ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-4/W4, 2017
4th International GeoAdvances Workshop, 14–15 October 2017, Safranbolu, Karabuk, Turkey

FAST SS-ILM: A COMPUTATIONALLY EFFICIENT ALGORITHM TO DISCOVER SOCIALLY IMPORTANT LOCATIONS

A.S. Dokuz [a], M. Celik [b]
[a] Nigde Omer Halisdemir University, Computer Engineering Department 51245 Nigde, Turkey - adokuz@ohu.edu.tr
[b] Erciyes University, Computer Engineering Department 38039 Kayseri, Turkey - mcelik@erciyes.edu.tr

KEY WORDS: spatial social media mining, social important locations mining, data mining, Twitter

ABSTRACT:

Socially important locations are places which are frequently visited by social media users in their social media lifetime. Discovering socially important locations provide several valuable information about user behaviours on social media networking sites. However, discovering socially important locations are challenging due to data volume and dimensions, spatial and temporal calculations, location sparseness in social media datasets, and inefficiency of current algorithms. In the literature, several studies are conducted to discover important locations, however, the proposed approaches do not work in computationally efficient manner. In this study, we propose Fast SS-ILM algorithm by modifying the algorithm of SS-ILM to mine socially important locations efficiently. Experimental results show that proposed Fast SS-ILM algorithm decreases execution time of socially important locations discovery process up to 20%.

1. INTRODUCTION

Socially important locations are places which are frequently visited by social media users in their social media lifetime (Dokuz and Celik, 2017). Socially important locations mining aims to discover important locations of social media users using their spatial social media histories. Discovering socially important locations reveal many information about spatial preferences of social media users, such as, which locations are important for a social media user, which locations are co-occurring among users, which user periodically visits a location, and which locations are common for a social media user group, etc.

Discovering socially important locations from social media datasets is challenging due to data volume and dimensions, high amount of spatial and temporal calculations, location sparseness in social media datasets, and computational complexity of current algorithms.

In the literature, several methods and algorithms are proposed to discover important locations. However, these studies have several limitations. Many of the studies used GPS or other data sources, and thus the methods and algorithms are based on these data sources. In addition, most of the studies do not aim to decrease the computational complexity of the algorithms.

Social media datasets are very sparse (Bao et al., 2015). In these dataset, some places are visited frequently and, in contrast, many places are visited rarely (e.g., only once). For example, Figure 1 shows the visited locations of a social media user in her/his social media history. Some statistical information about the user is also given in Table 1. As can be seen in the figure, the visited locations of the user are distributed among several cities. Many of the user's visit locations are in Istanbul, Turkey. However, the locations inside Istanbul are also distributed among different regions. As can be seen in the table, the user visited 95 distinct locations, and 65 of these locations are visited only once. For this user approximately 65% of the locations are visited only once. Because of this reason, these locations cannot be socially important locations of the user and they should be eliminated before socially important locations discovery process.

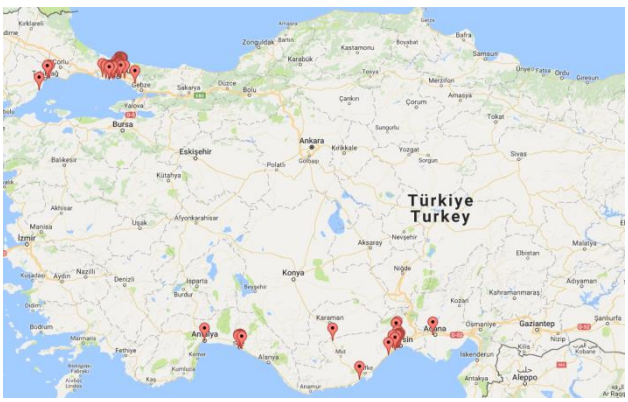

Figure 1. Visited locations of a social media user

Criteria	Value
Number of visited locations	95
Number of one-time visited locations	65
Number of socially important locations of the user	4

Table 1. Statistical information of the social media user

In this study, SocioSpatially Important Locations Mining (SS-ILM) algorithm, which is proposed by Dokuz and Celik (2017), is modified to decrease the execution time of the algorithm for faster discovery of socially important locations. The proposed Fast SS-ILM algorithm prunes rarely visited locations of social media users as early as possible, and so, unnecessary spatial calculations of non-frequent locations are avoided. With this strategy, the execution time of Fast SS-ILM algorithm decreases up to 20% with respect to the SS-ILM algorithm.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 introduces basic concepts of socially important locations mining and SS-ILM algorithm, and presents proposed Fast SS-ILM algorithm. Section 4

experimentally evaluates Fast SS-ILM algorithm. Section 5 presents conclusions and future works.

2. RELATED WORK

Spatial data mining gained huge attention in social media networking domain after social media networking sites started collecting user spatial data (Bao et al., 2015; Kefalas et al., 2016). Spatial co-location mining (Celik, 2015; Celik et al., 2008; Yu, 2016) and spatial clustering (Hu and Sung, 2005; Tung et al., 2001) are main topics of spatial data mining which could be used in social media datasets.

The literature related to socially important locations mining can be divided into two parts; user-level socially important locations mining and social group-level socially important locations mining. User-level socially important locations mining only aims to discover socially important locations of social media users. However, social group-level socially important locations mining aims to discover socially important locations for a social media user group.

In user-level socially important locations mining, Zhou et al. (2007) proposed clustering-based methods to mine personally important locations. Pavan et al. (2015) proposed an approach which is based on features of user movements to discover personal places of interest. Isaacman et al. (2011) developed clustering and regression based methods to discover personal important locations. Cao et al. (2010) proposed a framework to discover significant and semantically meaningful locations. Ying et al. (2011) proposed a method, which is based on clustering approaches, to predict next location of a user by considering geographical and semantic features. However, most of these studies are based on GPS data and these studies do not take into account social group preferences and they do not focus on developing computationally efficient algorithms.

In social group-level socially important locations mining, Zheng et al. (2009) proposed approaches and models to discover top *n* interesting locations and top *m* classical travel sequences by considering users' experiences and the relationships between users. Khetarpaul et al. (2011) proposed relational algebra based operations to discover interesting locations by analyzing trajectories of multiple users. Dokuz and Celik (2017) proposed an approach to discover socio-spatially important locations from a group of social media users. The proposed approach can also discover user-level socially important locations. Although these studies propose novel methods to discover social group-based socially important locations, these studies do not focus on developing computationally efficient algorithms

In this study, we propose Fast SS-ILM algorithm by modifying the algorithm of SS-ILM (Dokuz and Celik, 2017) to mine socially important locations efficiently. Proposed Fast SS-ILM algorithm prunes non-frequent locations as earlier as possible, and thus execution time of the algorithm decreases.

3. BASIC CONCEPTS AND MODELLING FAST SS-ILM ALGORITHM

In this section, first, basic concepts of socially important locations mining are given and then SS-ILM algorithm is introduced (Dokuz and Celik, 2017). Finally, the idea behind the Fast SS-ILM algorithm is presented.

3.1 Basic Concepts

Discovering socially important locations of a social media user group is composed of two parts, such as, user-level socially important locations discovery and social group-level socially important locations discovery (Dokuz and Celik, 2017). For user-level socially important locations discovery, the interest measures of location density and visit lifetime are used, and for social group-level socially important locations discovery, the interest measure of user prevalence is used as discussed in (Dokuz and Celik, 2017). The formulations of these interest measures are given as follows.

Definition 3.1.1. (Location Density): Location density is the proportion of number of occurrences of the user at a given location to the total number of occurrences of the user (Dokuz and Celik, 2017).

$$locationdensity_l^u = \frac{\#of\ occurrences\ of\ u\ at\ l}{\#of\ all\ occurrences\ of\ u} \quad (1)$$

Definition 3.1.2. (Visit Lifetime): Visit lifetime is the proportion of user's first and last visit of the location to the user's first and last occurrence in social media history (Dokuz and Celik, 2017).

$$visit\ lifetime_l^u = \frac{LastVisit_{l,u} - FirstVisit_{l,u}}{LastTime_u - FirstTime_u} \quad (2)$$

After calculation of each interest measure, the locations are checked whether they satisfy user-given *min_density* and *min_visit* thresholds. The locations which satisfy both thresholds are selected as socially important locations for user (SILU).

Definition 3.1.3. (User Prevalance): User prevalence is the fraction of the number of social media users who have location *l* as socially important location for user (SILU) to the total number of users (Dokuz and Celik, 2017).

$$user\ prevalence_l = \frac{\#of\ users\ who\ have\ l\ as\ SILU}{Total\ number\ of\ users} \quad (3)$$

After calculating user prevalence value, the locations are checked whether they satisfy *min_UP* threshold. The locations which satisfy *min_UP* threshold, are selected as socially important location for the user group (SIL).

3.2 SS-ILM Algorithm

SS-ILM algorithm, first, discovers user-level socially important locations (SILU) and from socially important locations for user (SILU) lists, it discovers social group-level socially important locations. Steps of SS-ILM algorithm are presented in Figure 2.

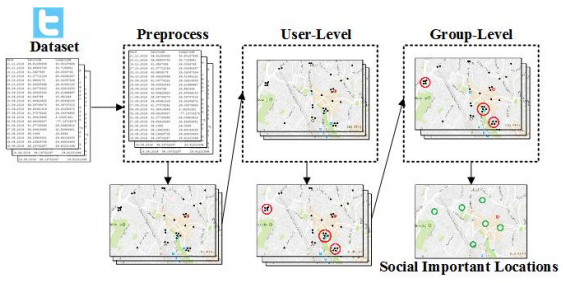

Figure 2. Steps of SS-ILM algorithm

As can be seen in Figure 2, first, the dataset is pre-processed and locations of each user is extracted. Then, using these locations of users, user-level socially important locations mining is performed and socially important locations for users (SILU) are discovered. Finally, social group-level socially important locations mining is performed and group-level socially important locations (SIL) are discovered.

3.3 Modelling Fast SS-ILM Algorithm

In this section, we introduce two new definitions, such as, occurrence count and candidate socially important location for user, to model our proposed algorithm, Fast SS-ILM. Based on these definitions SS-ILM algorithm (Dokuz and Celik, 2017) is modified.

Definition 3.3.1. Given a location *l* and a social media user *u*, the **occurrence count** of *u* at *l* is the number of occurrences of *u* at location *l*.

$$occurrencecount_l^u = \#of\ occurrences\ of\ u\ at\ l \quad (4)$$

Definition 3.3.2. Given a location *l* and a social media user *u*, the location *l* is a **candidate socially important location for user** *u* if occurrence count of user *u* at location *l* satisfies *min_occurrence* threshold value such as, $occurrencecount_l^u \geq min\_occurrence$

4. PROPOSED FAST SS-ILM ALGORITHM

Basic SS-ILM algorithm discovers socially important locations of social media users by calculating location density and visit lifetime values of every location that the users visit. However, some of the locations are visited rarely, i.e. one or two times. Thus, it's obvious that one time visit to a location will not end up to be a socially important location since it is not frequent. If an early pruning operation could be performed, these locations can be pruned before calculating location density and visit lifetime interest measure values and so the execution time of the algorithm will be decreased.

Fast SS-ILM algorithm aims to prune non-frequent locations before calculating location density and visit lifetime values, and thus decreases execution time of socially important location mining process. It, first, checks whether the locations satisfy minimum occurrence threshold for being a candidate socially important location, and then, the locations which satisfy *min_occurrence* threshold are further analyzed for being socially important location for users. Algorithm 1 presents the proposed Fast SS-ILM algorithm.

As can be seen in Algorithm 1, only steps 5, 6 and 14 are added to the classical SS-ILM algorithm (Dokuz and Celik, 2017). At step 5, *calculate-occurrence* function calculates occurrence count of location *l* for user *u*. If occurrence count of location *l* satisfies *min_occurrence* threshold, the location becomes a candidate socially important location which means that the location is visited enough number of times to be socially important location for the user and thus location density and visit lifetime values of the location are calculated to check that the location is actually socially important location for the user. Otherwise, the location is pruned and location density and visit lifetime values are not calculated. By applying an early pruning operation for non-frequent locations, unnecessary calculation of location density and visit lifetime values are avoided.

Algorithm 1. Fast SS-ILM Algorithm

Inputs:
D: Social media users' activity dataset
L: Set of extracted and labeled locations
min_occurrence: Minimum occurrence threshold
min_density: Minimum location density threshold
min_visit: Minimum visit lifetime threshold
min_UP: Minimum user prevalence threshold

Output: Social Important Locations (SIL) list

```
1.  allLocations = null
2.  for each user u in D
3.    SILU = null
4.    for each location l in L[u]
5.      occurrence = calculate-occurrence(l, u)
6.      if occurrence ≥ min_occurrence
7.        location_density = calculate-location-density(l, u)
8.        if location_density ≥ min_density
9.          visit-lifetime = calculate-visit-lifetime(l, u)
10.         if visit-lifetime ≥ min_visit
11.           SILU ← l
12.         end if
13.       end if
14.     end if
15.   end for
16.   allLocations ← SILU
17. end for
18. locs = calculate-UP(allLocations)
19. SIL=extract-socially-important-locations(locs, min_UP)
20. return SIL
```

5. EXPERIMENTAL EVALUATION

In this section, first the dataset is given, the pre-processing steps which are applied to dataset are discussed, and the experiments are presented. The experimental setup is given in Figure 3. The experiments are conducted to answer following questions:

- What is the effect of the number of users on execution time of algorithms?
- What is the effect of *min_occurrence* threshold on execution time of algorithms?
- How do socially important locations change with the increase of *min_occurrence* threshold?

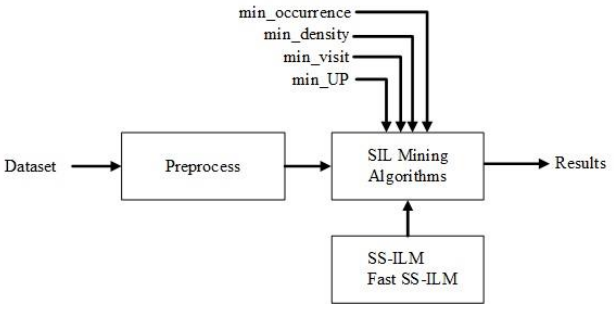

Figure 3. Experimental setup

The experiments are conducted on Intel Core i7 CPU with 3.40 GHz, and 8 GB of RAM.

5.1 The Dataset

Social media networks provide developers Application Programming Interfaces (APIs) (Twitter, 2017) to collect data from their servers. In this study, Twitter is used as a social media network and geographical Twitter data is collected. To collect data from Twitter servers, REST API and Streaming API were used. In addition, Twitter4j open source Java library (Yamamoto, 2017) was used to programmatically collect data from Twitter. Streaming API provides geographical boundary search on streaming tweets. To create the dataset with physically related social media users, Istanbul, Turkey based geographical search is performed and then users were collected. Approximately 2500 users were collected in this step. Then, REST API is used to collect all tweets of the users. Three parameters were collected for each user; date/time, latitude, and longitude. The dataset in this study is the dataset from Dokuz and Celik (2017).

5.2 Pre-processing Steps

In this section, data cleaning, user selection, temporal overweighting prevention, and location labeling procedures are explained.

5.2.1 Data Cleaning: In the experiments, we used the data from active Twitter users. In this study, active Twitter users are defined as users that send tweets no less than 50. If the number of tweets of a user is low, then the user is either passive or a new user. In the dataset, a proportion of Twitter users are spam users. To avoid spam users, we used two criteria; followers count and follower/friends ratio. If a users' followers count is less than 10 and follower/friends ratio is below 0.1, then this user is labeled as spam user and so that user was not included in the dataset. The values for these parameters are assigned according to many spam user detection literature and detailed information can be found in Benevenuto et al. (2010) and Zheng et al. (2015).

5.2.2 User Selection: The aim of this study is to compare Fast SS-ILM algorithm with SS-ILM algorithm, and thus the dataset and the users should be same. For this purpose, user selection approach of SS-ILM algorithm is applied to this study. The details of user selection approach can be found in Dokuz and Celik (2017).

5.2.3 Preventing Temporal Overweighting: When we analyzed the dataset, we realized that users may tweet (i.e., conduct social media activity) more than once at a location at the same time. If this behaviour becomes common, then a location might have more presence than its correct presence because the user was at that place once but tweeted several times. We defined this problem as temporal overweighting of a location. This problem is sometimes unintentional, such as a user has a conversation via tweeting to his/her friends and tweets several times within a short time span. To prevent temporal overweighting of a location, we defined a threshold, which is 60 minutes. If a user tweets more than once at the same location within 60 minutes, then we assume that this location information is not new and these tweets should be counted as once. With this approach, temporal overweighting of a location is prevented. However different approaches/criteria can also be applied to prevent temporal overweighting.

5.2.4 Location Labeling: The Twitter APIs provide accurate latitude-longitude pairs of user tweets. This approach is beneficial for getting fine-grained results, but also a problem for location labeling. For example, a shopping mall or a stadium might be located in 1 km2 area but we could define many distinct locations for this shopping mall or stadium because the accurate latitude and longitude pairs do not match. To overcome this problem, we defined a threshold for being same location for different latitude-longitude pairs. As used before in (Pavan et al., 2015), we defined this threshold as 100 m. If two locations are closer than 100 m, same labels are assigned to these two locations.

5.3 Experimental Results

In this section, the experiments are presented to evaluate the performances of proposed Fast SS-ILM algorithm and classical SS-ILM algorithm (Dokuz and Celik, 2017).

5.3.1 Effect of The Number of Users: In this experiment, we evaluated the effect of the number of users on runtime of algorithms of Fast SS-ILM and SS-ILM. The values of *min_occurrence*, *min_density*, and *min_visit* are set to 3, 0.01, and 0.05, respectively. We increased the number of users by 200 from 200 to 1000. The effect of the number of users is shown in Figure 4.

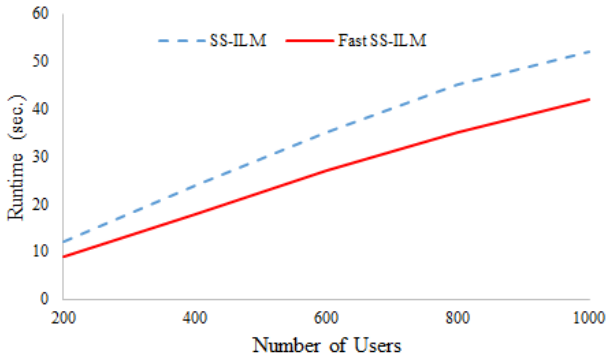

Figure 4. Effect of the number of users

As can be seen in Figure 4, both algorithms tend to increase runtime with the increase of the number of users. As can be seen, Fast SS-ILM algorithm consumes less time with the increase of the number of users and so it is more efficient on handling the increase of the number of users.

5.3.2 Effect of *min_occurrence* Threshold: In this experiment, we evaluated the effect of *min_occurrence* threshold on runtime of both algorithms. The values of *min_density*, *min_visit*, and the number of the users are set to 0.01, 0.05, and 1000, respectively. We increased *min_occurrence* threshold by 1 from 1 to 5. The effect of *min_occurrence* threshold is shown in Figure 5.

As can be seen in Figure 5, the runtime of SS-ILM algorithm keeps constant with the increase of *min_occurrence* threshold, however, the runtime of the proposed Fast SS-ILM algorithm decreases with the increase of *min_occurrence* threshold,.The proposed Fast SS-ILM outperforms the classical SS-ILM algorithm. Fast SS-ILM algorithm decreases runtime up to 20% with respect to the classical SS-ILM algorithm.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-4-W4-197-2017 | © Authors 2017. CC BY 4.0 License.

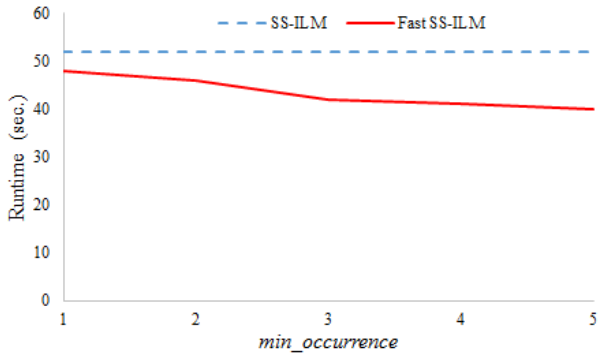

Figure 5. Effect of *min_occurrence* threshold

5.3.3 Evaluation of Extracted Results: In this experiment, we analyzed the effect of *min_occurrence* threshold on discovered socially important locations. The values of *min_density*, *min_visit*, and the number of users are set to 0.01, 0.05, and 1000, respectively. We compared top 10 locations with the increase of *min_occurrence* threshold from 1 to 5 by 1. Every value of *min_occurrence* is presented in Table 2 to check whether the results change with the increase of *min_occurrence* threshold. The locations are shown in Figure 6 and the results are shown in Table 2.

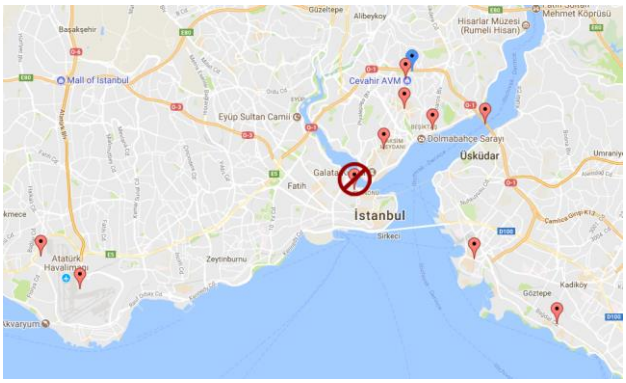

Figure 6. Socially important locations that are discovered by algorithms

SS-ILM	Fast SS-ILM (Varying *min_occurrence* threshold)				
	1	2	3	4	5
156	156	156	156	156	156
105	105	105	105	105	105
137	137	137	137	137	137
50	50	50	50	50	50
369	369	369	369	369	369
312	312	312	312	312	312
615	615	615	615	615	615
1177	1177	1177	**1120**	**1120**	**1120**
1120	1120	1120	**18**	**18**	**18**
18	18	18	**104**	**104**	**104**

Table 2. Top 10 socially important locations for algorithms and for *min_occurrence* threshold values of 1 to 5

As can be seen in Figure 6, red pins present SS-ILM algorithm results, and one additional blue pin presents extra location of 104 from *min_occurrence* value of 3. Dropped location is circled and overlined with red. Only one additional location is added to top 10 locations and one location is being dropped. Based on the figure, only one of the top 10 locations change and other locations remain same.

As can be seen in Table 2, up to *min_occurrence* value of 3, there is no change on discovered socially important locations. For the *min_occurrence* value of 3 and more, the locations' order changes. The reason for this is, the value of 3 is enough to discover candidate socially important locations for users who have relatively small social media history, and thus 3 and greater *min_occurrence* value changes the results of Fast SS-ILM algorithm. However, the top locations remain unchanged.

6. CONCLUSIONS AND FUTURE WORK

In this study, we proposed Fast SS-ILM algorithm to discover socially important locations in computationally efficient manner. The proposed algorithm is based on SS-ILM algorithm which is proposed by Dokuz and Celik (2017). Fast SS-ILM algorithm prunes non-frequently visited locations as early as possible and thus the number of candidate socially important locations decrease significantly. By decreasing candidate socially important locations, the spatial calculations of location density and visit lifetime measures decreases and so execution time of socially important locations discovery process decreases. Experimental results showed that the proposed Fast SS-ILM algorithm outperformed the classical SS-ILM algorithm.

As future works, Fast SS-ILM algorithm could be applied to big datasets and it could be applied to other application domains of social media mining, such as, location recommendation for social media users.

ACKNOWLEDGEMENTS

This research was supported by the Research Fund of Erciyes University, Project Number: FDK-2017-7233.

REFERENCES

Bao, J., Zheng, Y., Wilkie, D., Mokbel, M.F., 2015. Recommendations in location-based social networks: a survey. *Geoinformatica* 19, pp. 525-565.

Benevenuto, F., Magno, G., Rodriguez, T., Almeida, V., 2010. Detecting spammers on twitter. *Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference (CEAS)*, Redmond, WA, USA; July 2010, pp. 1-9.

Cao, X., Cong, G., Jensen, C.S., 2010. Mining significant semantic locations from GPS data. *Proceedings of VLDB Endowment*, 3(1-2), pp. 1009-1020.

Celik, M., 2015. Partial spatio-temporal co-occurrence pattern mining. *Knowledge and Information Systems*, 44(1), pp. 27-49.

Celik, M., Shekhar, S., Rogers, J.P., Shine, J.A., 2008. Mixed-drove spatiotemporal co-occurrence pattern mining. *IEEE Transactions on Knowledge and Data Engineering*, 20(10), pp. 1322-1335.

Dokuz, A.S., Celik, M., 2017. Discovering socially important locations of social media users. *Expert Systems with Applications*, 86, pp. 113-124.

Hu, T., Sung, S.Y., 2005. Clustering spatial data with a hybrid EM approach. *Pattern Analysis and Applications*, 8(1), pp. 139-148.

Isaacman, S., Becker, R., Caceres, R., Kobourov, S., Martonosi, M., Rowland, J., Varshavsky, A., 2011. Identifying important places in people's lives from cellular network data. In K. Lyons, J. Hihgtower, E.M. Huang (Eds.), *Pervasive Computing: 9th International Conference, Pervasive 2011, San Francisco, USA, June 12-15, 2011. Proceedings*, Springer, Berlin, Heidelberg, pp. 133-151.

Kefalas, P., Symeonidis, P., Manolopoulos, Y., 2016. A graph-based taxonomy of recommendation algorithms and systems in LBSNs. *IEEE Transactions on Knowledge and Data Engineering* 28, pp. 604-622.

Khetarpaul, S., Chauhan, R., Gupta, S.K., Subramaniam, L.V., Nambiar, U., 2011. Mining GPS data to determine interesting locations. *Proceedings of the 8th International Workshop on Information Integration on the Web: Conjunction with WWW 2011*, New York, NY, USA, pp. 1-6.

Pavan, M., Mizzaro, S., Scagnetto, I., Beggiato, A., 2015. Finding important locations: a feature-based approach. *16th IEEE International Conference on Mobile Data Management*, Pittsburgh, Pennsylvania, USA, pp. 110-115.

Tung, A.K.H., Hou, J., Han, J., 2001. Spatial clustering in the presence of obstacles. *Proceedings of the 17th International Conference on Data Engineering*, Washington, DC, USA, pp. 359-367.

Twitter, 2017. Twitter developers website, https://dev.twitter.com/ (27.07.2017).

Yamamoto, Y., 2017. Twitter4j java library. https://twitter4j.org/en/index.html (27.07.2017).

Ying, J.J.-C., Lee, W.-C., Weng, T.-C., Tseng, V.S., 2011. Semantic trajectory mining for location prediction. *Proceedings of 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, New York, NY, USA, pp. 34-43.

Yu, W., 2016. Spatial co-location pattern mining for location-based services in road networks. *Expert Systems with Applications*, 46, pp. 324-335.

Zheng, Y., Zhang, L., Xie, X., Ma, W.-Y., 2009. Mining interesting locations and travel sequences from GPS trajectories. *Proceedings of the 18th International Conference on World Wide Web*, New York, NY, USA, pp. 791-800.

Zheng, X., Zeng, Z., Chen, Z., Yu, Y., Rong, C., 2015. Detecting spammers on social networks, *Neurocomputing*, 159, pp. 27-34.