Computer Communications 100 (2017) 78–90

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

A deep dive into location-based communities in social discovery networks

Kanchana Thilakarathna[a,b,*], Suranga Seneviratne[a,b], Kamal Gupta[c], Mohamed Ali Kaafar[a,b], Aruna Seneviratne[a,b]

[a] Networks Research Group, Data61, CSIRO, Australia
[b] The University of New South Wales, Australia
[c] IIT Delhi, India

ARTICLE INFO

Article history:
Received 15 February 2016
Revised 4 November 2016
Accepted 20 November 2016
Available online 21 November 2016

Keywords:
Human mobility
Link prediction
Social discovery networks

ABSTRACT

Location-based social discovery networks (LBSD) is an emerging category of location-based social networks (LBSN) that are specifically designed to enable users to discover and communicate with nearby people. In this paper, we present the first measurement study of the characteristics and evolution of location-based communities which are based on a social discovery network and geographic proximity. We measure and analyse more than 176K location-based communities with over 1.4 million distinct members of a popular social discovery network and more than 46 million locations. We characterise the evolution of the communities and study the user behaviour in LBSD by analysing the mobility features of users belonging to communities in comparison to non-community members. Using observed spatio-temporal similarity features, we build and evaluate a classifier to predict location-based community membership solely based on user mobility information.

1. Introduction

Location-acquisition technologies like GPS on smartphones have rapidly promoted the use of location-based services. Foursquare[1] is a traditional Location-Based Social Networks (LBSN) which enables users to share their real-time locations, by checking into a set of venues in the proximity of their geographic location. The popularity of LBSNs has attracted substantial research interest as data collected from LBSNs enable studies of individuals' online and offline behaviours, ranging from human mobility modelling [1,2] to user behavioural analysis [1,3], user re-identification [4], user anonymity analysis [5,6] and social relationship recommendations [7,8].

A new category of LBSNs are Location-Based Social Discovery (LBSD) networks that are specifically designed to enable users to discover and communicate with nearby people such as Sweetr [9], WhosHere,[2] WeChat,[3] Yik Yak [10] and Momo.[4] Recently in April 2014, Facebook launched a new opt-in service called “Nearby Friends”, which enables users to share real time location and discover nearby friends [11]. As an emerging new type of services, LSBDs are yet to be thoroughly studied compared to number of measurement studies of LBSNs such as *Twitter*. This is primarily due to the inability to capture data and unavailability of real-world datasets.

The potential offered by LBSD services is unique, and is mainly driven by the different nature of these applications, where users typically discover and establish new social relationships based on their actual real-time location. The intrinsic key difference between general LBSNs and specific LBSDs lies then in the understanding of physical human social interactions versus distant online social interactions. The peculiar features of LBSDs offer new opportunities to perform in-depth studies of physical location-based social links at a large scale, and in particular observing human communities and groups which would in turn allow the design of efficient location-aware content delivery, caching and recommendation strategies.

[*] Corresponding author.
E-mail addresses: kanchana.thilakarathna@data61.csiro.au, kanchana.thilakarathna@nicta.com.au (K. Thilakarathna), suranga.seneviratne@data61.csiro.au (S. Seneviratne), kamal_ee@student.iitd.ac.in (K. Gupta), dali.kaafar@data61.csiro.au (M.A. Kaafar), aruna.seneviratne@data61.csiro.au (A. Seneviratne).

[1] https://foursquare.com.

[2] https://web.whoshere.net.

[3] http://www.wechat.com/en/.

[4] http://www.immomo.com.

http://dx.doi.org/10.1016/j.comcom.2016.11.008
0140-3664/Crown Copyright © 2016 Published by Elsevier B.V. All rights reserved.

In this paper, we study LBSD network communities using a dataset collected from an increasingly popular social discovery mobile application, "Momo". Momo offers three main functions, 1) It allows users to discover other Momo-enabled devices in their surroundings based on the geographical proximity. 2) It enables instant messaging that allows (subsequent) user communication. 3) Momo also provides users with function to join communities that are created in their proximity. Unless users explicitly opt-out of location updates, by default user status and location is publicly revealed and updated to other nearby users. Hence, by monitoring the Momo service, a rich set of spatial and temporal information about users and their communities can be collected. Overall, we analyse information of more than 176,000 Momo communities, comprising more than 1.4 million community members. We also study over 355 million geo-updates of 6.7 million Momo users collected over a period of 71 days. We share our collected dataset with the research community.[5]

The contributions of this paper are the following. First, we study the social structure of the new network comprising LBSD application users, with a focus on the community to community, user to user and user to location networks. We show that Momo users although having a higher activity, exhibit a lower geographical diversity with higher regularity. Further, the results show that Momo users tend to join communities in the local neighbourhood and thereby users in small cities are more connected to each other compared to larger cities. We compare the feasibility of random and degree-popularity based selection of locations and users in opportunistically propagating content. In particular, the results show that it is possible to reach more than 50% of users within 3 days through approximately 15% of the locations or users, providing opportunities for location-based caching or opportunistic delay-tolerant content delivery.

Second, we characterise user mobility behaviours in LBSDs and analyse mobility patterns of users depending on whether or not they are community members. We show that there is a high similarity in individual user trajectories within communities, which can be explained by the social discovery nature of the application itself.

Third, we leverage these location-based similarity metrics to train a classifier for community membership prediction through a supervised learning approach. We evaluate the efficiency of the classifier, that solely based on mobility patterns, aims to predict social interactions through LBSD community membership, for both cases of worldwide users and users belonging to a single city. Our results show that even sparse information derived from individual trajectories provides a very accurate prediction of community membership, with a precision up to 91% and recall of 72%.

The rest of this paper is organised as follows. We describe the dataset in Section 2. Then, we introduce and analyse structural characteristics of different subgraphs in which our data has been embedded in Section 3. In Section 4, we characterise user mobility features within and out of the communities for the purpose of community membership prediction. We evaluate the efficiency of community membership prediction in Section 5. We outline related work in Section 6. Finally, Section 7 concludes the paper.

2. Datasets and general characteristics

We first briefly describe the operations and unique features of the LSBD mobile application - Momo and the datasets used in our study. We follow up with the study of the basic characteristics and the evolution of the location-based communities in Momo over time.

2.1. An LBSD application: Momo

Momo is a mobile application, available for both iOS and Android devices, that has attracted more than 100 million users. Momo includes typical social network services ranging from instant messaging and status updates to location-based features including users' location "check-in" as a profile status and venues rating (reviews). However, Momo is primarily used as a social discovery service, where once a user launches the application, the user can *discover* a list of nearby Momo users ranked by their geographic proximity to the user's actual location. The location and the status of the user is simultaneously publicly revealed to other nearby users (and friends[6]). This discovery feature makes Momo quite unique compared to other typical LBSN services. Individuals can then send personal messages, share content or update status visible only to users nearby. They can also add these newly discovered users as new friends.

Another key distinguishing feature of Momo is the location-based communities. Momo users can create communities (groups) at a particular location, which are then advertised among nearby users. Likewise, users can discover communities created in their proximity, ranked by distance to their current location, and send a request to join the community. In essence, Momo communities are tightly linked to the location where they have been created, and the community membership indicates that users have at least once visited that location after the community has been created. While communities can be created for different purposes, the application is mainly intended to foster proximity-based social interactions (social discovery function) among users with similar interests. Community members can then share content only within the community, discover fellow members' respective locations and operate in a closed (multicast) "circle". To some extent, Momo communities are similar to the notion of circles of friends in OSNs, except that Momo friends happened to be at the same location.

As mentioned in the introduction, in addition to MoMo, there are number of other LBSDs or social networks that also provide LBSD services. For example, *YikYak* [10] allows to anonymously create or discuss message threads within a radius of five miles and mainly targets college campuses as a service to spread local news and gossips [12]. *Sweetr* [9] has similar features with Momo and allows to discover nearby users and share content such as text, photos, audio, and video with them. Two popular social networks *Facebook* and *WeChat* have LBSN features named as *Near By Friends* [11] and *People Nearby* [13], respectively. In general, LBSDs target applications such as dating, find people with similar interests, discuss or spread locally important news items, content sharing or game play with nearby users, and providing more personalised recommendations.

2.2. Datasets

Momo-enabled devices communicate with the server via a set of network APIs. The *nearby* API was used as in [14] to collect an extensive set of location updates originating from different monitored areas. By varying the geographic coordinates of a Momo client, a crawler was deployed to collect location updates of nearby users in every 15 min for a period of 71 days (from May to October 2012). This enabled the collection of location updates of more than 6 million users worldwide. We refer to this location updates dataset as Updates and Table 1 introduces basic characteristics of Updates.

The second dataset, Communities consists of a snapshot of the different communities existing in the Momo network as of

[5] Dataset will be available with the published version of this paper.

[6] Similar to other OSNs, Momo users unconditionally share their location with their friends irrespective of their location.

Table 1
Momo LBSD network summary.

Updates statistics		
	users ∈ communities	users ∉ communities
# of users	355,490	6,438,821
# of updates	45,797,455	311,668,101
# of location grids	204,585	551,685
Avg updates per user	128.83	48.40
Communities statistics		
# of communities	176,874	
# of distinct members	1,465,393	
Avg users per community	13.47	
Avg communities per grid	3.33	

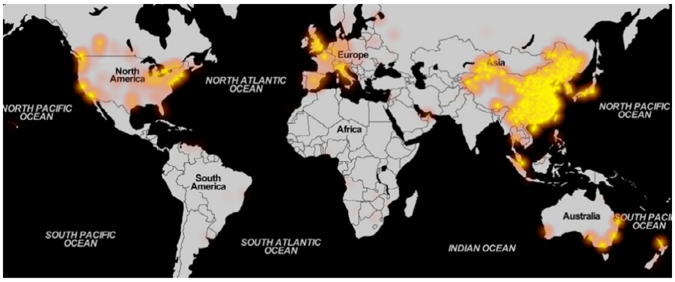


Fig. 1. Distribution of the Momo communities at the end of the data collection period.

February 2013. Using the *group* API, information about more than 176K Momo communities spreading over 48 countries, and comprising more than 1.4 million members, were collected. Communities dataset includes information such as the creation time, the creation location, the creator ID as well as the members IDs. Table 1 introduces the main characteristics of Communities dataset. The location grids correspond to the coverage areas of the Momo client that was used to crawl real-time activity of nearby users.[7] The Momo client recursively selected candidate monitoring points within the location grids of 1 × 1 km^2 area for purpose of discovering all other users' activities within the considered monitoring area. Further details of crawler design and the utilized Momo APIs can be found at [14].

The notion of communities has been rolled out several months after we start collecting Updates and as such Updates contains mobility information of only 24% of district community members (355,490). We also note that due to limitations of the Momo APIs, it was only possible to monitor the real-time behaviour of active users in the monitored areas as per the content of the Updates dataset. Fig. 1 shows the global distribution of the 176,874 communities. While Momo is most widely used in China, other areas in US, Canada, Europe and Australia also attract a significant number of communities. Beijing represents the most dense city with 8435 groups. Next, we present the basic characteristics and evolution of the Momo location-based communities since this feature has been rolled-out (in October 2012).

2.3. Characteristics of Momo communities

Table 1 introduces the main characteristics of Updates and Communities datasets. While the number of users belonging to communities is drastically lower than non-members, interestingly community members exhibit more than two times higher average number of updates per user compared to non-members. On average, each community comprises more than 13 members, and 3–4 communities co-exist on each grid.

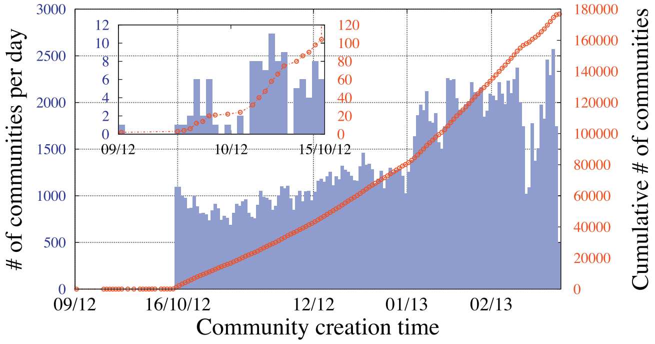


Fig. 2. Growth in community creations per day.

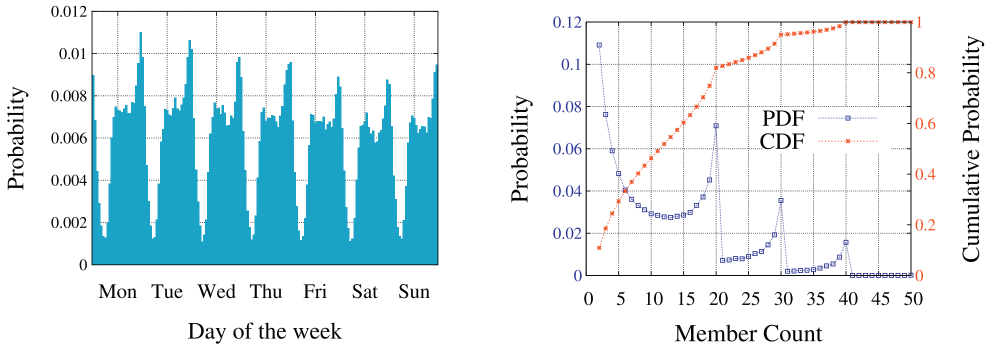
Fig. 2 depicts the daily community growth. The very first set of communities can be considered experimental, with a one-week slow start of less than on the average 10 new communities per day. The pace of community growth became extremely faster in early October 2012 with almost 1000 communities created each day. The growth received a major bursts with a daily increase of 2000 new communities at the end of January 2013 (corresponding to the Chinese New Year public holiday, and a the release of a new version of Momo on Android and iOS). Fig. 3a shows the daily community creation patterns in China.[8] There is a noticeable peak between 9:00 PM and 12:00 midnight, while newly created community distribution is merely steady throughout the day. The peak of community creations during late night suggests that Momo application is heavily used during that period of time as people would generally be more inclined to discover nearby users after work. Fig. 3a also indicates that comparatively fewer community creations are happening in weekends compared to weekdays. This type of information can be leveraged when designing value added services on top of LBSDs. For example, these observations allow advertisers to make decisions on what is the most suitable time to advertise some products for LBSD communities so that the maximum number of users can be reached.

The maximum number of users within a Momo community is set according to the level of activity of the community. Initially, the maximum size is limited to 20 members by default. To increase the maximum size from 20 to 30 (resp. 40 and 50), community members have to generate more than 300 messages per day for 10 consecutive days (resp. 40 and 100). Fig. 3b shows the distributions (PDF and CDF) of the community size. Overall, approximately 80% of the groups have less than 20 members. Notably, once a community is upgraded to the next level, there is a high likelihood that the community size increases to reach the maximum size of the new level (depicted by the sharp drop in the probability at the size limits). Since active participation is required to expand the community size, larger communities can be considered to contain sustainable social relationships.

Further investigating the potential of the dynamic structure of the social discovery aspects of the application, we find that 80% of the communities are created in only 40% of the locations as shown in Fig. 4a. We also observe that 50% of the locations accommodate only one community, i.e. the majority of the locations are either hot or cold spots for community creation. To realise the "rich-gets-richer" phenomena, we analyse the effect of prior popularity of an existing location (attractiveness of a location) on the

[7] Monitoring locations for clients are based on a modified version of the 2-dimensional closest point search algorithm in lattices [15].

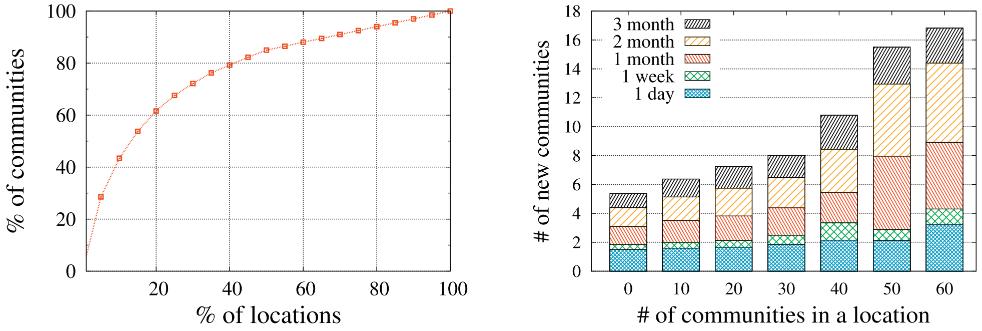
[8] Communities created in China represent 98% of the total number of monitored communities.



(a) Community creation time

(b) Community size

Fig. 3. Creation times and size of communities.



(a) Location diversity

(b) Location attractiveness

Fig. 4. Distributions of community creation location.

probability of new community creations at that location, as shown in Fig. 4b. We observe that if a location already attracted 51–60 communities, approximately 17 communities would be created at that location in the next 3 months. When the existing number of communities drops to 1–10 communities, it will only get on average 6 new communities in the next 3 months. Up to the size of 60 communities, there is a clear trend that the more popular the location is, in terms of previously created communities, the more new communities it will attract. Note that the statistics for a number of communities in a location larger than 60 are not illustrated, as we observe only 6 locations with more than 60 communities. Hotspots provide the spatial dimension for service providers who operate on top of LBSDs. For example, similar to previously identified peak times, location hotspots with respect to community activities also can be leveraged to efficiently reach large numbers of users for services such as advertising, promotions, and content distribution.

3. Social structure of Momo

In this section, we characterise the communities and community members in the Momo LBSD network. We first model Momo as a tripartite graph which users, communities and locations are represented by nodes as shown in Fig. 5. Communities are represented as *community nodes* - $c \in \mathbb{C}$. Community members are considered as *user nodes* - $u \in \mathbb{U}$, where an edge between two user nodes u_i and u_j is defined by their membership of the same community and is denoted by $e(u_i, u_j) \in \mathbb{E}_u$. Likewise, an edge $e(c_i, c_j) \in \mathbb{E}_c$ between two community nodes c_i and c_j reflects that both share at least one member. Edges $e(c_i, u_j)$ represent the membership of user nodes in communities. All locations visited by the users are represented as *location nodes* - $l \in \mathbb{L}$. For each location node l_i visited by a user node u_i, the graph comprises an edge $e(u_i, l_i) \in \mathbb{E}_l$, where location nodes represent grids of 1 × 1 km^2. In the following, we analyse the three sub-graphs shown in Fig. 5 which represent the Community Graph: $G_c(\mathbb{C}, \mathbb{E}_c)$, the User-User Graph: $G_u(\mathbb{U}, \mathbb{E}_u)$ and the User-Location Graph: $G_l(\mathbb{U}, \mathbb{L}, \mathbb{E}_l)$.

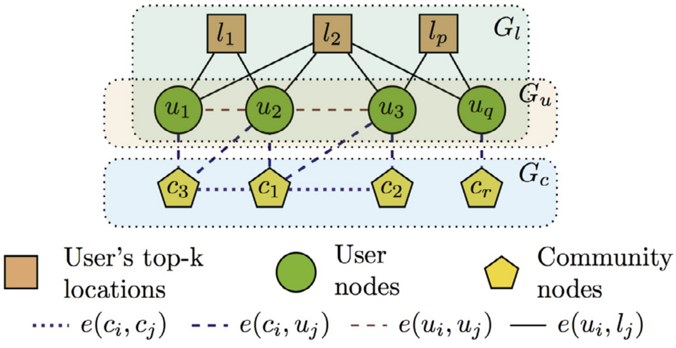


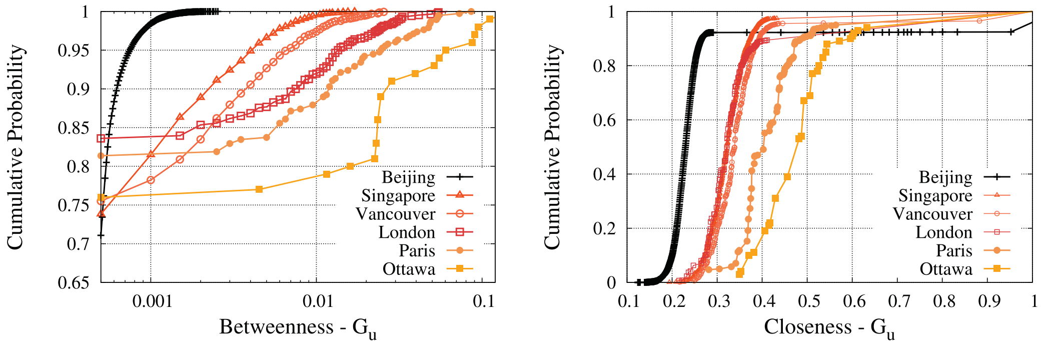
Fig. 5. Tripartite Graph representation of Momo.

Table 2
Summary of structural properties of G_c.

Vertices	Edges	Density	Diameter	Assortativity
64124	1,227,902	9.117e-05	14	0.216952

3.1. Community graph

Momo community graph, $G_c(\mathbb{C}, \mathbb{E}_c)$ corresponds to communities sharing at least one common member and comprises 164,124 nodes (communities) and 1,227,902 edges. In Table 2, we present the basic characteristics of the community graph. We observe that the Momo community graph has a very low density (measured as



(a) Betweenness centrality

(b) Closeness centrality

Fig. 6. Centrality of users within cities of different densities around the world.

Table 3
Summary of structural properties of G_u.

Property	Beijing	Shanghai	Vancouver	London	Paris	Ottawa
Nodes	72580	3004	1653	860	387	100
Edges	1.02e06	46834	24286	9252	5535	828
Density	0.00039	0.0103	0.0178	0.0251	0.0741	0.1673
Diameter	12	7	6	7	5	4
Assortativity	0.2412	0.5077	0.6221	0.7134	0.7279	0.7361

$||E_C||/(\binom{||\mathbb{C}||}{2}))$ indicating a lower number of edges between Momo communities. This suggests that Momo users join a smaller number of communities. We find that 95% of the users joined less than 4 communities indicating that they join only the most relevant groups. Moreover, community size can be only increased by showing community activities such as posts and chats. As a result, community creators will also add only the most relevant users to the communities who will potentially interact with rest of the members. Table 2 also shows that Momo community graph has a high degree assortativity. This indicates that communities with the same degree tend to be connected to each other in Momo. Again this is due to the limited number of community membership per user. As a result of the limited community membership, the size of the communities varies less. For example, we observed that 50% of the Momo community nodes have less than 10 users and this lower variation in community size causes high degree assortativity.

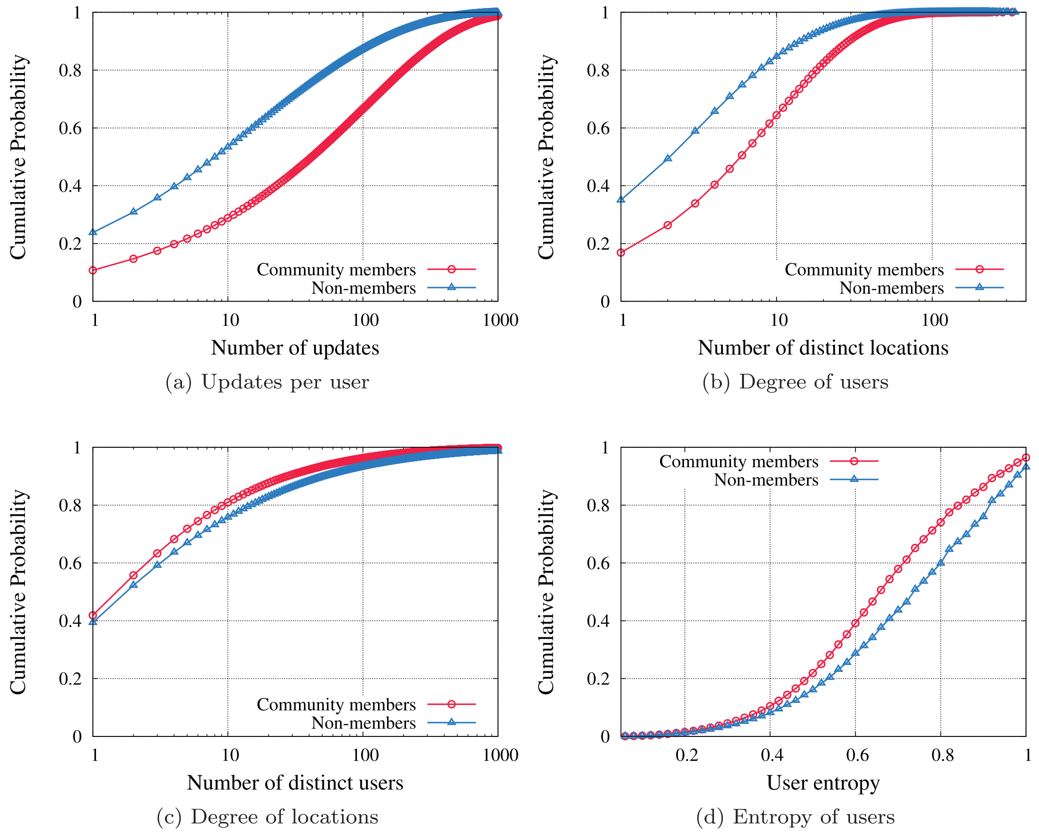
To understand which communities Momo users join, we calculate the assortativity coefficient, $r = \frac{tr(e)-||e^2||}{1-||e^2||} \in$ [-1,1] as per [16], for the Momo community graph according to the city where communities have been created. Here $e = E/||E||$ is the normalised mixing matrix. Elements of E, i.e. E_{ij} indicate the number of edges that connect communities created in city i to the communities created in city j and $||E||$ is the sum of the all elements in E. Assortativity coefficient, $r = 1$ means the edges between communities are highly assortative with group creation location while $r = 0$ means the edges are random. $r = -1$ means edges are disassortative to the group creation location. We find that the assortativity coefficient by city is 0.6982; a high value that indicates as expected that community membership in Momo is highly associated with the community location, as users tend to join communities in the same neighbourhood (i.e. consistent with location-based social discovery function of the application).

3.2. User-User graph

As shown previously, the connectivity of Momo community graph is highly related to the city of the community. We then characterise the behaviour of Momo users in geographical neighbourhoods. We select six major cities around the world and measure structural properties of the user-user graph G_u, within each city as summarised in Table 3. We observe that the lower the number of users in a city, the larger the density and degree assortativity in the user-user graph. This indicates that within smaller cities users are more likely to connect to each other. Although this may be not surprising as the social discovery nature of Momo drives such a behaviour, this observation finds application in opportunistic content delivery networks. Since an edge represents the same community membership, there is high probability of visiting the same location (at least the community creation location).

We further evaluate the significance of individual users in a community through Betweenness and Closeness centrality measures. Betweenness $B(u) = \sum_{s,t\in\mathbb{U}} \frac{\sigma(s,t|u)}{\sigma(s,t)}$, is the fraction of shortest paths that pass through a user u, where $\sigma(s, t)$ is the number of shortest paths between users s and t and $\sigma(s, t|u)$ is the number of shortest paths passing through u. Fig. 6a shows that betweenness of individuals increases with the density of the city. In particular, while there are 10–15% of users with betweenness greater than 0.01 in Paris and London, all users in Beijing are with betweenness less than 0.01. Therefore, there are more users of higher importance in terms of user to user connectivity in smaller cities compared to larger cities. The users of higher betweenness could be a good choice for content caching applications such as opportunistic content delivery, as they improve the dissemination of content effectively.

Fig. 6b shows the distribution of user closeness in all chosen cities. Closeness is defined as $C(u) = 1/\sum_{v\in\mathbb{U}} \frac{d(u,v)}{\|\mathbb{U}\|-1}$ and it represents the inverse of the average distance ($d(u, v)$) to all other users. Similar to betweenness, closeness of individuals increases with the density of the city they belong to. For instance, approximately 90% of users in Beijing have a closeness value less than 0.3, whereas more than 95% of users in Paris have a closeness value higher than 0.3. This indicates that the probability of two users having similar set of interests is higher in smaller cities due to the higher closeness in user-user relationships. Therefore, when the interests of a few users are known (especially the users with higher closeness), there is a higher probability that a significant portion of other users will have a similar set of interests in these small cities. This observation can be exploited for targeted advertising.



(a) Updates per user

(b) Degree of users

(c) Degree of locations

(d) Entropy of users

Fig. 7. Diversity and frequency of updates in Momo community members and non-members.

3.3. User-Location graph

Next, we analyse the bipartite graph $G_l(\mathbb{U}, \mathbb{L}, \mathbb{E}_l)$, considering Momo users (user nodes $\mathbb{U}$) and locations at which they checked-in as the location nodes $\mathbb{L}$. G_l consists of 355,490 user nodes belonging to at least one community connected to the set of 204,585 location nodes. There is an edge $e \in \mathbb{E}_l$ between a user node and a location node, if the user has updates in that location.

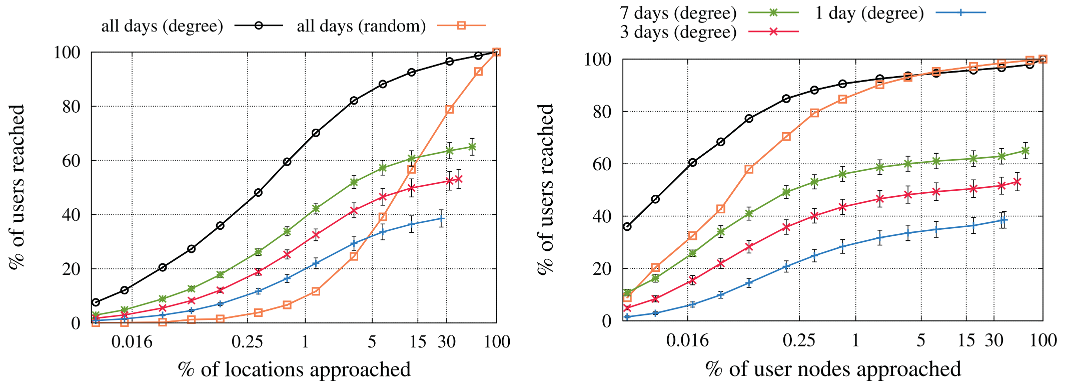
We characterise the user mobility based on the structural properties of G_l. Fig. 7a illustrates the activeness of users by the CDF of updates per a community member and a non-member. Approximately 35% of Momo community members have more than 100 updates compared to the 10% for Momo users who do not belong to any community. We believe that these non-members are probably using Momo as traditional LBSN service while the community members take advantage of the LBSD features.

In a traditional LBSN such as *Foursquare*, location updates are user driven, (i.e. when the user visits an important landmark and she decides to report). Thus, the data collected from such a LBSN application does not represent the actual mobility patterns of the user. However, in Momo, the application in background is continuously scanning for nearby users and therefore location information collected from such an app more accurately represents user trajectories. Therefore, it can be expected that the number of distinct locations per user in a LBSN dataset is less than the number distinct locations per user in a LBSD dataset, (i.e. Momo provides fine-grained rich sample of user trajectory data compared to LBSN service like *Foursquare*).

This is exemplified by the fact that Momo community members visit on average more distinct locations than non-members as shown in Fig. 7b. More than 85% of Momo non-members have visited less than 10 distinct locations, highlighting the application usage difference between Momo community members and non-members. Fig. 7c illustrates the number of distinct users who visited a particular location (1 × 1 km^2 area). There are nearly 40% of locations with only one distinct user. This could be related to the home location of individual users. For other locations, the distribution of community members and non-members are comparatively similar (Fig. 7c) despite the higher activeness of community members.

Moreover, we characterise the geographical diversity in user trajectories. Entropy of an individual user can be defined as $-\sum p\frac{\log p}{\log n}$, where p is the portion of updates at a location and n is the number of distinct checked-in locations. The entropy value of zero signifies that a user consistently visits only a single location, while entropy value of one indicates that a user has visited more than one location in equal proportions. Fig. 7d illustrates the fact that Momo community members visit a selected set of locations more than other locations as the entropy values are lower than non-members. Overall, these basic characteristics of user mobility suggest that there is a difference in mobility behaviours between community members and non-members, which is further exploited in the next section.

Then, we investigate how many of the user's top locations are representative of the entire graph. G_l^k, where $k = 1, 2$ and 5, represents the sub-graphs consisting of only k most frequently visited locations for each user. Table 4 summarises the structural properties of these sub-graphs where $g(G_l)$ is the largest connected component. G_l^1 consists of user's most frequent location only and therefore all user nodes are connected to just one location node. As a result, the largest connected component of G_l^1 consists of only 0.14% user nodes and one location node. However, G_l^2 has a largest connected component covering over 92% of the user nodes. Increasing k further has only an insignificant impact as G_l^5 and G_l^{all} cover only an additional 3–4%.



(a) Coverage of users through location nodes

(b) Coverage of users through user nodes

Fig. 8. Portion of users that can be reached from different portions of location nodes and user nodes.

Table 4
User-Location graph $G_l(\mathbb{U},\mathbb{L},\mathbb{E}_l)$.

	Top k locations			
	Top-1	Top-2	Top-5	All
Users	355,490	355,490	355,490	355490
Locations	62,687	84,370	120,062	204,585
Components	62,687	18,941	12,409	10,175
$\|u \in g(G_l)\|$	0.14%	**92.34**%	95.25%	**96.22**%
$\|l \in g(G_l)\|$	–	68.29%	86.57%	94.09%

To illustrate the applicability of these findings, we first select the top x% (with x varying from 0 to 100) of the location (204,585) and user (355,490) nodes, based on degree centrality in G_l and observe the coverage of user nodes. A node u is considered as covered by a node v, if there is a path between u and v. The baseline selection criteria in such applications would be to randomly select certain users or location nodes and then calculate the maximum number of users that can be reached. In Fig. 8a, we first compare random and degree based location selection, which demonstrates considerable improvement in coverage for degree based selection compared to random selection for the total trace duration. In particular, random 5% of the locations cover less than 30% of users, whereas top 5% of the locations with highest degree centrality cover more than 85% of users. Fig. 8b shows no significant improvement in degree based user selection due to the lower diversity of check-in locations of Momo users, i.e. users visit selective set of locations repeatedly as shown in Fig. 7b. If we select more than 1% of users, random and degree based selections provide nearly the same coverage and in some cases random selection performs better than degree based selection. Note that random selection does not require any user private information such as check-in patterns or connected networks. Therefore, these results provides insights to develop privacy-aware opportunistic content dissemination strategies that perform similar to schemes that collect number of user private information to make context-aware selection of users to replicate or cache content.

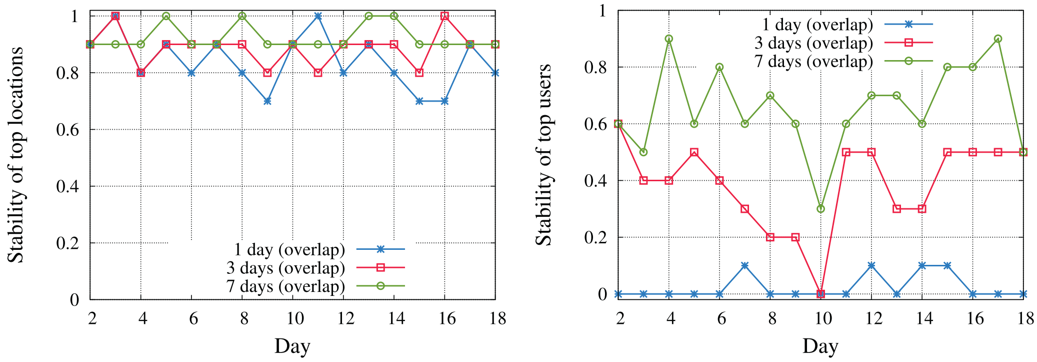
Then, the propagation time is constrained to 1, 3 and 7 days to investigate the effectiveness of opportunistic content dissemination in real-life applications. If we select top 5% (∼ 10K) of the location nodes, it is possible to cover more than 30% (∼ 107K) of the total number of users within just one day as shown in Fig. 8a. According to Fig. 8b, it is possible to achieve 30% of coverage within one day from selecting even lesser portion of user nodes (∼ 1%) due to the fact that users being dynamic are more likely to reach other users compared to stationary location nodes. Further, approximately 50% (∼ 178K) of users can be covered within 3 days by selecting 15% of location or user nodes. The portion of users that can be covered within 7 days are less than 70% in both cases.

As we have observed that the top few nodes play an important role in content delivery applications, we now measure the stability in rankings of nodes over time. Consider the set of top 10 user nodes on two consecutive days $\mathbb{U}_1$ and $\mathbb{U}_2$, and then the overlap of the two sets is given by $O(\mathbb{U}_1,\mathbb{U}_2)=\frac{\|\mathbb{U}_1\cap\mathbb{U}_2\|}{10}$. $O(\mathbb{U}_1,\mathbb{U}_2)$ represents the stability of the top 10 nodes in the considered time window. Fig. 9 shows the variation in the stability index ($O(\mathbb{U}_1,\mathbb{U}_2)$) of top 10 user and location nodes over a period of 18 days considering time windows of 1, 3 and 7 days. Location nodes are almost perfectly stable irrespective of the time window as shown in Fig. 9a, i.e. most popular locations of the day will be the most popular locations of the day after as well, with very high probability. The behaviour of user nodes however changes with time. Fig. 9b shows that the probability of having the same top 10 user nodes in two consecutive days is almost zero. However, the stability of the user nodes increases as we increase the window size. If we consider one week time window, the probability of having the same top 10 users in the next week increases to approximately 70% indicating regular behavioural patterns of users.

3.4. Summary of results

In this section, we characterised the communities and user behaviour in the Momo LBSD network and compared it with other online community networks. The main findings are;

- Momo users join only fewer number of communities compared to other community networks due to restriction in community memberships and the social discovery nature of the application. As a result, users tend to join only communities in the local neighbourhood. Consequently, users in small cities are more connected to each other compared to larger cities' users.
- LBSD users visit a limited set of locations regularly. Further, Momo community members are more active LBSD users compared to non-members.
- Degree (popularity) based selection performs better in location selection for content caching, while random selection is as effective as the degree in selecting users for content caching. Since random selection does not require any user private information, this observation provides insights to develop privacy-aware content delivery methods.
- Considering only the top 15% of the locations or the users with the highest degree centrality, allows to reach more than 50% of the users within 3 days providing insights for possible applications such as delay-tolerant content sharing and advertisement propagation.



(a) Stability of top 10 locations

(b) Stability of top 10 users

Fig. 9. Stability of the degree centrality $O(\mathbb{U}_1, \mathbb{U}_2)$ of the top 10 user and location nodes over 18 days.

- The rank of locations does not change significantly over time, while the rank of users is relatively unstable. However, due to regular behavioural patterns of users, rank of users shows a stable behaviour when considering longer time windows.
- The probability of creating new communities is higher on already popular locations (“rich-gets-richer”), which augments the applicability of services such as distributed content caching at top locations.

4. Characterising user mobility features

In this section, we characterise user mobility based on location updates of Momo users. We analyse the similarity in user mobility patterns with a focus on features that distinguish community members from non-members for the purpose of community membership prediction in Section 5. Extensive amount of information generated by LBSD users, including the time and frequency of locations they have visited, can be used for community membership prediction. While different types of information can be utilised, e.g. user attributes available from private/public profiles, friendship lists, etc. we only focus on information that is easily accessible by a LBSD application, i.e. time and location of user check-ins.

We consider all users who belong to at least one community and have at least one update in Updates dataset as our working set, $|\mathbb{U}| = 355,490$. We define the pairs of users belonging to the same community to have a positive (undirected) community membership - *Positives*. If a given pair of users does not belong to a same community, we assume a negative community membership - *Negatives*. However, when selecting users with negative community memberships, if two users who do not have a positive community membership are randomly selected from the set of world wide users, they are highly unlikely to show a mobile homophily. A more challenging scenario, is when users with negative community memberships are selected from a single city, that enables the possibility of them showing same mobility features as users with positive memberships. In the subsequent analysis, we consider both cases of positives/negatives extracted from users of either worldwide and belonging to a single city. A user belongs to a city if her most frequent updates are from that city. We present various mobility measures for comparing users in Beijing (city with the highest number of users) and Vancouver (city with the highest number of users outside China).

4.1. Spatial distance in updates

Let the updates trajectory of a user u be a list of tuples $< l_u, t_u >$, where $l_u \in \mathbb{L}_u$ represents locations and t_u timestamps. The location and timestamp of the i^{th} update is denoted by $l_u(i)$ and $t_u(i)$ respectively and the total number of updates is denoted by N_u.

We extract the most frequently visited location of a user u as,

$$MV(u) = \arg\max_{l \in \mathbb{L}_u} N(u, l)$$

where $N(u, l) = \sum_{i=1}^{N_u} \delta(l, l_u(i))$, and $\delta(x,\ y)=$ 1, if $x = y$ and 0 otherwise. $N(u,\ l)$ is the number of times user u has visited location l. The spatial distance between the most frequently visited locations of users u and v is then calculated as, $D(u, v) = \text{distance}(MV(u), MV(v))$.

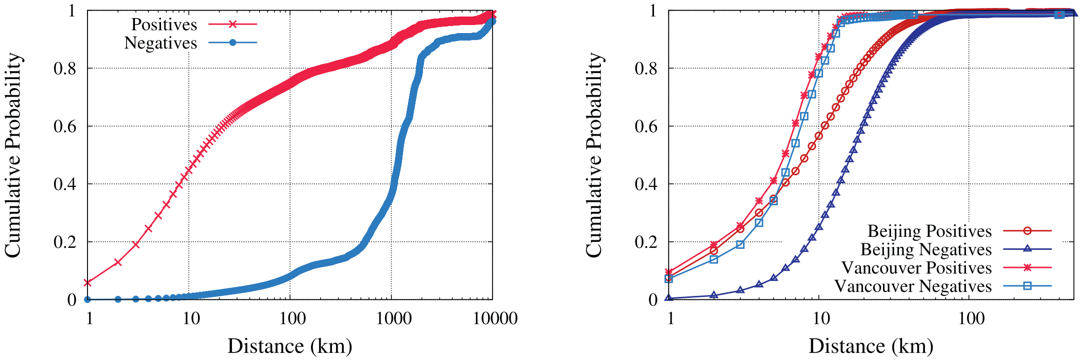
Fig. 10 shows CDFs of the spatial distance between the most frequently visited locations for sets of positives and negatives in Beijing, Vancouver and worldwide. In Fig. 10a, we observe a clear distinction between positives and negatives when we consider the global Momo network (note the log-scale in the x-axis). Fig. 10b shows that $D(u,\ v)$ are lower than 10km for majority of positives. Only 20% of negatives in Beijing exhibit such a geographical closeness. Our results for Beijing are to be expected and inline with what has been previously observed in [17] which shows that face to face human contacts and phone contacts are confined within a distance of 5 miles (8 km) and 100 miles (160 km) respectively.

Interestingly, the difference between the two curves is less significant in Vancouver, with almost 60% of the negatives having their distance lower than 10km, due to the smaller geographical area of the city. In Vancouver, a city with much less Momo users, it might be more challenging to rely solely on such a feature to separate positives from negatives.

4.2. Diversity in updates

Considering the spatial usage patterns of users, we focus on the probability of visiting the same locations by the users belonging to the same community (in particular community creation locations). A naive approach to capture this likelihood is to utilise the number of common locations visited by two users $\|\mathbb{L}_u \cap \mathbb{L}_v\|$. In Momo, 40% of the positives share at least one common location suggesting the intuition of co-location similarity between positives. We then compute the Jaccard-index to measure the similarity of the locations patterns visited by a pair of users as $JS(u, v) = \frac{\|\mathbb{L}_u \cap \mathbb{L}_v\|}{\|\mathbb{L}_u \cup \mathbb{L}_v\|}$.

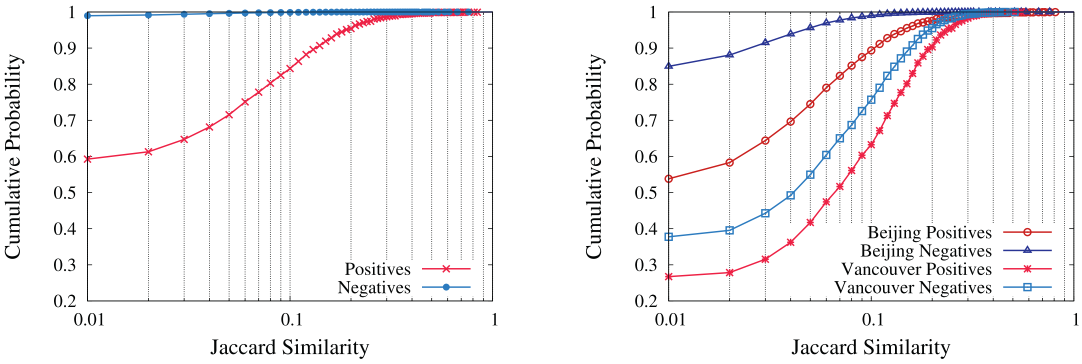
Fig. 11 shows the CDFs of Jaccard similarity between positives and negatives in global Momo network, Beijing and Vancouver. Again, there is a difference in location trajectories of users belonging to the same communities and users outside the respective communities. As can be seen from Fig. 11a, approximately for all the negatives in the global Momo network, similarity score is close to zero. This is expected because the two users to be considered are very likely to be in two countries. According to Fig. 11b, when considering users confined to a city, we still obtain $JS > 0$ for



(a) Global Momo network

(b) Beijing & Vancouver networks

Fig. 10. Distance between the most frequently visited locations $D(u, v)$.



(a) Global Momo network

(b) Beijing & Vancouver networks

Fig. 11. Jaccard Similarity of visited locations $JS(u, v)$.

some negatives. However, the observed similarity values are rather small with approximately 95% of the similarity values for negatives in Beijing being lower than 0.1. Similar to our previous observation for spatial distances, Vancouver users exhibit very high similarity irrespective of the community membership. We further evaluate the efficiency of such a feature in community membership prediction in Section 5.

4.3. Co-location in updates

Jaccard similarity does not capture how frequent two users happen to be in the same location. Intuitively, users belonging to one community may visit the same set of location more frequently than negatives. Here, we define possible co-location metrics to capture different aspects of co-location.

We consider the spatial co-location rate as,

$$SCoL(u, v) = \frac{\sum_{l \in \mathbb{L}} N(u, l) \times N(v, l)}{N_u \times N_v}$$

which represents the probability of users u and v to visit a common location. Since *SCoL* does not take into account the time at which users visit common locations, we define spatial-temporal co-location rate $STCoL(u, v)$ as,

$$STCoL(u, v) = \frac{\sum_{i=1}^{N_u} \sum_{j=1}^{N_v} H(\Delta - |t_u(i) - t_v(j)|)\delta(l_u(i), l_v(j))}{\sum_{i=1}^{N_u} \sum_{j=1}^{N_v} H(\Delta - |t_u(i) - t_v(j)|)}$$

where $H(x) = 1$, if $x > 0$ and $H(x) = 0$ otherwise.

STCoL captures the likelihood that two users have their updates co-located during a Δ time interval. Fig. 12a shows the variation of *STCoL* with different Δ values for positives. We observe that there are a very few users (less than 10%) who have visited the same location within a 1-hour time frame (i.e. $STCoL > 0$). Only 16% of users within the same community having a nonzero co-location rate (*SCoL*), in addition to the essential co-location at the community creation location.

Fig. 12b shows the CDFs of co-location rates of positives and negatives in two different cities, and we note that the difference between the distributions is significantly higher for Beijing users compared to Vancouver positives and negatives. This again suggests that it might be easier to differentiate between community members in larger cities (from the perspective of number of users and density of Momo network.)

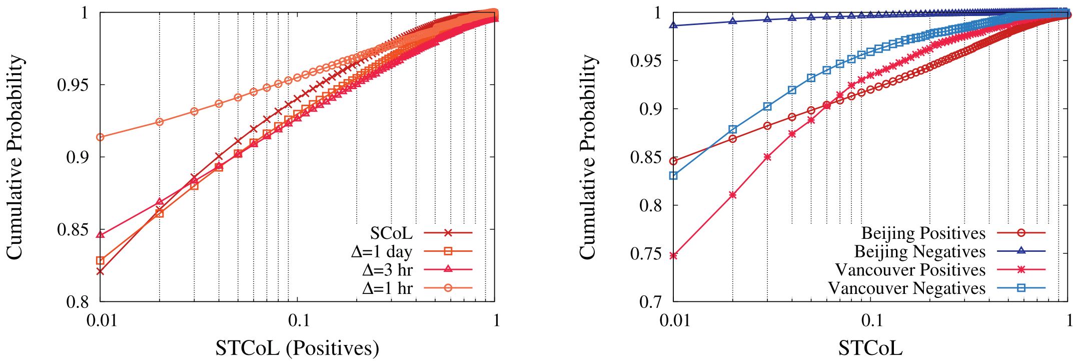
We further investigate possible other metrics and define a spatial-cosine similarity, spatial co-location normalised by the norm of the user trajectories. For each pair of users u and v,

$$SCos(u, v) = \frac{\sum_{l \in \mathbb{L}} N(u, l) \times N(v, l)}{N(u) \times N(v)}$$

where $N(u) = \sqrt{\sum_{l \in \mathbb{L}_u} N(u, l)^2}$.

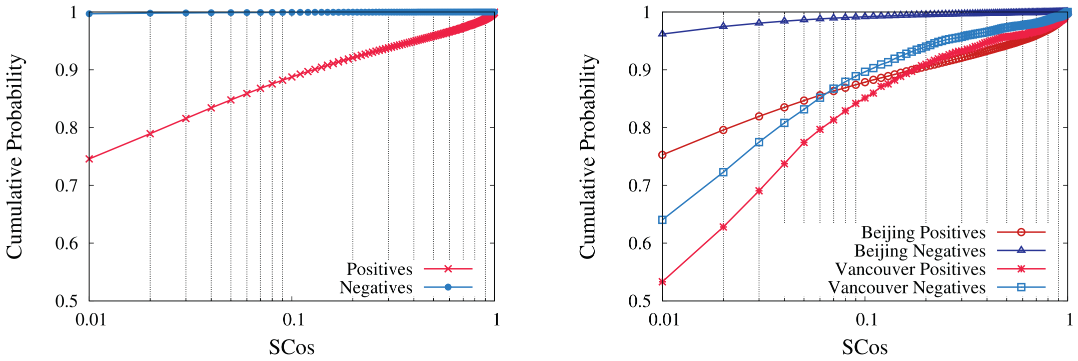
Fig. 13a and b depicts the CDFs of *SCos* values for positives and negatives for global MoMo network, Beijing and Vancouver. While *SCos* values are zero for almost all the negatives for global and Beijing, and although 75% of the positives also have a zero, 25% of them have *SCos* greater than zero increasing the chances of identifying a positive community membership. Vancouver *SCos* has similar pattern to other similarity measures for the city.

On the other hand, the popularity of a location might have an impact on the co-location rates and may drive user behaviour independently from whether or not they belong to the same community. In particular, high popular public locations such as railway stations and shopping malls may attract updates from a wide variety of users. A lesser popular location (e.g. a pub) may only attract users with positive community membership. To capture this, we normalise the co-location rates using the popularity of the location



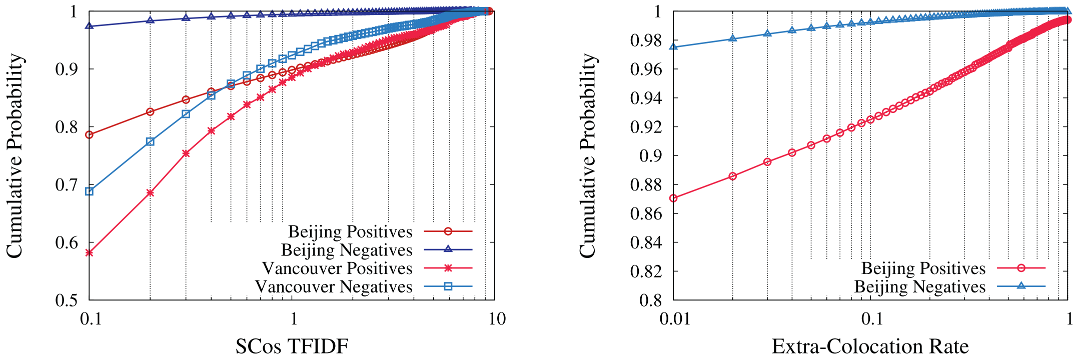
(a) Global positives (b) Beijing & Vancouver

Fig. 12. Spatial-temporal co-location of visited locations *STCoL*(*u*, *v*).



(a) Global (b) Beijing & Vancouver

Fig. 13. Spatial-cosine similarity of visited locations *SCos*(*u*, *v*).



(a) Beijing & Vancouver (b) Beijing

Fig. 14. Spatial-cosine similarity of visited locations *SCos*(*u*, *v*).

using a TF-IDF approach [4]. Specifically, we define location popularity as the inverse document frequency of a location l as follows: $N_{IDF}(l) = \log \frac{\|\mathbb{U}\|}{N(l)}$, where $N(l) = \sum_{u \in \mathbb{U}} H(N(u,l))$. $N(l)$ is the number of distinct users who visited the location l and $\mathbb{U}$ is the set of total users. A high value of $N_{IDF}(l)$ indicates that the location is rather unique to a very few users and conversely, smaller values indicate that the location is visited by a large number of different users. *SCosTFIDF* is the TF-IDF version of *SCos* and is calculated by multiplying *SCos* by $N_{IDF}(l)$. Fig. 14a shows that the difference between positives and negatives is larger in Beijing for *SCosTFIDF* than *SCos*. Vancouver users also have high *SCosTFIDF* similarity values, however the difference between the two distributions are similar to *SCos*.

Diving further into co-location, we capture co-location during office time (defined as between 6:00am to 6:00pm) and personal (nighttime and weekends) time of users. We define extra co-location rate by calculating *STCoL* only for personal time of users. Fig. 14b shows extra co-location for Beijing. The results are counter-intuitive as extra co-location for 87% of positives is zero.

In summary, it has been observed that there are number of mobility based features that can be used to distinguish community members from non-members. In the next Section, we use these features to evaluate the feasibility of community membership prediction among Momo users.

5. Community membership prediction

We model the problem of identifying positive community membership between users as a link prediction problem using mobility based features studied in Section 4. Over 99% of the commu-

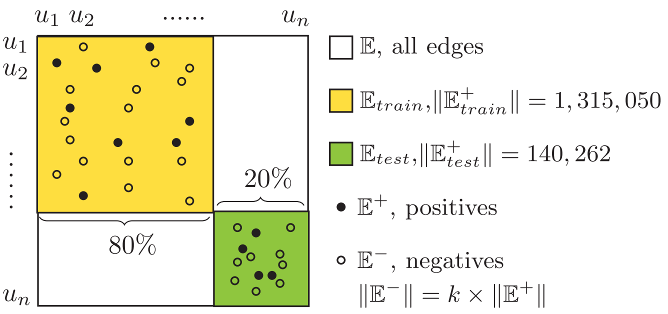

u_1 u_2 ...... u_n
u_1
u_2
u_n
20%
80%
$\mathbb{E}$, all edges
$\mathbb{E}_{train}, \|\mathbb{E}^+_{train}\| = 1,315,050$
$\mathbb{E}_{test}, \|\mathbb{E}^+_{test}\| = 140,262$
● $\mathbb{E}^+$, positives
○ $\mathbb{E}^-$, negatives
$\|\mathbb{E}^-\| = k \times \|\mathbb{E}^+\|$

Fig. 15. Training & test edges from adjacency matrix.

nities in our dataset were created after the end of Updates collection. Therefore, we are predicting community memberships based on user's past mobility data.

5.1. Experimental setup

Fig. 15 illustrates all the users found in our dataset in the form of an adjacency matrix. First, we divide the users in 4:1 ratio to seperate out the users to training and test sets so that 284,393 users are in the training set and 71,097 are in the test set. A positive membership or a positive edge exists between a pair of users if they belong to the same community. Two users not belonging to the same community are said to have a negative membership or a negative edge. When we train the model, we consider the positive and negative edges between all pairs of users that are only in the training set. Similarly, while evaluating the model we restrict ourselves to edges between user pairs that are only in the test set. To predict community memberships, we follow a supervised learning approach, where we train a classifier using the metrics discussed in Section 4 as features for positive and negative edges.

We highlight that the adjacency matrix is highly sparse because the total number of possible negative training edges are $\binom{284,393}{2} - \|\mathbb{E}^+_{train}\| \approx 40\ bn$, i.e. there are only 3 positive relationships for every 10^5 possible user pairs. This not only may lead to immense computational challenges in terms of time and memory, but also creates a huge class imbalance in our classification model. Therefore, we progressively vary the number of negative instances. We randomly generate negative training edges, k times the number of positive training edges and then train the classifier with only $(k+1) \times \|\mathbb{E}^+_{train}\|$ edges. We vary k from 1 to 5 and study the impact of increasing the number of negative samples on the prediction performance. To mitigate the class imbalance problem, we also provide our classifier with a cost sensitivity matrix that weighs an error made on miss-classifying a true positive, k times the error made on miss-classifying a true negative.

We show that it is possible to reconstruct the community membership network of given users with high accuracy without any knowledge of the existing community membership of the user. Unlike several of the previous work in link prediction [4,7,18,19], we consider the more challenging case of not taking any existing links between training and test users (unshaded portions of Fig. 15) for training the classifier as this will artificially increase the accuracy of the prediction since some of the test users are already used for training. In particular, for training the classifier, we consider only the cases where both users of an edge belong to the training set which is shown in the $\mathbb{E}_{train}$ portion of Fig. 15.

5.2. Evaluation

We first evaluate the performance of the similarity metrics discussed in the previous section. In a simple model, we can set a threshold value for each metric to maximise the model's accuracy and predict whether or not a test edge is positive based on the threshold. Table 5 summarises the performance of each of the features for the whole Momo network. To measure the efficiency of such a simple classifier, we use traditional precision/recall metrics. We also calculate F-measure, which is the harmonic mean of precision and recall. Area under the ROC curve (AUC) is another way of measuring the prediction performance, where instead of fixing a threshold, we vary it and calculate the area under the corresponding ROC curve. From Table 5, we observe that while it is possible to achieve high precision from the individual measures, recall values remain significantly low. The *distance* metric exhibits a higher recall value at the expanse of a very low precision.

Table 5
Individual feature performance.

Feature	Accuracy	Precision	Recall	F-measure	AUC
Distance	0.804	0.069	**0.986**	**0.131**	**0.789**
SCos	**0.840**	0.982	0.042	0.080	0.700
SCos-tfidf	0.835	**0.992**	0.012	0.025	0.700
STCoL	0.803	0.984	0.016	0.031	0.635
Jaccard	0.834	0.986	0.003	0.006	0.700
Extra-CoL	0.803	0.881	0.025	0.050	0.572

Table 6
Prediction results.

	k	Accuracy	Precision	Recall	F-measure	AUC
Global	1	0.825	**0.911**	**0.719**	**0.804**	0.847
	2	0.863	0.875	0.689	0.771	0.847
	3	0.887	0.833	0.683	0.751	**0.849**
	4	0.901	0.811	0.657	0.726	0.848
	5	**0.911**	0.784	0.643	0.707	**0.849**
Beijing	1	0.693	**0.860**	**0.552**	**0.696**	0.756
	2	0.756	0.768	0.386	0.514	0.764
	3	0.808	0.763	0.338	0.469	**0.767**
	4	0.826	0.671	0.253	0.368	0.759
	5	**0.849**	0.608	0.272	0.376	0.757
Vancou.	1	0.738	0.887	**0.547**	**0.677**	**0.776**
	2	0.806	**0.900**	0.471	0.618	0.749
	3	0.846	0.893	0.436	0.586	0.744
	4	0.869	0.873	0.401	0.550	0.745
	5	**0.887**	0.877	0.372	0.522	0.742

We then move to combine the above features and train a classifier to perform the prediction. We use Vowpal Wabbit,[9] as it employs a limited memory variation of standard BFGS algorithm to efficiently learn on very large sparse datasets. We build our training model with all $e \in \mathbb{E}^+_{train}$, considering increasing subsamples of negative edges from $\mathbb{E}^-_{train}$ and all features shown in Table 5. We also use the same proportion of positives and negatives in the test set. Table 6 summarises the performance of our prediction model for the complete dataset.

Combining all features yields to significant improvement in the classifier performance as indicated by the improved accuracy, F-measure and AUC. Precision has slightly reduced compared to individual feature performance, while recall has significantly improved. Without any prior knowledge of user's demographics or social graph information, the results show that a major portion of Momo's communities can be rebuilt with 91.1% of precision and 71.9% recall. Increasing the size of negatives in the training set, decreases the precision-recall values (while improving accuracy). This is to be expected as we increase the number of negatives keeping the number of true positives the same. Therefore, the likelihood to link the positives decreases.

Learning from a smaller social graph makes the classification problem more challenging as the similarity in features between

[9] http://hunch.net/~vw/.

positives and negatives increases. When model is trained from users of a city only, we could still achieve a precision rate of 86% in Beijing and 90% in Vancouver which is slightly lower than global results, as shown in Table 6. In essence, we show solely based on partial and sparse mobility data of an LBSD network, it is still possible to reconstruct the community graph with a high precision.

6. Related work

There has been an extensive research work on **online communities** characterisation and detection spanning a number of fields of study. McAuley et al. [20] analyse social circles in Google+, Facebook and Twitter and develop a statistical model to detect user circles from the network's structural properties and user profile information. Yang et al. [21] consider various interest-based online communities such as YouTube, LiveJournal and Friendster for the purpose of detecting online communities by leveraging the underlying online friendship network. Traud et al. [22] propose assortativity coefficient based on user attributes as an effective graph-metric in detecting communities using Facebook data collected from 100 American universities. More recently, using Facebook and Twitter data Dunbar et al. [23] show that online social network communities show the same layered structure as found in offline networks.

In contrast to online communities, LBSD services like Momo, allow users to discover and join communities whose members are nearby. This makes LBSD communities quite unique, and different from other communities in online social networks previously analysed in the literature [20–23]. To the best of our knowledge, this is the first study of LBSD communities, characterising user behaviour in such communities and showing peculiar features of the network graph along with the differences to online communities.

Most of the research related to our work has focused on LBSNs with an emphasis on **user behaviour analysis**. Li and Chen [3] provide a quantitative analysis of Brightkite by classifying users into different behavioural groups based on their in-app usage, mobility patterns and online social relationship. Pelechrinis et al. [24] study the LBSN Gowalla by defining user similarity metrics that account for the entropy of the visited locations. Cheng et al. [25] also analyse Gowalla by augmenting their data with twitter's geo-tagged status updates to show that user trajectories follow a levi-flight pattern. Wang et al. [26] propose a framework to identify overlapping communities in Foursquare using the user-location check-in graph and the attributes of the users and venues. Similarly, Lim et al. [27] detect communities in LBSNs by augmenting the friendship graph of Foursquare with spatio-temporal links in the likes of being in a common-location and applying standard community detection algorithms such as LabelProp [28] and InfoMap [29].

Our work also studies to which extent **sparse location data** extracted from in-app usage can be used in detecting communities. User mobility patterns have been well studied [30–33] to demonstrate the predictability of user mobility and its correlation with individual's attributes and behaviour. Similarly, the likelihood of online social ties based on the geographical distances and co-occurrences has also been studied for geo-tagged Flickr photographs [7], Facebook [19], Gowalla and Brightkite [2]. Another aspect of user mobility patterns involves identifying groups of users who are travelling together [34,35]. For instance, Sen et al. [34] train a Support Vector Machine classifier to identify user groups who are walking together inside a shopping mall based on data collected from their smartphones.

Research work on **social link prediction** include [4] and [18] which propose a number of network proximity measures and evaluate the benefits of different metrics to predict social relationship. The evaluation is based on a Call Detail Record (CDR) dataset obtained from a cellular operator and updates data from the LBSN Gowalla. Duan et al. [36] propose an ensemble enabled approach for the social link prediction problem that scales efficiently for larger networks and measure its performance in number of social network datasets such as YouTube, Flickr, and Wikipedia. More recently, Tang et al. [37] explored the problem of negative link prediction in social networks which allows to define friends as well as foes.

The different nature of LBSD data (human driven updates related to activities in mobile social networking or in cellular networks versus physical closeness between users when joining communities) necessitates a specific study of this environment. Chen et al. in [14] studies user activity patterns in Momo and shows that user re-identification is possible by analysing spatial-temporal characteristics of the user mobility patterns. As opposed to this work, we focus our research on the characterisation of the location-based communities in the Momo network. In addition, we propose a community membership prediction machine learning approach based on spatial and temporal similarity features observed in user mobility data, e.g. by observing different user mobility patterns within and outside a community. We further explore the mobility traces to evaluate the potential of content dissemination leveraging on user mobility. We also note that [14] studied the Momo network before the community membership feature of the system was introduced. Xue et al. [13] study the behaviour of the WeChat users who are using the *People Nearby* feature in terms of anonymity and demographics and show that users tend to be anonymous when using LBSNs and more male users tend to use LBSN services compared to female users. In another work, Xue et al. [38] highlight the inherent privacy risks associated with LBSNs that share users location as bands with other users and propose counter measures to alleviate this problem. In contrast, our work in this paper focuses on studying the user behaviour of communities in LBSNs and predicting common community memberships.

7. Discussion and concluding remarks

Using a unique dataset of a popular mobile application, we provided a first in-depth analysis of user behaviour and communities in LBSD networks. We analysed the evolution of more than 176K communities and characterised spatial and temporal factors that affect the rate of community creation such as time of the day and the high influence of geographical location on user's choice when joining communities. More specifically, our analysis showed that LBSD users are more active on night times (9:00 pm – 12:00 midnight) compared to day time. Also, it was found that more LBSD activities are happening on weekdays compared to weekends. Many online services utilize on cloud storage and processing services such as Amazon and rely on dynamic resource allocations. This temporal usage statistics will provide insights to efficiently allocate cloud resources for LBSD services and thereby reduce the operating cost for LBSD service providers. Also it was found that majority of the locations in the network are either hot-spots or cold-spots in related to community activities and hot-spot are more likely to attract more communities showing a *"rich-get-richer"* phenomenon. Such spatial usage insights help other service providers who provide services on top of LBSDs such as advertisers to make better decisions about their service offerings. Moreover, these rich-locations are the most effective places to deploy technologies for ad-hoc local communication and content sharing.

We model the Momo network as a tripartite graph with different node types for *locations, users, and communities* and studied the structural properties of the graph. We found that due to limited amount of community memberships that are available for users they tend to join only the communities in the local neighbourhood. As a result, users in smaller cities are more connected compared to users in large cities, indicating any value added service developed on top of LBSDs will have a greater impact on smaller cities. We also investigated how the Momo LBSD network can be leveraged in related to an opportunistic content distribution application where content can be cached in either locations or users in order to propagate to the rest of the users in the network. Our analysis showed that, degree (popularity) based selection performs better in location selection for content caching, while random selection is as effective as the degree in selecting users for content caching. These results provide insights to design improved content delivery protocols and content recommendation models by only using sparse location information of users without compromising user privacy by collecting their trajectory information.

Afterwards, we analysed the similarity in user mobility patterns and showed that there is a difference in spatial and temporal behaviour between Momo community members and non-members. Moreover, it is possible to reconstruct these communities accurately by predicting community memberships, solely based on mobility based features. The results of our supervised learning classifier show that it is possible to predict 82% of the possible community memberships between users with a precision up to 91% and recall of 72% in global Momo network. Furthermore, the classification problem becomes more challenging for smaller geographical areas such as small cities due to the observed high similarity of user mobility patterns.

We envision several possible applications and future work of our findings. In particular, our results demonstrate the high potential for service providers who may monitor and retain data on user's activities to infer real-life social relationships. For instance, advertisement networks can leverage location information to link users and learn shared interests without access to any other permission or user-provided explicit data. In one hand, this suggests that sharing location information can potentially lead to massive privacy risk of the user. On the other hand, this shows the potential to enable personalised services such as targeted advertisements and personalised movie recommendations without collecting personal information. In our future work, we aim to investigate on exploiting these insights in relation to such applications.

References

[1] S. Scellato, A. Noulas, R. Lambiotte, C. Mascolo, Socio-spatial properties of online location-based social networks, in: Proceedings of the fifth International AAAI Conference on Weblogs and Social Media (ICWSM), 2011.

[2] E. Cho, S.A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: Proceedings of the 17th ACM SIGKDD, ACM, 2011, pp. 1082–1090.

[3] N. Li, G. Chen, Analysis of a location-based social network, in: Computational Science and Engineering, 2009. CSE'09. International Conference on, 4, IEEE, 2009, pp. 263–270.

[4] D. Wang, D. Pedreschi, C. Song, F. Giannotti, A.-L. Barabasi, Human mobility, social ties, and link prediction, in: Proceedings of the 17th ACM SIGKDD, ACM, 2011, pp. 1100–1108.

[5] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, B.Y. Zhao, Whispers in the dark: analysis of an anonymous social network, in: Proceedings of the 2014 Conference on Internet Measurement Conference, ACM, 2014, pp. 137–150.

[6] C.L. Nemelka, C.L. Ballard, K. Liu, M. Xue, K.W. Ross, You can yak but you can't hide, in: Proceedings of the 2015 ACM on Conference on Online Social Networks, ACM, 2015. 99–99

[7] D.J. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, J. Kleinberg, Inferring social ties from geographic coincidences, Proc. Natl. Acad. Sci. 107 (52) (2010) 22436–22441.

[8] J. Cranshaw, E. Toch, J. Hong, A. Kittur, N. Sadeh, Bridging the gap between physical location and online social networks, in: Proceedings of the 12th ACM Ubicomp, ACM, 2010, pp. 119–128.

[9] Sweetr URL http://www.sweetr.com.

[10] Yik Yak URL https://www.yikyak.com.

[11] Techcrunch, Facebook launches "nearby friends" with opt-in real-time location sharing to help you meet up, 2014

[12] Techcrunch, Yik Yak, 2016

[13] M. Xue, L. Yang, K.W. Ross, H. Qian, Characterizing user behaviors in location-based find-and-flirt services: anonymity and demographics, Peer-to-Peer Networking Appl. (2016) 1–11. http://link.springer.com/article/10.1007/s12083-016-0444-5.

[14] T. Chen, M.A. Kaafar, R. Boreli, The where and when of finding new friends: analysis of a location-based social discovery network, in: Seventh International AAAI Conference on Weblogs and Social Media, 2013.

[15] E. Agrell, T. Eriksson, A. Vardy, K. Zeger, Closest point search in lattices, Inf. Theory IEEE Trans. 48 (8) (2002) 2201–2214.

[16] M.E. Newman, Mixing patterns in networks, Phys. Rev. E 67 (2) (2003) 026126.

[17] D. Mok, B. Wellman, J. Carrasco, Does distance matter in the age of the internet? Urban Stud. 47 (13) (2010) 2747–2783.

[18] S. Scellato, A. Noulas, C. Mascolo, Exploiting place features in link prediction on location-based social networks, in: Proceedings of the 17th ACM SIGKDD, ACM, 2011, pp. 1046–1054.

[19] L. Backstrom, J. Leskovec, Supervised random walks: predicting and recommending links in social networks, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, 2011, pp. 635–644.

[20] J. McAuley, J. Leskovec, Learning to discover social circles in ego networks, in: Advances in Neural Information Processing Systems, 1, 2012, pp. 539–547.

[21] J. Yang, J. Leskovec, Defining and evaluating network communities based on ground-truth, in: Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, ACM, 2012, p. 3.

[22] A.L. Traud, P.J. Mucha, M.A. Porter, Social structure of facebook networks, Physica A 391 (16) (2012) 4165–4180.

[23] R. Dunbar, V. Arnaboldi, M. Conti, A. Passarella, The structure of online social networks mirrors those in the offline world, Soc. Netw. 43 (2015) 39–47.

[24] K. Pelechrinis, P. Krishnamurthy, Location affiliation networks: bonding social and spatial information, in: Machine Learning and Knowledge Discovery in Databases, Springer, 2012, pp. 531–547.

[25] Z. Cheng, J. Caverlee, K. Lee, D.Z. Sui, Exploring millions of footprints in location sharing services., in: Fifth International AAAI Conference on Weblogs and Social Media, 2011.

[26] Z. Wang, D. Zhang, X. Zhou, D. Yang, Z. Yu, Z. Yu, Discovering and profiling overlapping communities in location-based social networks, Syst. Man Cybern. IEEE Trans. 44 (4) (2014) 499–509.

[27] K.H. Lim, J. Chan, C. Leckie, S. Karunasekera, Detecting location-centric communities using social-spatial links with temporal constraints, in: European Conference on Information Retrieval, Springer, 2015, pp. 489–494.

[28] U.N. Raghavan, R. Albert, S. Kumara, Near linear time algorithm to detect community structures in large-scale networks, Phys.Rev. E 76 (3) (2007) 036106.

[29] M. Rosvall, C.T. Bergstrom, Maps of random walks on complex networks reveal community structure, Proc. Natl. Acad. Sci. 105 (4) (2008) 1118–1123.

[30] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (2010) 1018–1021.

[31] M.C. Gonzalez, C.A. Hidalgo, A.-L. Barabasi, Understanding individual human mobility patterns, Nature 453 (7196) (2008) 779–782.

[32] T. Pontes, G. Magno, M. Vasconcelos, A. Gupta, J. Almeida, P. Kumaraguru, V. Almeida, Beware of what you share: Inferring home location in social networks, in: Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on, IEEE, 2012, pp. 571–578.

[33] J. Krumm, Inference attacks on location tracks, in: Pervasive Computing, Springer, 2007, pp. 127–143.

[34] R. Sen, Y. Lee, K. Jayarajah, A. Misra, R.K. Balan, Grumon: fast and accurate group monitoring for heterogeneous urban spaces, in: Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems, ACM, 2014, pp. 46–60.

[35] H. Du, Z. Yu, F. Yi, Z. Wang, Q. Han, B. Guo, Group mobility classification and structure recognition using mobile devices, in: 2016 IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, 2016, pp. 1–9.

[36] L. Duan, C. Aggarwal, S. Ma, R. Hu, J. Huai, Scaling up link prediction with ensembles, in: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, ACM, 2016, pp. 367–376.

[37] J. Tang, S. Chang, C. Aggarwal, H. Liu, Negative link prediction in social media, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, ACM, 2015, pp. 87–96.

[38] M. Xue, Y. Liu, K.W. Ross, H. Qian, Thwarting location privacy protection in location-based social discovery services, Secur. Commun. Netw. 9 (2016) 1496–1508.