longitude, and altitude). The GeoLife dataset was collected by user mobile devices over a time period of five years. It represents users' movements history like going to work, returning to home, and all kinds of activities in the day life of underlying users [12]. In order to test the proposed algorithm over different dataset, GPS points for a group of Iraqi tourists are used to extract their behavior during their visit to tourism places in the city of Erbil. This city was chosen because it is considered the most important Iraqi provinces in terms of the diversity of tourism areas.

3. PROPOSED SYSTEM

As mentioned earlier, this work produces a tourist prediction system based on the POI of tourists using different trajectory datasets. For easing the reading flow of this paper, the proposed system can be explained according to the applied steps as follows:

3.1. Data cleaning (preprocessing)

The first step is the analysis and preprocessing of the dataset to remove possible noise from the data. Data cleaning is a technique to detect and either remove or correct inconsistencies or missing data in a dataset [13]. Such inconsistent data may affect the results of the study. Noise in data may be caused by many different reasons, such as error in electronic devices (e.g. GPS loggers), software error or human mistake. After cleaning, the data must be consistent with the other similar data in the system. For example, it appears that there are points on the path, conflicting from the pattern of the path. At this point, the person suddenly takes a very high speed, for example more than 200 km/s, in less than 5 seconds, can be removed for invalidity. To remove this type of noise, a solution based on individual velocity is adopted along the path. This is done by calculate the velocity taken from the individual of each point on the tracks, and then checking whether the speed of the individual hesitates to a high value between any two points.

3.2. Feature extraction

The next step is finding and extracting new features from the dataset. The new extracted features are stay points and POI.

3.2.1. Extracting stay points

Stay points: are geographic areas where the individual has spent a long time in their surroundings center point. Stay Point are extracted and grouped from user points based on the time and distance, taken on a route to a geographic area. Stay points can be detected automatically from a user's GPS trajectory by seeking the spatial region where the user spent a period exceeding a certain threshold [14]. In this section, the stay points are detected from users' mobility paths by seeking the spatial region where the user stayed for while. The algorithm that has been proposed in [15] was adopted in order to extract stay points as shown in Figure 1. In the proposed system, if the tourist spent more than 35 minutes within a distance of 200 meters, the point is detected as stay point. In other words, a cluster is detected and allocated. The extracted stay point information contains mean coordinates, arrival time (S.arvT) and leaving time (S.levT) for each tourist, individually. At the other point, the threshold is selected based on the average time, computed from [16] and [17]. The first one takes time threshold = 20 minutes, while the other specified the optimal time in

between 10 to 60 minutes. Haversine Distance (HD) formula, presented in [12], is used in this research to calculate the distance between two points on a sphere. The formula is given by the following equation:

$$\text{Distance}= 2\,R\sin^{-1}\left(\sqrt{sin^2\left(\frac{\emptyset i-\emptyset j}{2}\right)+cos(\emptyset i)cos(\emptyset j)sin^2\left(\frac{\varphi i-\varphi j}{2}\right)}\right) \quad (1)$$

where (R) represents earth radius, ∅ and φ are correspondingly the latitudes and longitudes of points (i,j), respectively.

Algorithm StayPoint_Detection(*P*, *distThreh*, *timeThreh*)

Input: A GPS log *P*, a distance threshold *distThreh*
and time span threshold *timeThreh*
Output: A set of stay points *SP*={*S*}
1. *i*=0, *pointNum* = |P|; //the number of GPS points in a GPS logs
2. **while** *i* < *pointNum* **do**,
3. *j*:=*i*+1;
4. **while** *j* < *pointNum* ***do***,
5. *dist*=Distance(p_i, p_j); //calculate the distance between two points
6. **if** *dist* > *distThreh* **then**
7. $\Delta T=p_j.T-p_i.T$; //calculate the time span between two points
8. **if** ΔT>*timeThreh* **then**
9. *S.coord*=ComputMeanCoord({p_k | *i*<=*k*<=*j*})
10 *S.arvT*= $p_i.T$; *S.levT*=$p_j.T$;
11. *SP*.insert(*S*);
12. *i*:=*j*; break;
13. *j*:=*j*+1;
14. **return** *SP*.

Figure 1. Stay point detection algorithm

3.2.2. Extract point of interest

POI is an important venue/location in the physical world, such as a shopping mall or a theatre, lake. Generally, POI belongs to one or more categories like education, entertainment, arts, food and dining, government, health & beauty, home & family, shopping, sports, and nature [18]. After counting stay points in the previous phase, we should now be able to discover locations where people spend a lot of time frequently in their surroundings. To find such places, the following steps are applied:

- Interesting points are collected using the density-based clustering algorithm to find groups containing at least k points within them. These clusters represent the regions that are frequently visited. Therefore, very likely to be region of interest.
- Each region is represented by center point, which is a point of interest. These locations can be a restaurant, a shopping center, a university building, or a tourist attraction.

DBSCAN algorithm of [19] is adopted in this work. This algorithm composes a group of points and clusters together as well as the points that are packed strongly within a given threshold distance in space and marks points as outliers that lie alone in low density regions. DBSCAN requires two parameters; the first one is epsilon, which is the maximum distance between two samples to be considered in the same neighborhood while the second one is the minimum number of points, required to form a dense region. To estimate these two parameter, the authors of [20] proposed a heuristic to determine them with regards to the "thinnest" cluster in the database. In their experiment, the authors indicate that the optimal value of for k > 4. Thus, in this work, we set k = Minimum Points in cluster = 4.

3.3. Find nearest places/prediction

The finding of the nearest tourism places for the tourist is the next step in proposed system. This is to ease the prediction of the recommended places for the tourist that can satisfy his/her requests. The evaluation of nearest places is performed using KNN method. KNN search is one of the most fundamental problems, which has been extensively studied in various fields of computer science, such as data mining, information retrieval, and spatial databases [16]. Using the user GPS position and POIs, the KNN query can find the closest POI from that tourist (smallest distance from the tourist). In spatial databases, the KNN query can be used in finding the nearest POI, such as a restaurant to a tourist's current location. In this work, Geopandas is used as an open source geospatial data processing method as an application in Python language [21]. As a result, the system is now able to predict the names of the tourist's favorite

places through the preferred type of tourism places (previously extracted) and the nearest tourism places form him/her.

Figure 2 illustrates the working steps of the proposed system for predicting the recommended tourism places for interesting tourists using the information of GPS and POI for them. It is clearly shown that the importance of evaluating the POI for tourists in interesting area to predict the tourism place can be attended. The trajectory data for each user is collected. This data is cleaned up to remove any possible noise. This is to extract the POI for them, individually. The real Geolocation of the underlying users (tourists) is collected from their smart phones to evaluate the nearest distance between them and POI, which recommended as tourism places.

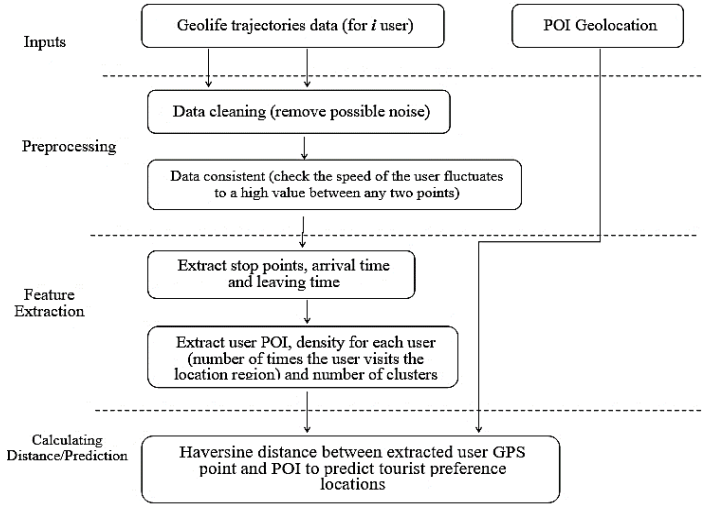


Figure 2. Flow diagram of proposed system

4. RESULTS

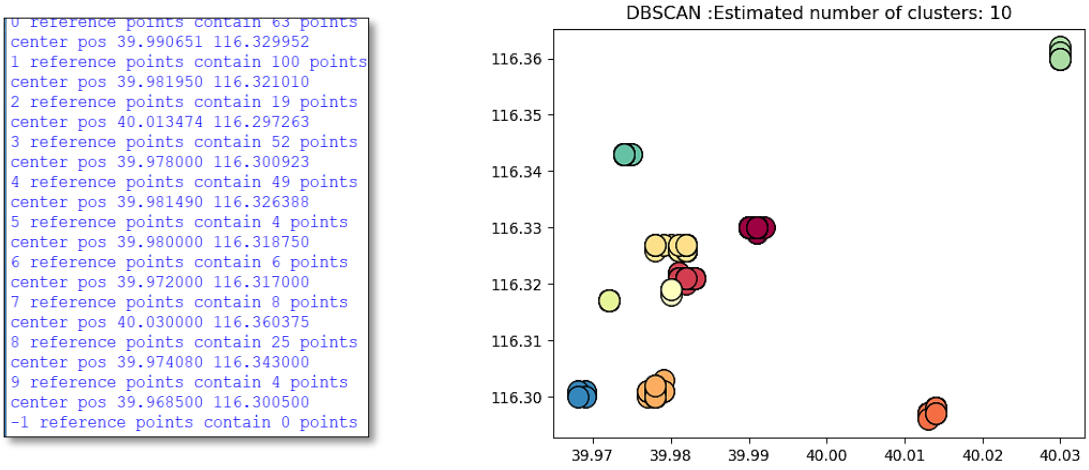
After applying reprocessing operation on the considered datasets, stop points algorithm was applied on refined users' trajectories. The number of extracted stop points is equal to 13320 for 181 users (tourists). Figure 3 shows part of evaluated stop points after mining all the trajectories with extracted feature (user id, longitude, latitude of point, arriving time, leaving time and total spending time). Table 1 represents the users with their clustered points of POI that are detected by DBSCAN algorithm. Figure 4 shows the execution of the clustering algorithm for one tourist using Sklearn.cluster in python 3 [22] with charts that shows Number of points in each cluster with center position of cluster and estimated number of clustered points of interest and the noise points.

H	G	F	E	D	C	B	A	
Total spending time		leaving time		arriving time	longitude	laltitude	user_id	1
01:05:42	04:08:07	23/10/2008	03:02:25	23/10/2008	116.299	39.984	0	2
05:10:03	09:42:25	23/10/2008	04:32:22	23/10/2008	116.324	40.000	0	3
00:44:55	01:23:21	28/10/2008	00:38:26	28/10/2008	116.297	40.011	0	4
00:58:45	02:22:11	28/10/2008	01:23:26	28/10/2008	116.297	40.009	0	5
02:06:46	05:03:02	28/10/2008	02:56:16	28/10/2008	116.324	40.000	0	6
01:07:45	01:15:08	04/11/2008	00:07:23	04/11/2008	116.297	40.011	0	7
01:41:25	02:56:38	04/11/2008	01:15:13	04/11/2008	116.297	40.008	0	8
01:39:15	03:31:07	10/11/2008	01:51:52	10/11/2008	116.332	39.975	0	9
00:41:20	00:58:24	11/11/2008	00:17:04	11/11/2008	116.297	40.011	0	10
00:35:32	03:32:57	12/11/2008	02:57:25	12/11/2008	116.324	40.001	0	11
00:41:44	04:21:37	12/11/2008	03:39:53	12/11/2008	116.322	40.009	0	12
03:04:33	06:50:41	13/11/2008	03:46:08	13/11/2008	116.321	40.008	0	13
00:44:15	07:36:14	13/11/2008	06:51:59	13/11/2008	116.317	39.995	0	14
02:55:42	14:01:21	13/11/2008	11:05:39	13/11/2008	116.324	40.000	0	15
01:52:45	03:52:15	14/11/2008	01:59:30	14/11/2008	116.324	40.000	0	16
00:45:04	04:46:19	14/11/2008	04:01:15	14/11/2008	116.321	40.009	0	17
02:18:39	07:05:03	14/11/2008	04:46:24	14/11/2008	116.321	40.008	0	18
00:46:28	11:17:54	14/11/2008	10:31:26	14/11/2008	116.321	40.009	0	19
04:57:16	16:16:00	14/11/2008	11:18:44	14/11/2008	116.322	40.009	0	20
02:22:27	03:30:45	15/11/2008	01:08:18	15/11/2008	116.324	40.000	0	21
01:00:16	10:54:16	18/11/2008	09:54:00	18/11/2008	116.322	40.010	0	22
01:50:26	13:23:06	19/11/2008	11:32:40	19/11/2008	116.321	40.010	0	23
02:57:45	04:52:09	22/11/2008	01:54:24	22/11/2008	116.324	40.001	0	24
00:46:39	05:38:53	22/11/2008	04:52:14	22/11/2008	116.323	40.009	0	25
03:31:38	10:51:29	02/12/2008	07:19:51	02/12/2008	116.325	40.002	0	26

Figure 3. Extracted stay points

Table 1. Number of stop points and number of clusters/POI

User id	No of trajectories	No of stop points	Number of clusters/ POI
0	171	186	5
1	71	39	4
2	175	135	5
3	322	553	13
4	395	587	13
5	86	63	4
12	77	66	4
13	144	89	7
17	391	361	16
22	146	238	11
23	34	42	6
24	101	77	10
30	296	514	11
35	74	330	10
38	110	163	6
39	227	158	11
42	150	36	5
52	104	98	9
84	215	100	8
92	157	56	4
104	115	67	5
119	45	75	5
126	263	85	7
144	610	83	10
163	809	145	17
167	385	197	9



(a) (b)

Figure 4. Extracted POI for one user (a) Number of points in each cluster with center position of Cluster/POI (b) Estimated number of clusters

The extracted POI of users are projected on dynamic map (from Google) with geonames of POI for Beijing as shown in Figure 5. Beijing POI were collected from website in [23]. Figure 6 shows the POI for each tourist and the type of each place. By Applying KNN the distances between tourist GPS points and Beijing POI are calculated to find K nearest interested places from tourist to finally predict the preferred places. The following assumptions are imposed when calculating a distance for region of interest:

– Circular region of interest: the center of region represents POI
– The Radiuses of regions are different from each other. Assuming lake area to be different in size compared with mall.

To increase the validity of the proposed system an essential need for further test. A number of donor tourists are selected to extract their GPS stop points within Erbil city (one of the most famous cities in Iraq with its varied tourist places). In order to predict their preferred tourism type, the distances of these points from famous tourist places (can be considered as ROI) are calculated. A database is created for the most

well-known tourism locations in Iraq with their positions (latitude, longitude) and the type of interest for each place as shown in Figure 7. For example, the distance using KNN is calculated by assuming the radius from the center of Erbil Castle = 200 meters while Sersank Resort = 500 meters and Family Mall 50 meters. The results are shown in Figure 8. Table 2 represents the stop points of two users with their distances from the center point of region.

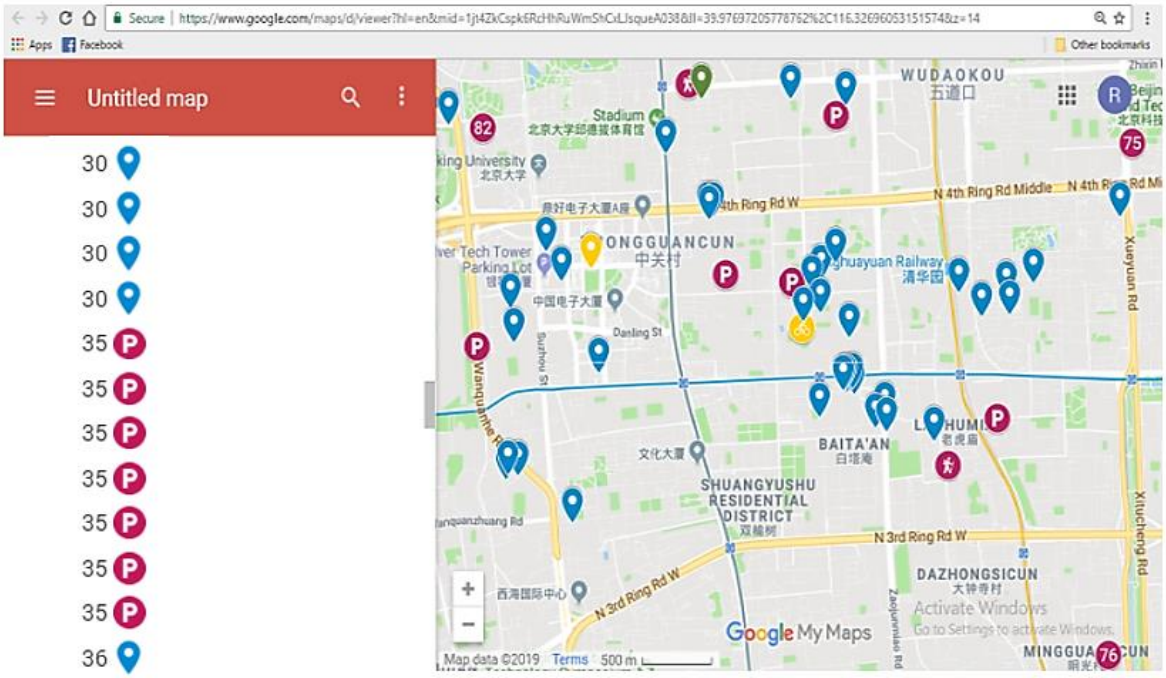


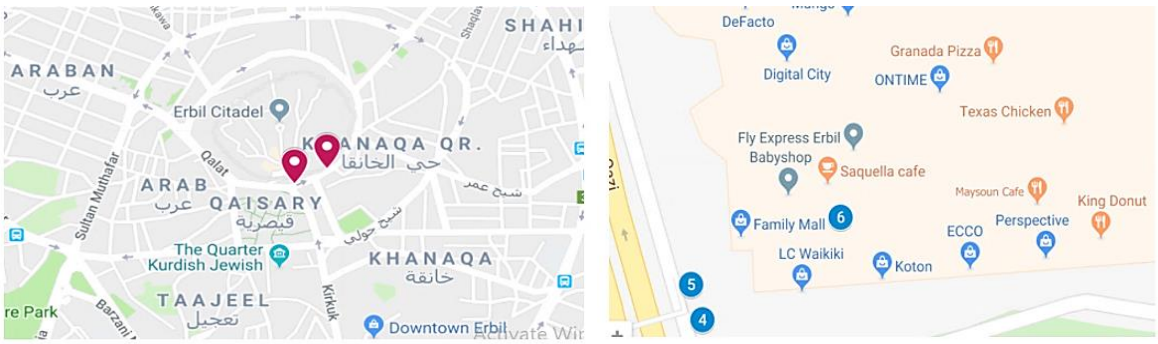
Figure 5. Dynamic google map shown Beijing POI and user POI

U_id	Long	Lat	Type of POI(feature class)
0	116.3239	40.00338	college
0	116.318	39.99243	railroad station
0	116.297	40.00975	populated place
0	116.2978	39.college(	nature
0	116.339	39.9715	college
1	116.3272	39.97896	college
1	116.3064	40.01397	nature
1	116.3natur	39.9982	college
1	116.3101	39.98217	populated place
2	116.3374	39.9263	hotel
2	116.3858	39.90019	nature(lake)
2	110.7722	37.41686	populated place
2	116.209	39.897	university
2	116.323	39.93717	nature
3	116.3191	39.9915	railroad station
3	116.36colle	39.90822	nature(lake)
3	116.3228	39.93589	hotel
5	116.3212	40.01106	nature
5	116.327	40.001	college
6	116.3398	39.98083	college
7	116.3437	39.98068	college
7	116.105	40.08867	populated place
8	116.3287	39.98157	nature
8	116.356	39.958	populated place
9	116.3285	39.99497	college

Figure 6. POI types

	City	Place names	Latitude	Longitude	1. Adventure	Culture	Environmental	Health	Nature	Religious	sport	shopping	business
57	Mosul	The forests of Mosul	36.378563	43.12003	1		1		1		1		
58	Dohuk	Amadiyah Castle	37.09085417	43.48397982		1	1		1				
59	Dohuk	The Abbasid Bridge	37.090582	43.483951		1			1				
60	Dohuk	Great Duhok Mosque.	36.85778	42.999204		1				1			
61	Dohuk	The Great Mosque in Aqra	36.861742	42.997208		1				1			
62	Dohuk	Resort Zawya	36.90466	43.135918	1	1	1		1				
63	Dohuk	Shranche waterfall	37.232618	42.846165	1		1		1				
64	Dohuk	Sarsink	37.04633	43.338697	1		1		1				
65	Dohuk	Screaming	37.022839	43.29227	1		1		1				
66	Dohuk	Solav	36.861459	42.996873	1		1		1				
67	Dohuk	Prazani Park	36.84568	43.004286					1		1		
68	Dohuk	Park Azadi	36.845834	42.99124			1		1		1		
69	Dohuk	Phin Resort	36.860233	42.952941	1		1		1				
70	Dohuk	Duhok Mall	36.866609	42.957144								1	1
71	Dohuk	Dohuk Governorate	36.875833	43.003611	1		1		1				
72	Dohuk	Kelly Sheeran	36.867905	42.948857	1		1		1				
73	Dohuk	Family Mall	36.852067	42.88363								1	1
74	Arbil	The total waterfall on you	36.631242	44.446435	1		1		1				
75	Arbil	Behalal Falls	36.617353	44.497719	1		1		1				
76	Arbil	Shaklawa Resort	36.409844	44.320179	1		1		1			1	
77	Arbil	Chandelar Cave	36.831651	44.221044	1	1	1		1				
78	Arbil	Resort Haj Omran	36.673205	45.048166	1		1		1				
79	Arbil	Museum of Syriac Heritage	36.234235	43.988721		1							
80	Arbil	Ganarok Resort	36.190811	44.022689	1		1		1				

Figure 7. Iraqi POI with types of interest



User 0 User1

Figure 8. Dynamic google map of tourists GPS points

Table 2. Clustering performance measure

User id	Silhouette Coefficient	User id	Silhouette Coefficient
0	0.165	36	0.3
1	0.364	38	0.4
2	0.318	52	0.12
3	0.03	68	0.5
5	0.3	84	0.1
6	0.114	85	0.1
7	0.415	92	0.2
9	0.638	104	0.1
12	0.351	112	0.037
13	0.332	119	0.6
14	0.4	126	0.083
15	0.382	157	0.274
17	0.419	159	0.724
18	0.04	165	0.484
23	0.2	167	0.228
24	0.1	22	0.458
30	0.6	179	0.087
35	0.9		

5. CLUSTER PERFORMANCE MEASURE

In this work Silhouette Index [24] is applied to measure clustering performance. This measure is adopted based on its accuracy, popularity and simplicity of implementation. Silhouette index gives an idea about the samples similarity with other samples within the same cluster (cohesion) and dissimilarity with other samples in other clusters (separation). It ranges (from −1 to +1), where the higher value means it is within cluster similarity and the lower value means it is the intra-cluster similarity [24]. Table 2 shows the Silhouette of each user after applying DBSCAN clustering on Geolife dataset. Most of coefficients have the value (between 0.1 and 0.9), which represents perfect clustering and other less than zero which represents worst clustering.

6. EVALUATION

In this work the experimental results are evaluated using precision, recall as:

$$Precision = \frac{number\ of\ correct\ stops\ found}{number\ of\ stops\ found}$$

$$Recall = \frac{\text{number of correct stops found}}{\text{number of correct stops}} \quad (2)$$

the recall value of DBSCAN method is about (0.591489), while the precision is about (0.371658). As a bench mark, these values are compared with the results in [25], where the best recall value was about (0.36). This means DBSCAN just discovered (36%) of the correct stops with precision at (0.5). Therefore, the proposed work in this paper outperforms the previous work results in term of recall.

7. CONCLUSION

A tourism places prediction and recommendation was proposed. In this work, DBSCAN and nearest neighbor were used to extract and predict the types of tourism places preferred by the tourists using the visited places by them. Such information was used in the tourism recommendation systems or by tourism agencies to know places of tourist attractions. Clustering algorithm was evaluated using Silhouette Coefficient. The system evaluation showed the outperformance of the proposed system over previous ones in term of recall. The system could be expanded using these results in tourist recommendation systems to provide suggestions to the tourist depending on the type of places he/she prefers.

REFERENCES

[1] S. Kantola, M. Uusitalo, V. Nivala, and S. Tuulentie, "Tourism resort users' participation in planning : Testing the public participation geographic information system method in Levi , Finnish Lapland," *Tour. Manag. Perspect.*, vol. 27, pp. 22–32, 2018.
[2] F. Piccialli and A. Chianese, "The internet of things supporting context-aware computing: a cultural heritage case study," *Mob. Networks Appl.*, vol. 22, pp. 332–343, 2017.
[3] S. J. Miah, H. Q. Vu, J. Gammack, and M. McGrath, "A Big Data Analytics Method for Tourist Behaviour Analysis," *Inf. Manag.*, vol. 54, no. 6, pp. 771–785, 2017.
[4] E. Ospina, F. Moreno, and I. A. Uribe, "Using criteria reconstruction for low-sampling trajectories as a tool for analytics," *Procedia Comput. Sci.*, vol. 51, pp. 366–373, 2015.
[5] J. Zhang, T. Wu, and Z. Fan, "Research on Precision Marketing Model of Tourism Industry Based on User's Mobile Behavior Trajectory," *Mob. Inf. Syst.*, vol. 2019, pp. 1-14, 2019.
[6] W. Zheng *et al.*, "Understanding the tourist mobility using GPS: How similar are the tourists?," *Tour. Manag.*, vol. 71, pp. 54–66, 2019.
[7] Z. Zhu, L. Shou, and K. Chen, "Get into the spirit of a location by mining user-generated travelogues," *Neurocomputing*, vol. 204, pp. 61–69, 2016.
[8] R. Amjad and M. S. Croock, "Dominated destinations of tourist inside iraq using personal information and frequency of travel," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 17, no. 4, pp. 1723-1730, 2019.
[9] C. K. Ke, M. Y. Wu, W. C. Ho, S. C. Lai, and L. T. Huang, "Intelligent Point-of-Interest Recommendation for Tourism Planning via Density-based Clustering and Genetic Algorithm," *PACIS 2018 Proceedings*, 2018.
[10] J. M. Luo, H. Q. Vu, G. Li, and R. Law, "Tourist behavior analysis in gaming destinations based on venue check-in data," *J. Travel Tour. Mark.*, vol. 36, no. 1, pp. 107–118, 2019.
[11] J. Li, L. Xu, L. Tang, S. Wang, and L. Li, "Big data in tourism research: A literature review," *Tour. Manag.*, vol. 68, pp. 301–323, 2018.

[12] Y. Zheng, H. Fu, X. Xie, W.-Y. Ma, and Q. Li, "Geolife GPS trajectory dataset-user guide," *Microsoft Res.*, [Online], Available online https//www. microsoft. com/enus/research/publication/geolife-gps-trajectory-dataset-user-guide, 2011.
[13] V. Ganti and A. Das Sarma, "Data cleaning: A practical perspective," *Synth. Lect. Data Manag.*, vol. 5, no. 3, pp. 1–85, 2013.
[14] S. Phithakkitnukoon, T. Horanont, A. Witayangkurn, R. Siri, Y. Sekimoto, and R. Shibasaki, "Understanding tourist behavior using large-scale mobile sensing approach: A case study of mobile phone users in Japan," *Pervasive Mob. Comput.*, vol. 18, pp. 18–39, 2015.
[15] T. Mun Heng, "Mining User Similarity Based on Location History," *Econ. Plan. Ind. Policy Glob. Econ. Concepts, Exp. Prospect.*, no. c, pp. 29–42, 2015.
[16] D. A. Peixoto, "Mining Trajectory Data," pp. 1–23, 2013.
[17] Y. Zheng, L. Zhang, X. Xie, and W. Y. Ma, "Mining interesting locations and travel sequences from GPS trajectories," *Proc. 18th Int. Conf. World wide web - WWW '09*, 2009.
[18] W. Luan, G. Liu, C. Jiang, and L. Qi, "Partition-based collaborative tensor factorization for POI recommendation," *IEEE/CAA J. Autom. Sin.*, vol. 4, no. 3, pp. 437–446, 2017.
[19] M. Ester, H. P. Kriegel, J. Sander, X. Xu, and others, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *KDD-96 Proceedings*, vol. 96, no. 34, pp. 226–231, 1996.
[20] J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications," *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 169–194, 1998.
[21] K. Jordahl, "GeoPandas: Python tools for geographic data," [Online], Available: https//github. com/geopandas/geopandas, 2014.
[22] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
[23] Rebele, Thomas, Fabian Suchanek, Johannes Hoffart, Joanna Biega, Erdal Kuzey, and Gerhard Weikum. "YAGO: A multilingual knowledge base from wikipedia, wordnet, and geonames." *International Semantic Web Conference*, pp. 177-185, 2016.
[24] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987.
[25] T. K. Dang, N. Thoai, and others, "Hybrid stop discovery in trajectory records," *2013 24th International Workshop on Database and Expert Systems Applications*, 2013.