Event-based Social Networks: Linking the Online and Offline Social Worlds

Xingjie Liu[⋆], Qi He[†], Yuanyuan Tian[†], Wang-Chien Lee[⋆], John McPherson[†], Jiawei Han[⋄]
[⋆] The Pennsylvania State University, [†] IBM Almaden Research Center,
[⋄] University of Illinois at Urbana-Champaign
[⋆] {xzl106, wlee}@cse.psu.edu, [†] {heq, ytian, jmcphers}@us.ibm.com,[⋄] hanj@cs.uiuc.edu

ABSTRACT

Newly emerged event-based online social services, such as Meetup and Plancast, have experienced increased popularity and rapid growth. From these services, we observed a new type of social network – *event-based social network* (EBSN). An EBSN does not only contain online social interactions as in other conventional online social networks, but also includes valuable *offline* social interactions captured in offline activities. By analyzing real data collected from Meetup, we investigated EBSN properties and discovered many unique and interesting characteristics, such as heavy-tailed degree distributions and strong locality of social interactions.

We subsequently studied the heterogeneous nature (co-existence of both online and offline social interactions) of EBSNs on two challenging problems: community detection and information flow. We found that communities detected in EBSNs are more cohesive than those in other types of social networks (e.g. location-based social networks). In the context of information flow, we studied the event recommendation problem. By experimenting various information diffusion patterns, we found that a community-based diffusion model that takes into account of both online and offline interactions provides the best prediction power.

This paper is the first research to study EBSNs at scale and paves the way for future studies on this new type of social network. A sample dataset of this study can be downloaded from `http://www.largenetwork.org/ebsn`.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software - *Information networks*

General Terms

Algorithms, Experimentation.

Keywords

Event based Social Networks, Social Network Analysis, Social Event Recommendation, Online and Offline Social Behaviors, Heterogeneous Network

KDD'12, August 12–16, 2012, Beijing, China.

1. INTRODUCTION

Newly emerged event-based online social services, such as Meetup (`www.meetup.com`), Plancast (`www.plancast.com`), Yahoo! Upcoming (`upcoming.yahoo.com`) and Eventbrite (`www.eventbrite.com`) have provided convenient online platforms for people to create, distribute and organize social events. On these web services, people may propose social events, ranging from informal get-togethers (e.g. movie night and dining out) to formal activities (e.g. technical conferences and business meetings). In addition to supporting typical online social networking facilities (e.g. sharing comments and photos), these event-based services also promote face-to-face *offline* social interactions. To date, many of these services have attracted a huge number of users and have been experiencing rapid business growth. For example, Meetup has 9.5 million active users, creating 280,000 social events every month; Plancast has over 100,000 registered users and over 230,000 visits per month.

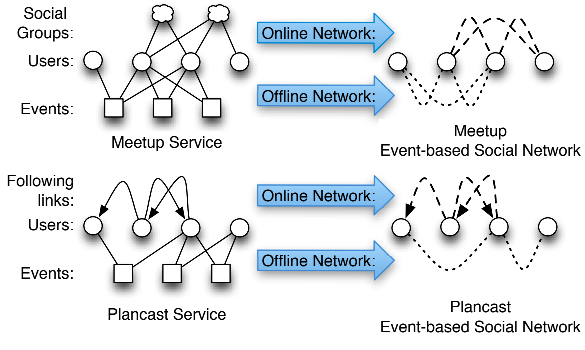


Figure 1: Event-based Social Network Examples

As these event-based services continue to expand, we identify a new type of social network – *event-based social network* (EBSN) – emerging from them. Like conventional online social networks, EBSNs provide an online virtual world where users exchange thoughts and share experiences. But what distinguishes EBSNs from conventional social networks is that EBSNs also capture the face-to-face social interactions in participating events in the offline physical world. Fig. 1 depicts two example EBSNs from Meetup and Plancast. In Meetup, users may share comments, photos and event plans with members in the same online social groups (e.g. "bay area photographers", "Nevada county walkers"). In Plancast, users may directly "follow" others' event calendars. Bi-directional co-memberships of online social groups in Meetup or uni-directional subscriptions in Plancast ultimately constitute an *online social network* represented as the dashed lines on the right side of Fig. 1. Meanwhile, in both cases, users' co-participations of the same events derive their offline social connections. These connections collectively form an *offline social network* denoted as dotted lines in Fig. 1. The online and offline social interactions jointly define an EBSN.

Recent location-based online social networking services, such as Foursquare (`foursquare.com`) and Gowalla (`gowalla.com`), represent another type of popular social network, called a location-based social network (LBSN). They are somewhat similar to EBSNs, as they capture online social interactions as well as offline location checkins. However, unlike the offline social events that incur a group of people with social interactions, location checkins from LBSNs mostly represent individual behaviors, i.e. a particular user was at a specific location at a specific time. Although in [5], adjacent checkins were treated as one kind of reason for social network tie creation. It is estimated that adjacent checkins have only a 24% chance to lead to a new social friendship in Gowalla. Therefore, in this paper, we only compare EBSNs against the online social networks in LBSNs.

To the best of our knowledge, this paper is the first work to identify an event-based social network as a co-existence of both online and offline social interactions, and comprehensively study its properties. Our study revealed the many aspects of EBSNs that are significantly different from conventional social networks. As to be shown in our analysis, social events present very regular temporal and spatial patterns. In addition, both online and offline social interactions in EBSNs are extremely local. For example, we found that 70.65% of Meetup online friends and 84.61% of Meetup offline friends live within 10 miles of each other. To our surprise, the degree distributions of the Meetup EBSN do not follow the usual power law distribution, but are more heavy-tailed than power law. Furthermore, we found that the online and offline social interactions in an EBSN are positively correlated, implying a synergistic relationship between the two parts.

Community structure detection is a very useful approach for analyzing social networks. However, to correctly detect communities in an EBSN, one has to consider both online and offline social interactions. In this paper, we employ an extended Fiedler method to incorporate this heterogeneity during the community detection process. Through experiments, we demonstrate the advantage of this method to other approaches. We also observed that the detected communities in the Meetup EBSN are more cohesive than those of the Gowalla LBSN.

To further investigate information flow over EBSNs, we also study the problem of event participation recommendation. Due to the short life time of an event, the event participation recommendation problem significantly differs from the usual recommendation problem for movies or places. Recommendation of an event is only valid after the event is created and before the event starts. This leads to a cold-start problem. In this paper, we design a number of diffusion patterns that capture the information flow over the heterogeneous EBSNs. Through experiments we demonstrate that the diffusion pattern that takes the community structures into account yields the best prediction power.

The rest of this paper is organized as follows. We describe the related work in Section 2 and formally define EBSNs in Section 3 . We examine the properties of EBSNs in Section 4 and further investigate the community structures in Section 5. In Section 6, we tackle the event participation prediction problem to study the information flow over EBSNs. Finally, we conclude the paper in Section 7.

2. RELATED WORK

Offline social interactions in the physical world have always been important in sociology [9]. One line of work is to study the origin of social relationships. In [12], Feld proposed a focus theory in which individuals organize their social interactions around foci, such as workplaces, families, etc; whereas [20, 16, 3] utilized affiliation to explain the construction of social connections. Chapter 4 of [11] provides a nice summary on these topics. Under the above theories, social events can be viewed as one type of focus or affiliation that creates the social interactions between participants.

Thanks to the popularity of event-based social network services, such as Meetup and Plancast, we are now able to get our hands on large scale social data with rich information on both online activities and offline social events. In [24], Sander and Seminar attended 40 social events in Meetup and concluded that participants in Meetup social events have social structures instead of just strangers meeting strangers.

Similar to event-based social networks, location-based social networks also contains "online" social interactions and "offline" checkin information. Although adjacent location checkins may indicate implicit social interactions and social ties [5], checkins are usually sporadic [21] and largely represent individual behaviors. The geographical features of users were also examined to infer social ties in [7, 26]. In comparison to these work, the "offline" information (social events) studied in this paper does not only contain location, but also time and people involved.

3. EVENT-BASED SOCIAL NETWORKS

In this section, motivated by popular event-based social services, we define event-based social networks and describe how to construct the networks from collected datasets.

3.1 Event-based social services

As various online social networking services become prevalent, a new type of event-based social service has emerged. These web services help users to create social event proposals, disseminate the proposals to related people, and keep track of all participants. To foster efficient communication and sharing, these event-based services also provide online social networking platforms to connect users with others with similar interests. Below, we describe two examples of such event-based social services: Meetup and Plancast.

Meetup is an online social event service that helps people publish and participate in social events. On Meetup, a social event is created by a user by specifying when, where and what the event is. Then, the created social event is made available to selected users or public, controlled by the event creators. Other users may express their intent to join the event by RSVP ("yes", "no" or "maybe") online. To facilitate online interactions, meetup.com also allows users to form social groups (e.g. "bay area single moms", "Nevada county walkers") to share comments, photos and event plans.

Similar to meetup.com, Plancast is another web service that helps users create and organize events online. Users also RSVP to express their intent to join social events. In

Meetup		Gowalla	
# Users	5,153,886	# Users	565,642
# Events	5,183,840	# Locations	2,838,143
# RSVPs	42,733,136	# Checkins	36,804,656
# Groups	97,587	# Social links	2,431,625
# Memberships	10,704,068		

Table 1: Dataset Statistics

contract to Meetup which adopts social groups to connect users online, Plancast allows users to "follow" others' social event calendars to establish online connections.

3.2 Event-based Social Networks Definition

Based on the event-based social services described above, we formulate a new type of social network, called an event-based social network (EBSN).

Like any social network, EBSNs capture social interactions among users. However, different from others, ESBNs incorporate two forms of social interactions: *online social interactions* and *offline social interactions.*

Online social interactions. In EBSNs, users can interact with each other online without the need of physical contact. For example, people can share thoughts and experiences with those in the same social group in Meetup. In Plancast, user comments and event plans are pushed to those who "follow" the user.

Offline social interactions. Social events play a major role in ESBNs. In a social event, people physically get together at a specific time and location, and do something together. Therefore, the social events in EBSNs represent the offline social interactions among event participants.

Definition: Formally, we define an EBSN as a heterogeneous network $G = \langle U, A^{\text{on}}, A^{\text{off}} \rangle$, where U represents the set of users (vertices) with $|U| = n$, A^{on} stands for the set of online social interactions (arcs), and A^{off} denotes the set of offline social interactions (arcs). The online social interactions of an EBSN form an online social network $G^{\text{on}} = \langle U, A^{\text{on}} \rangle$, and the offline interactions of an EBSN compose an offline social network $G^{\text{off}} = \langle U, A^{\text{off}} \rangle$. □

Note that the online social network or the offline social network of a EBSN can be either directed or undirected. For simplicity, we only focus on undirected online and offline networks in this paper.

The online social network [1, 18] or the offline social network [2, 22] alone is not new and has been studied extensively before. But the co-existence of both is what makes EBSNs special. As shown later in this paper, these two forms of social networks in EBSNs are intertwined but also have their own distinct characteristics at the same time.

3.3 Representative Datasets Description

To effectively study EBSNs and explore the unique properties against related LBSNs, we collected data from the popular event-based web services Meetup and the popular location-based social service Gowalla. In this section, we introduce the basic dataset statistics, as well as how EBSN and LBSN are established from these datasets.

Meetup EBSN. We crawled meetup.com from Oct 2011 to Jan 2012. The collected data statistics are shown in Table 1. With the Meetup dataset, the online EBSN is constructed by capturing the co-membership of online social groups: users u_i and u_j are connected in the online social network G^{on} if they are members of the same social group. Let g_r denote a group with $|g_r|$ members, then $(u_i, u_j) \in A^{\text{on}}$ if and only if $\exists g_r$ such that $u_i \in g_r$ and $u_j \in g_r$. We consider users of a smaller group more closely connected than those of a larger group. Therefore, we adopt a similar approach as in [19] to define the edge weights:

$$w_{i,j}^{\text{on}} = \sum_{\forall g_k, u_i \in g_k \wedge u_j \in g_k} \frac{1}{|g_k|}. \qquad (1)$$

The offline social network of the EBSN, G^{off}, is constructed in a similar way based on the co-participation of social events: user u_i and u_j are connected if they co-participated in the same social event. If we use e_k to represent a social event with $|e_k|$ participants, and $u_i \in e_k$ to denote the fact that u_i participated e_k, then the weight of the offline social interaction between u_i and u_j is defined as

$$w_{i,j}^{\text{off}} = \sum_{\forall e_k, u_i \in e_k \wedge u_j \in e_k} \frac{1}{|e_k|}. \qquad (2)$$

Gowalla LBSN. Gowalla is a popular online location-based social networking service that allows individual user to "checkin" their current locations (as well as comments/photos) and share with their friends. Gowalla requires users to explicitly specify their friends. Users need to mutually accept each other as friends to establish an online social link.

We crawled Gowalla from Sep 2011 to Nov 2011 and collected a subset of the users' online social networks and place checkins. The total numbers of users and locations are also summarized in Table 1. As discussed before, although this LBSN provides offline location checkins, these checkins cannot directly form an offline social network. Thus, the Gowalla LBSN only has an online social network in this study.

4. PROPERTIES OF EBSNS

In this section, we analyze the Meetup dataset to highlight the unique properties of EBSNs. As social events play a central role in EBSNs, we first study those properties specifically associated with social events. Then, we examine the network properties of EBSNs.

4.1 Social Events

Social events provide a platform for users to get-together physically. A social event is characterized by two major features: *event time* and *event location.* First, we observe that social events exhibit regular temporal patterns. Fig. 2 depicts the social event time pattern on weekly scale. It is clear that in every weekday there is a small spike around 2pm in the afternoon, followed by a higher spike at 8pm in the evening. On weekends, events distribute relatively even throughout the day.

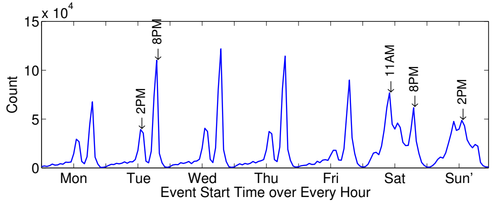


Figure 2: Social event time histogram over every hour of one week.

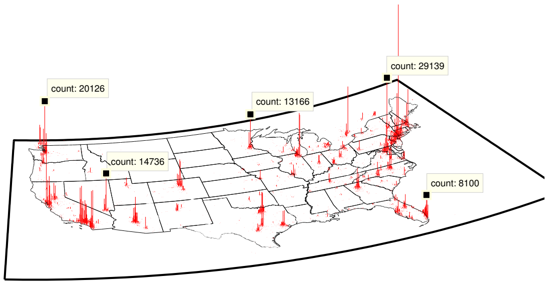


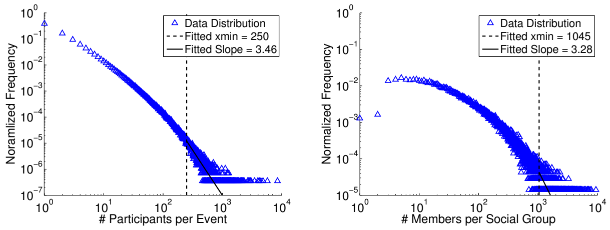
Figure 3: Social event geographical histogram. Each bar represents the number of social events in 100 square miles.

We also observe that social events are mainly located in urban areas. Fig. 3 depicts a US event geographical histogram with 100 square miles as a geographical unit.

4.2 Event and Group Participation

To understand the basic network properties of the Meetup EBSN, we need to first study the event participation and group membership in Meetup. As shown in Fig. 4(a), most of the events are small with just a few participants, but big events with a large number of participants (the heavy tail) do exist in a non-trivial quantity. Similarly, Fig. 4(b) shows that large groups do have significant presence. We examine how these two distributions fit the power law curve by Kolmogorov-Smirnov test [6]. This approach estimates the following 3 parameters:

- xmin: the best fitted cutoff value so that *only* values larger than *xmin* fit a power-law distribution;
- $\hat{\alpha}$: the slope of the best fitted power-law distribution so that values larger than *xmin* follow distribution $x^{-\hat{\alpha}}$;
- p-value: the statistical significance of the goodness of the power-law fitting, (p-value larger than 0.1 suggests a significant good fit).



(a) # participants per event (b) # members per group

Figure 4: Histogram of the number of participants per event and number of members per group.

By estimating the above parameters, we find that only after xmin= 250 does the event size follow a power-law distribution with a high statistical significance (with p-value 0.357). Similarly, the number of members per group follows a power-law distribution non-significantly with $\hat{\alpha} = 3.28$ only after the number of events is greater than 1045 (with p-value 0.088). These two results suggest that although most events and social groups are in small scale, large events and large groups do show significant presence in the Meetup dataset.

4.3 Network Properties

Now we study the network properties of the Meetup ESBN by comparing it against the Gowalla LBSN. Table 2 lists some network properties of the Meetup EBSN online social network G^{on}, offline social network G^{off}, combined network G as well as the Gowalla LBSN social network. First, it can be clearly seen that the EBSN online social network is much denser than the EBSN offline social network, (larger strongly connected component SCC, higher clustering coefficient and lower average degree of separation). This is due to the fact that a user connects to more people online than in actual social events. Secondly, all three EBSN social networks (G^{on}, G^{off} and G) are much denser than the Gowalla LBSN, because Meetup users interact with each other by co-joining social groups or co-participating social events whereas Gowalla users have to mutually establish friendships to get connected.

	Meetup EBSN			Gowalla LBSN
	G^{on}	G^{off}	G	
Mean Degree	1,786.1	140.7	1,560.6	10.64
Median Degree	623	40	463	3
SCC. Ratio	0.999	0.993	0.997	0.987
Clustering Coef.	0.438	0.267	0.429	0.137
Degree Separation	3.00	4.25	3.07	4.47
Degree Fitted xmin	3,765	536	7,490	47
Degree Fitted $\hat{\alpha}$	2.49	2.53	2.50	2.53
Degree Fitting p-value	0.000	0.000	0.000	0.124

Table 2: Network statistics comparison between EBSN and LBSN.

To dig deeper into the network properties of EBSN, we first study the degree distributions in Fig. 5. Again, we apply the Kolmogorov-Smirnov statistic to examine whether these distributions fit the power law distribution. The estimated parameters are listed in the bottom of Table 2. While the Gowalla LBSN conforms to the power law distribution, all three of the EBSN forms are more heavy-tailed than power law. This heavy tail phenomenon in the Meetup EBSN is correlated with the significant presence of big events and big social groups found in Section 4.2.

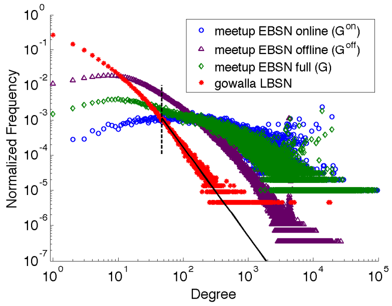
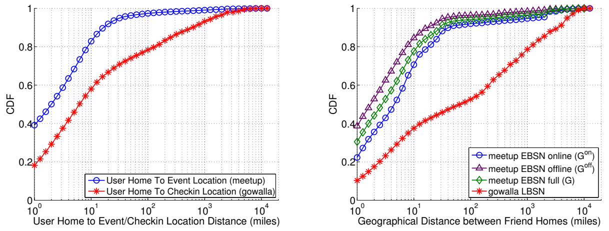


Figure 5: Degree distribution comparison between EBSN and LBSN.

Next, we analyze the correlation between each user's online interactions and offline interactions. By applying Pearson correlation, we observe positive correlation between online and offline degrees (0.368) as well as between online and offline cluster coefficients (0.393). This implies that the online social network and the offline social network work together synergistically in the Meetup EBSN – each have a positive effect on the other.

4.4 Locality of Social Interactions



(a) locality of events (b) locality of friends

Figure 6: Localities of Meetup EBSN and Gowalla LBSN.

In the following, we further analyze on the geographic aspects of social interactions. In Fig. 6(a), we examine the distance of a Meetup event location and a Gowalla checkin location to the user's home location [4, 5]. As illustrated by this figure, although both events and checkins tend to be local to users' home locations, the possibility of an event participation in Meetup decreases more dramatically as the distance increases. As observed, 81.93% of events participated in by a user are within 10 miles of his/her home location. This indicates that people's social activities are much more location constrained than place checkins. This is because people's checkins are usually sporadic [21] and largely represent individual behaviors. Social events, which need all participants to meet at the same spot, must be located close to all the participants in most cases.

Next we compare the distances between friends' home locations in the Meetup EBSN against the Gowalla LBSN. As depicted in Fig. 6(b), friends in Meetup, no matter in online, offline, or the combined social networks, are much geographically closer to each other than in Gowalla LBSN. This is because both online and offline social networks in Meetup EBSN revolve around social events, which require participants to physically get together at the same location. In comparison, it is perfectly fine and usual for a Gowalla user to share a location checkin when he/she visits some new places. Not surprisingly, offline friends in Meetup EBSN tend to live closer to each other than the online friends. 84.61% of offline friends live within 10 miles to each other.

5. EBSNS COMMUNITY STRUCTURE

In this section, we investigate the community structures of EBSNs. Due to the heterogeneity of EBSNs, communities are defined by both online and offline interactions [1]. As a result, previous community detection algorithms on homogeneous networks do not directly apply to EBSNs. Thus, we employ an extended Fiedler method to detect communities in EBSNs and compare it against the previous approaches. We also use the Gowalla LBSN as a comparison to further study the unique features of the Meetup EBSN.

5.1 Clustering on Homogeneous Networks

For homogeneous social networks like the online or offline network of an EBSN, we use the popular Fiedler method offered by the Graclus tool [10] to partition networks. The partitioned clusters are treated as user communities. Let A define the adjacency matrix of a network. The popular Normalized Cut (NCut) [27] shown in Eq. 3 is applied as the graph partition objective function for each binary cut.

$$\min \frac{\mathbf{y}^T L \mathbf{y}}{\mathbf{y}^T D \mathbf{y}}, \quad \textit{subject to}\ \mathbf{y}^T D \mathbf{1} = 0, \mathbf{y} \neq 0. \tag{3}$$

In Eq. 3, D is the diagonal matrix in which each diagonal value is the sum of the corresponding row ($D_{ii} = \sum_j A_{ij}$), $L = D - A$ is the Laplacian matrix, $\mathbf{y}$ is the column vector with $y_i \in \{1, -b\}$ and b is some data-dependent constant. The column vector $\mathbf{y}$ represents the graph cutting results of the current binary cut, since all nodes with $\mathbf{y}_i = 1$ are clustered into one cluster and the other nodes with $\mathbf{y}_i = -b$ are clustered into another cluster. If $\mathbf{y}$ is relaxed to take on real values, Eq. 3 is equivalent to solving the generalized eigenvalue system $L\mathbf{y} = \lambda D\mathbf{y}$, where $\mathbf{y}$ is the Fiedler vector corresponding to the second smallest eigenvalue.

5.2 Clustering on Heterogeneous EBSNs

5.2.1 *Baseline 1: Linear Combination*

Given an EBSN G, we have two separate but correlated networks $G^{\text{on}} = \langle U, A^{\text{on}} \rangle$ and $G^{\text{off}} = \langle U, A^{\text{off}} \rangle$. Both G^{on} and G^{off} share the same user set U. As a result, the clustering process should consider the correlation between G^{on} and G^{off}. The simplest way to leverage both online and offline social interactions is to combine them linearly

$$A = \gamma * A^{\text{on}} + (1 - \gamma) * A^{\text{off}}. \tag{4}$$

Here A defines a linearly combined adjacency matrix with a weighting parameter γ to differentiate two types of interactions. We name this naive method as *LinearComb* and use it as a baseline for comparison. The major problem of *LinearComb* is that after the linear combination, the social interaction type information is missing in the new matrix A.

5.2.2 *Baseline 2: Generalized SVD*

As another baseline, we utilize Generalized Singular Vector Decomposition (GSVD) to incorporate online and offline social interactions in the clustering process by following Theorem 5.1.

THEOREM 5.1. *Given two EBSN social interaction matrices $A^{on} \in R^{n \times n}$ and $A^{off} \in R^{n \times n}$, there exists unitary matrics $\mu, \nu \in R^{n \times n}$, reversible matrix $Y \in R^{n \times n}$ and rectangular diagonal matrices Σ_1 and Σ_2 such that:*

$$A^{on} = \mu \Sigma_1 Y^T, \quad A^{off} = Y \Sigma_2 \nu^T.$$

The proof of Theorem 5.1 can be found in [14]. In Theorem 5.1, the singular vectors of matrix Y (from the second columns and onwards) collectively offer a consistent clustering on users by leveraging both online and offline social interactions. In this method, the singular vectors of the 2^{nd} to m^{th} smallest singular values are used as $m - 1$ dimensional indicator vectors for users. Then, a classic K-means algorithm is conducted on this space to generate user communities. We name this method *GSVD*.

One shortcoming of *GSVD* is that as Y is not a unitary matrix, its values on different column vectors vary a lot in ranges. Therefore, the partitioning information embedded in Y cannot be simply differentiated by the symbol sign as the classic SVD does. In experiments, we also found that the performance of *GSVD* is rather sensitive to the choice

[1] Although a *group* or an *event* in Meetup somewhat captures the behaviors of a set of users either online or offline, it is the combination of online and offline interactions that defines a community in EBSNs.

Algorithm 1: HeteroClu

```
Input: EBSN $G = \langle U, A^{\text{on}}, A^{\text{off}} \rangle$, # clusters $K$
Output: User cluster set $C$
Initialize $C = \{C_1, C_2, \ldots, C_n\}$, where each $C_i = \{u_i\}$;
Initialize normalized weights
    $\bar{w}_{ij} \leftarrow (\sum_{u_a \in C_i, u_b \in C_j} w_{ab})/(|C_i| \cdot |C_j|)$ for connected $C_i, C_j$;
while $|C| > M$ do                                /* bottom-up cluster */
    Find the largest $\bar{w}_{ij}$;
    Merge $C_i$ and $C_j$, update related normalized weights;
while $|C| < K$ do                                /* top-down partition */
    Binary cut all $M$ clusters following the objective Eq. 5;
    if $C_i$ is the cluster with the minimum cut cost then
        delete $C_i$ from $C$;
        Add spitted parts of $C_i$ into $C$;
return $C$
```

of similarity measures on the singular vectors of Y. After many comparions, we chose the *city block* similarity measure for *GSVD*.

5.2.3 Extended Fiedler Method

We now propose an algorithm that clusters online and offline interactions at the same time. This algorithm employs the following objective function based on normalized cut (Eq. 3):

$$\min\ \alpha \frac{\mathbf{y}^T (D^{\text{on}} - A^{\text{on}})\mathbf{y}}{\mathbf{y}^T D^{\text{on}} \mathbf{y}} + (1-\alpha)\frac{\mathbf{y}^T (D^{\text{off}} - A^{\text{off}})\mathbf{y}}{\mathbf{y}^T D^{\text{off}} \mathbf{y}}, \qquad (5)$$
$$\textit{subject to } \mathbf{y}^T D^{\text{on}} \mathbf{1} = 0, \mathbf{y}^T D^{\text{off}} \mathbf{1} = 0, \mathbf{y} \neq 0.$$

The above objective function contains two parts, each part alone is a normalized cut objective function on individual online or offline social networks. But the linear combination of both defines a global optimization over the heterogeneous EBSN. Coupling factor α is used to weigh the importance of each network. Note that each part is a normalized value between 0 and 1. Therefore, the size of the individual online or offline network is not captured in Eq. 5. A naive way to assign the importance of the two parts is to set $\alpha = 0.5$. However, since online and offline networks have different network density, we set α as $\frac{sum(A^{\text{on}})}{sum(A^{\text{on}}) + sum(A^{\text{off}})}$.

Similar objective functions to Eq. 5 have been used in the high-order co-clustering problem on multiple types of heterogeneous objects [13]. Solving the new objective function (Eq. 5) is non-trivial, as it represents a typical quadratic fractional programming problem. In [13], the similar function was first approximated to be a quadratically constrained quadratic programming problem by fixing two denominators of the function as constants. Then, the standard semidefinite programming is applied to compute $\mathbf{y}$ efficiently.

In this paper, we use a heuristic algorithm shown in Algorithm 1 to solve the clustering problem with the objective function defined in Eq. 5. This algorithm first employs a bottom-up clustering algorithm on the linear combination of online and offline social networks as defined in Eq. 4, to generate M ($M << K$) giant loose clusters in a bottom-up fashion. This step defines a local greedy merge procedure. Then it uses the top-down recursive binary cut procedure to cut large clusters to smaller ones until K clusters are achieved. This step defines a global recursive cut procedure.

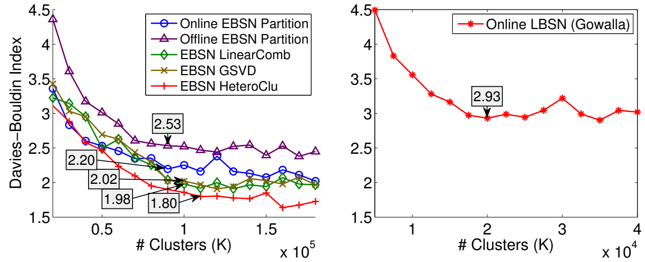


Figure 7: Community dectection performance. The score inside the grey rectangle is the DB index under the optimal K based on the "knee" method.

5.3 Community Structure Evaluation

5.3.1 Evaluation Settings

To measure the quality of user communities, we use the collected user tags as the external ground truth of latent community semantics. 78,158 unique user tags were collected from Meetup and treated as the Meetup tag space T with $|T| = m$. For each user u_i, we built a binary user-tag vector $\mathbf{u_i} = \{t_{i1}, t_{i2}, \ldots, t_{im}\}$ where $t_{ik} = 1$ if u_i selects the tag t_k; otherwise $t_{ik} = 0$. After normalization, the similarity between two users u_i and u_j is measured by the cosine similarity $\mathbf{u_i} \cdot \mathbf{u_j}$. There are no user tags available in Gowalla. Instead, we aggregated all location tags of a user's checkins to build the user-tag vector, in which t_{ik} is the number of checkins associated to tag t_k of user u_i. In total, 680 unique tags were collected in Gowalla.

The standard Davies-Bouldin (DB) index [8] was used to measure the cohesiveness of communities, which is given by

$$DB = \frac{1}{K} \sum_{k=1}^{K} \max_{k \neq j} \left(\frac{2 - \sigma_k - \sigma_j}{1 - \mathbf{c_k} \cdot \mathbf{c_j}} \right), \qquad (6)$$

where K is the number of communities, $\mathbf{c_k} = 1/|C_k| \sum_{u_i \in C_k} \mathbf{u_i}$ is the centroid vector of cluster C_k after renormalization, and $\sigma_k = 1/|C_k| \sum_{u_i \in C_k} \mathbf{u_i} \cdot \mathbf{c_k}$ is the average similarity of users in cluster C_k to their centroid. A smaller DB index value indicates a more cohesive community.

5.3.2 Results

Determining the optimal K for a clustering has been an open problem for decades. For a fair comparison on various approaches and datasets, we used a simple yet popular method that identifies the "knee" [15] in the plot of DB index vs. K to determine the optimal K for each clustering first; and then compare the corresponding DB index under the optimal K. The DB index value corresponding to the "knee" can be seen as the best clustering performance that one method can achieve.

Fig. 7 compares the best DB index of each method based on the "knee" method. Note that since the DB index averages over all the worst separated clustering pairs, it is possible that the DB index has a value greater than 2.

As shown in Figure 7, the communities for the Meetup EBSN are more cohesive than those for Gowalla LBSN. One interesting finding is that users in online Meetup EBSN communities are more cohesive than users in offline Meetup EBSN communities (by 0.33), indicating that users tend to have more similar interests if they belong to same groups, compared to those who participated similar events. However, the combination of online and offline interactions does play an important role in the clustering process, as three methods *LinearCom*, *GSVD* and *HeteroClu* outperformed individual networks. The *LinearCom* is only slightly better than individual networks (by 0.18) but worse than *HeteroClu* (by 0.22), indicating that a simple linear combination cannot differentiate heterogeneous types of social interactions effectively. The *GSVD* has almost the same performance as *LinearCom*, suggesting that after relaxing the constraint on the unitary matrix of SVD decomposition, the generalized SVD lost some disambiguation power on clustering. Lastly, *HeteroClu* leads the pack in comparisons. It is the only method that achieved the best DB index (around 1.8) *sufficiently* under 2, indicating that its worst pairs of clusters were reasonably separated.

6. EBSNS INFORMATION FLOW

In this section, we study how information flows over this unique network structure. A good scenario that can be used to examine the information flow on EBSNs is the problem of recommending users to participate in social events *only* based on the topological structure of EBSNs. With this application, we can study how information flows from one user to the online/offline friends and how the information flow pathways latently drive the social event participation process.

Unlike classic movie/book recommendations, event participation recommendation is more challenging due to the short life time of social events. An event is non-existent until its creation time t_c. And after the start time t_s of an event, participation recommendation becomes meaningless. Due to the very limited history of an event from time t_c to t_s, event participation recommendation suffers from the cold-start problem heavily.

Now, let's formally define the event participation problem as follows: given an event e, at time t ($t_c < t < t_s$), the task is to predict users who will RSVP "yes" to event e between t and t_s. The EBSN built upon the collective data before t will serve as the network structure and all the users who responded "yes" to e between t_c and t are the positive training examples for the prediction, notated as set S [2].

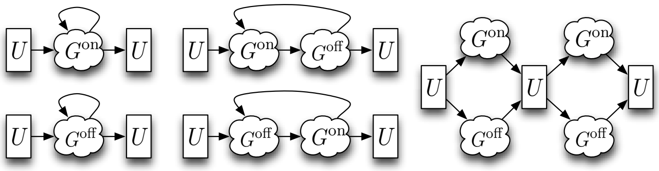
6.1 Event-Centric Diffusion

Not to deviate from our goal of studying the information flow over the EBSNs' unique network structures, we only rely on the topological structure of EBSNs and the already responded users for event participation prediction.

6.1.1 Basic Event-Centric Diffusion

We design a simple yet efficient event-centric diffusion model for the problem. We define $f_i \geq 0$ as the initial score of node u_i, where only users in set S (the set of users already RSVPed "yes") have $f > 0$ and the rest of the users have $f = 0$. For simplicity, we initialize $f = 1/|S|$ for users in S. We use the column vector $\mathbf{v}^k = \{v_1^k, v_2^k, \ldots, v_n^k\}$ to represent the probabilities that users have been visited after the k-th diffusion step, and $v_i^0 = f_i$.

The basic event-centric diffusion, named *DIF*, can be expressed as $\mathbf{v}^{k+1} = \mathcal{D} \cdot \mathbf{v}^k$, where $\mathcal{D}$ defines the non-symmetric information transition matrix of a network for time t. Each element in $\mathcal{D}$ is defined as $d_{ij} = \frac{w_{ij}}{\sum_l w_{il}}$. If we run the model on the heterogeneous EBSN, we can use the linearly combined adjacency matrix (Eq. 4). d_{ij} is the empirical probability of information flow from user u_i to user u_j. Clearly, $d_{ij} \neq d_{ji}$. If u_i has a larger degree than u_j, the influence of u_i on u_j is less than that of u_j on u_i.



(1) single channel (2) cascaded channels (3) paralleled channels

Figure 8: Typical EBSN information flow patterns.

This basic diffusion model is event-centric because $\mathbf{v}^k$ represents personalized probabilities only corresponding to the current event e. A similar diffusion method has also been studied by [17] for link prediction. Because this diffusion process does not converge to the stationary distribution of information flow, a self-loop on every node is necessary; otherwise the information will be diverged far away quickly. The self-loop weight follows the same definitions of Eq. 1 and Eq. 2.

6.1.2 Diffusion over EBSNs

An EBSN contains both online and offline social interactions, but the basic diffusion model *DIF* does not take this heterogeneity into account. Accommodating different forms of social interactions, there exist at least three information flow patterns, as shown in Figure 8. The online and offline social networks G^{on} and G^{off} of an EBSN basically defines two kinds of channels for the flow of information. Figure 8(1) depicts the basic diffusion model *DIF* over a single channel exclusively, whereas Figure 8(2) define a cascade model, abbreviated as *DIF-cascade*, in which information interchangeably flows from one channel to the other. The simplest cascade diffusion model can be defined as $\mathbf{v}^{k+1} = \mathcal{D}_c \cdot \mathbf{v}^k$, where $\mathcal{D}_c$ is a cascaded transition matrix for time t, and $\mathcal{D}_c = \mathcal{D}^{\text{on}} \cdot \mathcal{D}^{\text{off}}$ or $\mathcal{D}^{\text{off}} \cdot \mathcal{D}^{\text{on}}$. Finally, in Figure 8(3), information flows over two channels concurrently. We call this model *DIF-parallel*. The simplest parallel diffusion model is $\mathbf{v}^{k+1} = \mathcal{D}_p \cdot \mathbf{v}^k$, where $\mathcal{D}_p$ defines a linearly combined transition matrix for time t, and $\mathcal{D}_p = \gamma\mathcal{D}^{\text{on}} + (1-\gamma)\mathcal{D}^{\text{off}}$. The parameter γ is used to measure the importance of each type of social interactions. It plays the same role of γ in Eq. 4. Thus, *DIF-parallel* is equivalent to *DIF* on the linearly combined adjacency matrix (Eq. 4). Undoubtedly, there are more complex information diffusion processes (i.e., a mixture of *DIF-cascade* and *DIF-parallel*). But we will leave them for future work.

6.1.3 Community-Based Diffusion

Information is often circulated more rapidly inside its own community, especially for those small-scale local communities. As a result, we design a community-based diffusion model in which information tends to, but is not restricted to, flow within the scope of its own community.

Specifically, in this model, $\mathbf{v}^{k+1} = \mathcal{D}_m \cdot \mathbf{v}^k$, where $\mathcal{D}_m$ defines the community-based information transition matrix. Each element of $\mathcal{D}_m$ is defined as

$$d'_{ij} = \begin{cases} \frac{(1-\beta)w_{ij}}{N} & \text{if } u_j \notin C(u_i), \\ \frac{\beta w_{ij}}{N} & \text{if } u_j \in C(u_i), \end{cases}$$

[2] For simplicity, the event creator is treated as the first user with RSVP "yes".

where $C(u_i)$ is the community of u_i, β is a parameter used to control weight of information flows inside its community versus outside, and N is the normalization factor so that $\sum_j d'ij = 1$. We name this model *DIF-com*.

Since *DIF-com* only adjusts the weights of edges on top of the basic *DIF* model (can be seen as a combination with *DIF*), it can be further combined with other complex diffusion models, including *DIF-cascade* and *DIF-parallel*. The names of the two combinations are *DIF-com-cascade* and *DIF-com-parallel*, respectively. Note that *DIF-com* on G based on the linearly combined adjacency matrix (Eq. 4) is equivalent to *DIF-com-parallel*.

6.2 Information Flow Evaluation

6.2.1 Experimental Settings

As discussed before, event participation recommendation suffer from a typical cold-start problem. When an event is created, except for the creator, it is unknown to all the other users. To simplify the problem, we treat the event creator as the first user who responded "yes" to the event. In evaluation, we can start the recommendation process immediately after the event creation, or wait for a while until there are a few responded users. We first focus on the latter case: given a testing event, we set the first k responded participants as the seed users, where k is randomly determined. The former case is a much harder problem and is examined at the end of the evaluation.

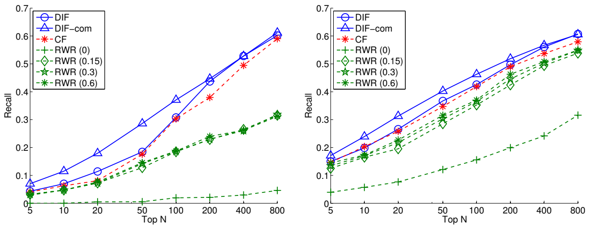
We split the Meetup data into two sequential parts (cut around Mar 2011). The first part of data (on or before Mar 2011, take up 80%) are used for training and the second part of data (after Mar 2012, take up 20%) are used for testing. Given a testing event, we recommend top 5, 10, 20, 50, 100, 200, 400, 800 users to it respectively. We choose to recommend a large number of users, because 1) in practice event organizers often broadly advertise their events to the public; and 2) we want to see the long-term trend of such a recommendation system. For the recommended top N users, we compute *recall* to evaluate the performance. *recall* is defined as the percentage of users who would respond "yes" to the testing event that are covered by the top N recommendations. Finally, we average the *recall* for all testing events under the same top N.

6.2.2 Compare Event-Centric Diffusion Models with Classic Baselines

There are two popular baselines found in the prior art that can be efficiently applied to such an event participation recommendation problem. One is Collaborative Filtering (CF) [25], and the other is the random walk model [23]. Note that due to the extremely short life time of events, most supervised recommendation (link prediction) methods suffer from severe sparsity of labeled data. As a result, they do not apply to the event participation recommendation problem.

For the baseline CF, the users who ever participated in similar groups or events in the Meetup training data are recommendation candidates. They are then ranked by their Jaccard similarities to the responded users. The Jaccard similarity between two users is simply based on their past group or event participation count vectors.

For the baseline random walk model, we applied the *random walk with restart* (RWR) model. In the RWR baseline, there is a certain chance (probability β) with which the information will flow back to the starting users at each step of information flow. By setting various β, we have various RWR baselines with names like RWR (0.3). When $\beta = 0$, RWR downgrades to the basic random walk model.



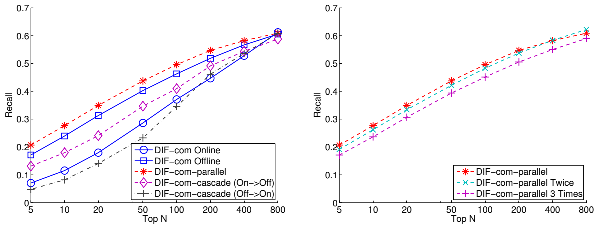
(a) Online EBSN (b) Offline EBSN

Figure 9: Prediction on individual EBSNs.

As both CF and RWR were initially designed for homogeneous networks, we compared them with the basic event-centric diffusion models on individual G^{on} and G^{off} in Fig. 9. From all diffusion models on G^{on} in Fig. 9(a) and G^{off} in Fig. 9(b), *DIF-com* outperforms *DIF* and *CF*, and RWR models perform the worst. By soft-restricting information flow in the same user communities, *DIF-com* can guarantee most closely related friends are recommended. The weighting strategies of *DIF* and *CF* differ only slightly, thus they yield similar prediction results. The poor performance of *RWR* indicates that identified network hubs are not relevant to the testing event. By raising return probabilities of RWR, the prediction performance does not improve much even with β as high as 0.6. In addition, by comparing Fig. 9(a) and Fig. 9(b), we find the offline EBSN has better prediction power when N is small but online EBSN gradually catches up and even surpasses the offline EBSN as N grows large. This is because offline social interactions are able to capture closely related friends who are very likely to participate in the same events, but the recommended users tend to be regulars to similar events. In comparison, online social interaction can introduce non-regulars to the events and increase the coverage of the recommendation.

6.2.3 Compare Various Diffusion Patterns on EBSNs

In the previous section, we showed that *DIF-com* has the best recommendation performance for individual online and offline social networks of an EBSN. As discussed in Section 6.1.3, *DIF-com* actually represents one kind of diffusion pattern on a whole EBSN (equivalent to *DIF-com-parallel* based on the linearly combined adjacency matrix (Eq. 4)). It is thus interesting to further compare various diffusion models we discussed in Section 6.1.2 on the whole EBSNs (with both online and offline social interactions). All diffusion models can be enhanced by communities since *DIF-com* has been shown to outperform the rest of the methods in the previous section. For a fair comparison, we use communities detected by Algorithm 1 for all methods. The detailed comparisons are given by Fig. 10. Fig. 10(a) compares three diffusion models over the heterogeneous EBSNs against individual online/offline networks. Only the paralleled diffusion model outperforms the online or offline only model. This means that the joint presence of online and offline social interactions can improve the prediction performance. The reason that cascade diffusions are worse is because values are diffused twice to those far away users. Similarly, In



(a) EBSN diffusion patterns (b) EBSN recursive diffusion

Figure 10: Prediction on the heterogeneous EBSNs.

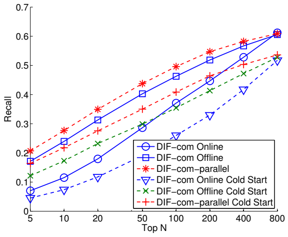


Figure 11: Comparison to cold-start scenarios.

Fig. 10(b), we see that repeating the parallel diffusion model also deteriorates the performance.

6.2.4 *Examine the Effect of Cold-Start*

In this section, we would like to examine how the cold-start phenomena hurts the recommendation performance. It is well-accepted that as the size of responded users decreases, the recommendation performance will get worse. We simply verify this well-known conjecture using Fig. 11. In Fig.11, the prediction performances for those cold start cases (the event creator is the only seed for an event) are slightly worse than random-start cases. However, the recalls achieved by diffusion from a single user are still fairly good, indicating that using diffusion to predict event participation on EBSNs is satisfactory even on the extreme cold start cases.

7. CONCLUSION

In this paper, we have identified and formally defined a new type of social network, EBSN. By using the Meetup dataset, we studied the unique features of EBSNs including basic network properties, community structures and information flow over EBSNs. Our research revealed many aspects of EBSNs that are significantly different from conventional social networks and LBSNs. We hope this paper paves the way for future studies on this interesting type of social networks.

Acknowledgements

We would like to thank Jon Kleinberg for helping us nail down the background of the problem, Bin Gao for his explanation on the related work [13] and Jiang Bian and Mao Ye for their valuable discussions.

8. REFERENCES

[1] Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong. Analysis of topological characteristics of huge online social networking services. In *WWW*, 2007.

[2] P. S. Bearman, J. Moody, and K. Stovel. Chains of Affection: The Structure of Adolescent Romantic and Sexual Networks. *American Journal of Sociology*, 2004.

[3] C. Borgs, J. Chayes, J. Ding, and B. Lucier. The hitchhiker's guide to affiliation networks: A game-theoretic approach. *arXiv:1008.1516v1*, 2010.

[4] Z. Cheng, J. Caverlee, K. Lee, and D. Sui. Exploring millions of footprints in location sharing services. In *ICWSM*, 2011.

[5] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD*, 2011.

[6] A. Clauset, C. Shalizi, and M. Newman. Power-law distributions in empirical data. *Arxiv preprint arxiv:0706.1062*, 2007.

[7] D. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, and J. Kleinberg. Inferring social ties from geographic coincidences. *PNAS*, 2010.

[8] D. Davies and D. Bouldin. A cluster separation measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1979.

[9] I. de Sola Pool Manfred. Contacts and influence. *Social networks*, 1979.

[10] I. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: spectral clustering and normalized cuts. In *KDD*, 2004.

[11] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

[12] S. L. Feld. The focused organization of social ties. *American Journal of Sociology*, 1981.

[13] B. Gao, T.-Y. Liu, X. Zheng, Q.-S. Cheng, and W.-Y. Ma. Consistent bipartite graph co-partitioning for starstructured high-order heterogeneous data co-clustering. In *KDD*, 2005.

[14] G. Golub and C. Loan. *Matrix Computations*. Johns Hopkins Univ. Press, 1996.

[15] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall Prentice-Hall advanced reference series, 1988.

[16] S. Lattanzi and D. Sivakumar. Affiliation networks. In *STOC*, 2009.

[17] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *KDD*, 2010.

[18] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *SIGCOMM*, 2007.

[19] M. Newman. Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality". *Physical Review E*, 2001.

[20] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. In *National Academy of Sciences*, 2002.

[21] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. An empirical study of geographic user activity patterns in foursquare. In *ICWSM*, 2011.

[22] J. F. Padgett and C. K. Ansell. Robust Action and the Rise of the Medici, 1400-1434. *The American Journal of Sociology*, 1993.

[23] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. 1999.

[24] T. Sander and S. Seminar. E-associations? using technology to connect citizens: The case of meetup.com. In *Annual Meeting of the American Political Science Association*, 2005.

[25] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *WWW*, 2001.

[26] S. Scellato, A. Noulas, and C. Mascolo. Exploiting place features in link prediction on location-based social networks. In *KDD*, 2011.

[27] J. Shi and J. Malik. Normalized cuts and image segmentation. *TPAMI*, 2000.