MobiGroup: Enabling Lifecycle Support to Social Activity Organization and Suggestion with Mobile Crowd Sensing

Bin Guo, *Senior Member*, Zhiwen Yu, *Senior Member*, Liming Chen, *Member*, Xingshe Zhou, *Member*, Xiaojuan Ma, *Member*

***Abstract*—**This paper presents a group-aware mobile crowd sensing system called MobiGroup, which supports group activity organization in real world settings. Acknowledging the complexity and diversity of group activities, this paper introduces a formal concept model to characterize group activities and classifies them into four organizational stages. We then present an intelligent approach to support group activity preparation, including a heuristic rule-based mechanism for advertising public activity and a context-based method for private group formation. In addition, we leverage features extracted from both online and offline communities to recommend ongoing events to attendees with different needs. Compared to the baseline method, people preferred public activities suggested by our heuristic rule-based method. Using a dataset collected from 45 participants we found that the context-based approach for private group formation can attain a precision and recall of over 80%, and the usage of spatial-temporal contexts and group computing can have more than a 30% performance improvement over considering the interaction frequency between a user and related groups. A case study revealed that, by extracting the features such as dynamic intimacy and static intimacy, our cross-community approach for ongoing event recommendation can meet different user needs.

***Index Terms*—Mobile crowd sensing, group computing, cross-community sensing and mining, social activity organization.**

I. Introduction

Many technologies facilitate group interaction [1]. For example, group management tools [2-6] can help analyze historical interaction data of online communication (e.g., email, Facebook) and offline co-located social events [7-11]. However, group activities involve more complex processes such as group formation and event publicity in addition to intra-group interaction during the events.

This is a challenging problem. First of all, *different types of group activities may vary in goals, needs, constraints, flows, organizations,* and *interaction patterns*. For instance, some events are *closed* or *private* (e.g., a party). In other cases, activities are *open* to the public. In addition, each organizational stage of an activity may require different technical support. For example, one challenge of *activity preparation* is to locate and invite potential attendees. In contrast, the core issue of *running an activity* is recognizing and monitoring ongoing events. Group would benefit from a conceptual model for automatic processing. Second, *there is a lack of technical infrastructure for group activity logging and mining*. For online communities, social web portals can capture virtual interaction data for further use, such as social tie detection. Data from real-world group activities are harder to obtain, requiring specialized models, methods, and mechanisms. It is more difficult to extract information and infer knowledge from physical group activities, since the data tend to be noisy and incomplete. Third, *activity organization in real-world settings is often influenced by various social and physical contextual factors*, such as user location, activity venue/time, existing participants, and so on.

Mobile crowd sensing (MCS) [12] leverages crowd-contributed data collected via smartphone sensors in the *physical space* as well as mobile social networks in the *cyber space*. MCS has been employed in numerous application areas, yet its use in group activity organization is under-investigated. Our work aims to exploit the cross-space sensing nature of MCS to support the lifecycle of real-world group activities.

In this paper, we present the MobiGroup, a group-aware system that provides assistance throughout various group activity organizational stages. It exploits smartphone sensing to capture online/offline social interactions and empowers group formation and management. We extend [13] by (1) addressing the activity lifecycle in real-world settings; (2) characterizing the complexity and diversity of social activity organizational processes in a formal concept model; and (3) providing intelligent facilitations for social activity preparation. Specifically, our contributions include the following:

- *A generic and multi-viewed group activity model*. The activity model classifies the lifecycle of group activities into four stages. Based on the model, we develop a framework that can adapt supports to the characteristics and organizational stages of a group activity.
- *Context-aware approaches to group activity preparation*. For activities that are open to the *public*, we propose a heuristic rule-based strategy to disseminate information of an activity according to its popularity and group preferences [14]. For *private* activities that often consist of a similar set of participants, we use a social graph model to characterize the closed activity participation network, and develop a context-based group computing method for highly-relevant group recommendation.
- *Cross-community approach to ongoing activity suggestion*. To encourage participation of ongoing events, we propose a mechanism for recommending activities to potentially interested users by extracting static/dynamic interaction features of both online and offline communities [15].

The rest of the paper is organized as follows: Section II discusses related work. Section III proposes the group activity concept model and Section IV presents the framework for group activity organization. We describe our methods for planned

Manuscript received xx; accepted xx. This work was partially supported by the National Basic Research Program of China 973 (No. 2015CB352400), the National Natural Science Foundation of China (No. 61332005, 61373119), the Scientific and Technology New Star of Shaanxi Province (2014KJXX-39).

B. Guo, Z. Yu and X. Zhou are with Northwestern Polytechnical University, Xi’an, China (e-mail: guob@nwpu.edu.cn).

L. Chen is with De Montfort University, UK (e-mail: liming.chen@dmu.ac.uk). X. Ma is with the Hong Kong University of Science and Technology (e-mail: mxj@cse.ust.hk).

activity preparation and ongoing activity recommendation in Section V and VI respectively, and present an evaluation in Section VII. We conclude the paper in Section VIII.

II. Related Work

Groupware refers to software that can help people achieve common goals in collaborative work [1]. Examples include email and group editing/conferencing tools [16, 17]. ContactMap provided an editable group visualization tool to depict personal contacts and groups [6]. SocialFlow [2] displayed social groups mined from email data. Researchers from Google proposed a method that can suggest a recipient group upon email composition [3]. These systems extracted social groups from online interactions. They did not address group activity organization in real-world scenarios.

Mobile group activity augmentation refers to the techniques that aim to augment group activity organization using mobile devices. *Many studies focus on group activity sensing and sharing*. For instance, CenceMe exploited mobile phones to infer people's presence (e.g., dancing at a party with friends) and then shared this information over social network media [7]. Movi collaboratively took photos and shared the social activities through co-located mobile phones [8]. Our work adopts an audio-based smartphone sensing method to recognize social activities. Rather than simply sharing the ongoing activity information with friends online, we take an opportunistic approach to selectively send the message to friends that meet certain criteria (e.g., distance, static/dynamic intimacy). *The participant suggestion for real-world activity organizers is another way to augment group activities*. Flocks [9] was a system that supports dynamic group creation on the basis of user profiles and physical proximity. SOCKER allowed the building of ad hoc groups based on opportunistic data dissemination [10]. These systems mainly grouped people already located nearby and did not to recruit like-minded contacts who are not yet gathered but could be. Furthermore, no existing studies have given a systematic investigation of the generic process of group activity organization. MobiGroup, to the best of our knowledge, provides the first concept model of this field.

(3) Cross-community sensing and mining. People are involved in multiple communities, either online or offline. Different communities are implicitly interlinked [10, 11]. For example, co-location in the real world is echoed in connectivity online [18]. We view online and offline communities as complementary networks and leverage "cross-community sensing and mining (CSM)" techniques [15] to support group activity organization. Several studies explore the integration of features from hybrid communities. For example, Tang *et al.* [19] transferred and integrated knowledge from different social networks for social tie prediction. Our previous work [20] combined pervasive sensing and Web intelligence techniques for social contact management. MobiGroup facilitates group activity organization by considering both online (e.g., friendship, comments) and offline features (e.g., location, ambient sounds).

III. Group Activity Modeling

A. Group Activity Characterization

Group activities can be held in face-to-face or in online manners. In our study group activities are referred to traditional *meeting-based activities*, i.e. *a crowd of people that gather together at a certain time and place for a specific purpose*. We explore the key concepts of group activities in the following.

(1) Group activity type. Based on the organization manner, we can divide group activities into planned and unplanned (or opportunistic) activities [21] (Fig. 1). Planned activities usually have explicit activity initiators, and require participant invitation or activity advertising, such as organizing a workshop/party. Unplanned activities are usually held in an opportunistic manner and do not have a clear preparation process, such as meeting someone in a store. Based on the *openness* of an activity to its participants, we can categorize group activities into public/open activities and private/closed ones [22, 23]. In general, public activities are widely-announced and do not limit participation. The participants of private activities are relatively static (or fixed) and are not recruited through open calls but by personal invitations. For instance, *Bob, a university student, often has his lunch with a similar subset of his contacts, e.g., with B, C, D on one day and C, D on another day.*

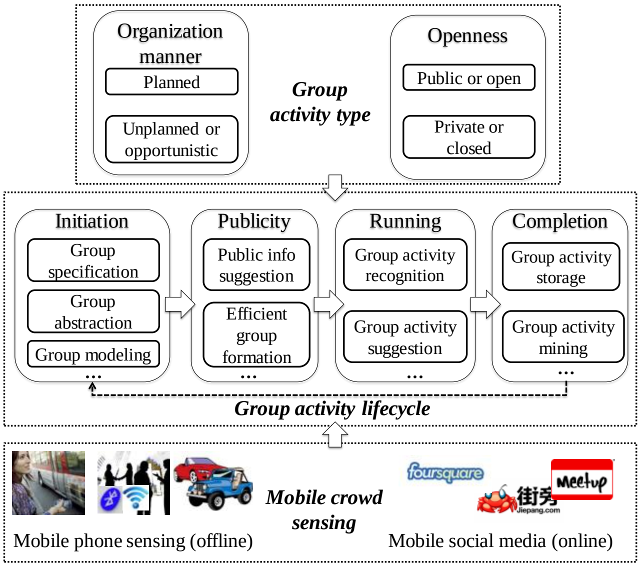


Fig. 1. Conceptual graph of group activity organization.

(2) Activity lifecycle. Organizing a group activity can be characterized as a four-stage lifecycle: activity initiation, publicity, running, and completion. Different technical support should be given to each stage.

- *Initiation and publicity*. These two stages are about activity *preparation*. For public activities, activity advertising and recommendation are critical. For private activities, since the participants are relatively static and fixed, it is desirable to suggest the right members to the private activity initiator.
- *Running*. Unplanned activities usually do not have a preparation process and the running stage is crucial in its lifecycle. Within this stage, the activity participants can share their status online and the system can suggest relevant people to join them. Depending on the situation, participants may prefer to involve different cohorts of contacts based on distinct intimacy metrics, such as close friends, or the ones interacting frequently recently.
- *Completion*. After a group activity completes, its information should be kept in the backend server and used for individual/group pattern learning. The learned information will be helpful to feed the prior three stages.

(3) Activity tags. To facilitate activity information sharing and advertising, each activity instance will be associated with *tags*. We use broadly two types of tags: category and semantic tags.

1) Category tags. Category tags are defined based on the nature and task of activities such as meeting, concert, or party.

2) Semantic tags. In addition to category tags, in information systems we often characterize items by their semantics [24, 25]. We define the following two *semantic tags*.

- ***Hot***. An activity instance's hotness can be measured by the number of people who interact with it (e.g., pressing the 'like' button, reposting messages), and this number usually reaches a peak value within a short period of time.
- ***Social.*** People from *existing groups* usually show high similarity, which is often defined as group-preference [14, 26]. This inspires us to characterize an activity instance at the social structure level. For instance, we can recommend a 'social' activity to an existing online group if a portion of members from this group likes it. Groups can be extracted using community detection methods [27].

'Hot' is a global feature and it indicates the interaction dynamics of the whole community (with loose connections). In contrast, 'social' is a local feature, representing the preferences of a group of highly-connected people.

B. Group Formation

Another important concept pertaining to group activity organization is the formation of groups. We define three types of groups.

- *Activity participation groups* are groups formed by people who participate in activities in the real world. For public activities, *dynamic groups* are usually formed through open calls. For private activities, *static groups* that usually consist of similar sets of social contacts are often formed. The participants of each activity instance form a *raw group*, and *logical groups* can be distilled when studying implicit grouping rules and merging similar raw groups based on historical data. For instance, as presented in Section III.A, the lunch group $\{B, C, D\}$ and $\{C, D\}$ can be merged under certain conditions.
- *Third-party groups*. We can use the social relationship and interaction from other communication mediums. As in [15], an understanding and prediction of group activities can be attained by mining the data from heterogeneous sources from cyber, physical, and social spaces. We thus employ third-party groups that can be detected from online cyber portals, e.g., location-based social networks (LBSNs), to assist real-world activity organization. The analysis of third-party groups can be conducted at two levels of abstraction: *structure level* and the *interaction level* (e.g., comments, likes, retweets). As presented in [23], the social network structure mirrors relatively 'static' user connections. It can be analyzed at either the global-scale (e.g., the whole community [27]) or the local-scale (e.g., ego-network analysis [28]). Interaction-based analysis often reveals dynamic connecting features (e.g., the interaction frequency between two linked users changes over time).
- *Context-based groups*. Context is an important factor for group formation, in particular, in the case of ongoing activities. A widely-acknowledged definition of context is "any information that can be used to characterize the situation of an entity [29]". In this study, the entity is group activity, and all information pertaining to activity such as user location, activity time and venue, social intimacy, user preferences, and behavior similarity, are useful contexts that can impact the formation of activity groups.

Sometimes we need to combine the features of different group types to achieve *hybrid grouping*, e.g., the formation of groups that meet both spatio-temporal contexts and high interaction frequency.

IV. The MobiGroup Architecture

Based on the study of the group activity model, we have developed MobiGroup to address: (1) how to support intelligent advertising of public activities, (2) how to facilitate the suggestion of highly relevant groups to prepare private activities, and (3) how to recommend ongoing activities according to social and physical contexts. The layered architecture of MobiGroup is in Fig. 2.

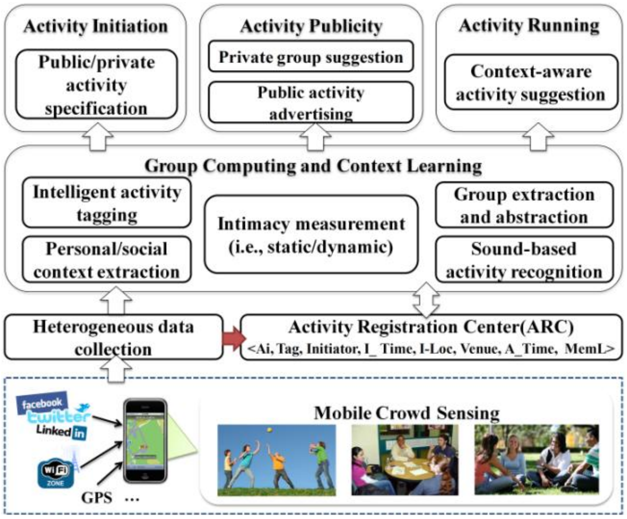

Activity Initiation
Public/private activity specification
Activity Publicity
Private group suggestion
Public activity advertising
Activity Running
Context-aware activity suggestion
Group Computing and Context Learning
Intelligent activity tagging
Personal/social context extraction
Intimacy measurement (i.e., static/dynamic)
Group extraction and abstraction
Sound-based activity recognition
Heterogeneous data collection
Activity Registration Center(ARC)
<Ai, Tag, Initiator, I_ Time, I-Loc, Venue, A_Time, MemL>
facebook
twitter
LinkedIn
WiFi ZONE
GPS ...
Mobile Crowd Sensing

Fig. 2. MobiGroup system architecture.

The first layer is the *mobile crowd sensing* layer, which consists of smartphones enhanced by various sensors (e.g., Bluetooth, WiFi, GPS, accelerometers, microphone) and third-party social network (SN) services.

The second layer is the *data collection and storage* layer which contains two modules: *heterogeneous crowdsourced data collection* and *activity registration center (ARC)*. The former provides MobiGroup with gateways to collect the needed data from either smartphone sensors or third-party services while the latter transforms raw data to social activity logs and inserts them into the ARC repository. The activity logs will be used for historical data-based group extraction and suggestion.

To facilitate group activity organization, we have developed the Activity Registration Model (ARM) for ARC. As shown in Fig. 3, in ARM, each activity instance includes an initiator, the initiation place (*I-Loc*) and time (*I_Time*), the activity venue (*Venue*) and time period (*A_Time*), and a number of activity members or participants (*MemL*). We use an example to illustrate this model: *Bob is in the lab and wants to invite some friends to have dinner together at the Golf restaurant*. Here Bob is the activity initiator (he initiates the activity in his lab), and the activity venue is the Golf restaurant. For unplanned activities that often lack the initiation process, the ARM works as follows: the initiator is by default the current activity sharer and the members are those who participate in the activity after suggestion; the initiation place is the same as the activity venue.

The third layer is the *group computing and context learning* layer that contains the following components. The *intelligent tagging* module is responsible for assigning tags to the published public activities using heuristic rules. The *context extraction* module is in charge of inferring social/personal contexts (e.g., user location, preferences, who is together with) from raw sensory data. The *group extraction and abstraction* module is tasked to extract raw activity groups from activity logging data or third-party groups. In addition, it also distills logical groups from raw groups. The *sound-based activity recognition* module is intended to predict the type of ongoing activities by analyzing the ambient sounds sensed from smartphones. The *intimacy measurement* module offers social structure or interaction based metrics for intimacy calculation.

The fourth layer is the *activity support layer*, which meets the aforementioned needs on group activity organization.

V. Planned Group Activity Preparation

A. Group Activity Initiation

There are two types of group activities in terms of their openness, and we provide different support for them.

(1) Private activity. Users can initiate an activity through its category tag, venue, and time. A category tag can be selected by an initiator from a predefined tag set. To facilitate the selection of a venue, we divide the city map into 100m*100m cells, each can be chosen as the activity venue. A third-party Point of Interest (PoI) dataset such as DBPedia [30] and LBSN check-ins [31] can be leveraged for fine-grained venue setting. As private activities are often associated with static groups, a list of group members for a private activity can be recommended at the initiation time based on ambient contexts. We describe it in detail in Section V.C.

(2) Public activity. To better communicate with the public, we need to learn from existing communication mediums. Posters are a popular and easy-to-use medium for public activity advertising. However, they suffer from problems such as spatio-temporal constraints and low dissemination speed [32]. Using paper posters as the metaphor, we digitize this medium to address its weaknesses. Specifically, when initiating a public activity, people can publish a digital post to MobiGroup for sharing. The metadata (e.g., activity venue and category tag) for the activity post can be specified by the publisher. Users can browse the public activities from a map (e.g., by activity venues). They can interact with the posts and "save" the interested ones that they want to attend (referred to as "savers" to those posts).

B. Publicity Support for Public Activities

The publicity of planned activities is to advertise the activity posts to relevant people within a social community. In terms of the manner activity posts are acquired, there are two generic modes, i.e. *pull* and *push* [33]. *Pull* means that people can browse and query activity posts using pre-specified category tags. By *push*, we compute the semantic tags and automatically circulate the recommended posts to potential interested people. Semantic tags are crowd-related knowledge and need to be learned from large-scale crowd-post interaction data. The methods are presented below.

(1) Public activity sensing. To facilitate public activity publicity, it is important to record crowd interactions with the posts. The following information regarding to the posts are kept.

- ***Savers to a post.*** The people who interact with a post *p* (i.e. the 'savers') are recorded, forming a post group *G*(*p*).
- ***Interaction time.*** It indicates the interaction time of each user to a post, i.e., when the post is saved by the user.
- ***Social links.*** The social links among users are extracted from third-party SN services.

(2) Feature extraction. The following features are extracted from crowd-post interactions for semantic tag prediction.

- ***Post group size.*** It denotes the number of 'savers' to a post (i.e., the post group size), formulated as $|G(p)|$.
- ***Temporal interaction context.*** The popularity of a hot post usually reaches a peak within a short period of time, similar to trending topics in Twitter. We divide one day into *L* time intervals. Let the number of savers to post *p* at the time interval T_i ($0 < i \le L$) of date *d* be $N(p,T_i,d)$ and the average number of saves to any posts at the same time interval during the past few days (we used one week in the current study, because the posts published earlier show little relation with the current post [34]) be $AvgN(T_i)$. As formulated in Eq. (1), we use *II* to measure activity post *p*'s interaction index during T_i. It is a dynamic feature and we use the maximum *II* in the interaction history of *p* as its current hot index *HI*, as formulated in Eq. (2).

$$II\left(p,\ T_i,\ d\right) = N(p,T_i,d)\ /\ AvgN(T_i) \tag{1}$$

$$HI(p) = Max(\{II\left(p,\ T_i,\ d\right) : d_{pub}(p) \le d < d_{Now}; 0 \le i \le L\}) \tag{2}$$

where $d_{pub}(p)$ refers to the publishing date of *p*, *Now* refers to the current time, and d_{Now} returns the date of the current time.

- ***Post group density***. We use the link density [35] to characterize the density of social connections among the savers of a post group, which is the fraction of the existing numbers of connections of savers in the post group against all possible connections. The metric is given in Eq. (3).

$$LinkD(G(p)) = \frac{2 * NumOfConns(G(p))}{|G(p)| * (|G(p)| - 1)} \tag{3}$$

where *NumOfConns* denotes the number of existing connections among the members of the post group *G(p).*

(3) Heuristic-based activity advertising. We use heuristic rules to infer semantic tags. Specifically, we use the group size and the temporal interaction context to characterize a 'hot' post. In the interaction history with an activity post *p*, if $|G(p)|$ is among the top *k*% of all the 'active' posts and *HI* exceeds a threshold Th_{hi} at a certain time, it is tagged 'hot'. A post will expire when it passes the starting point of its *A_Time*, and it is viewed as an active post before expiration, formated by Eq. (4). The whole rule is defined in Eq. (5).

$$Active(p) \leftarrow Now < A\_Time(i) \tag{4}$$

$$\begin{aligned} Hot(p) \leftarrow {} & HI(p) \ge Th_{hi} \wedge \\ & p \in \{i \,|\, Top_k(|G(i)|, \{m \,|\, Active(m)\}) = 1\} \end{aligned} \tag{5}$$

where the *Top* function determines whether $|G(i)|$ is within the top *k%* of all active posts.

We use the density of a post group to determine whether a given activity post should be labeled as 'social'. For example, we check the group density of a certain activity's savers, and if the density value is above a pre-defined threshold Th_{ss}, the post is tagged 'social'. As mentioned in Section III.A, 'social' posts are suggested to existing online group(s) if a proportion of members from an online group save the post. By applying a community detection method over the social network of users (can be obtained from third-party services), implicit groups (forming a group set called *DetGroups*) among them can be obtained. We can locate the group that contains the most people of a post group and recommend the associated post to the other members of this group. We formulate it in Eqs. (6) and (7).

$$G_{social} = MaxGroup(G(p), DetGroups), \quad if\ \ LinkD(G(p)) > T_{ss}. \tag{6}$$

$$Rec(i, p) \leftarrow \forall i \in G_{social}, i \notin G(p) \tag{7}$$

where *MaxGroup* returns the group from *DetGroups* that contains the most people of *G(p)*, and *Rec(i,p)* denotes that *p* should be recommended to user *i*.

C. Group Suggestion for Private Activity Preparation

Private activities usually show regular patterns and relatively fixed group members, based on which we design an approach to supporting private activity group formation. As in Fig. 3, the approach consists of two modules: *offline group computing* and *online group suggestion*. The prior module analyzes the relations among raw groups and merges them into logical groups, which can work offline with historical social activity data. Online group suggestion leverages real-time contexts for group filtering and ranking.

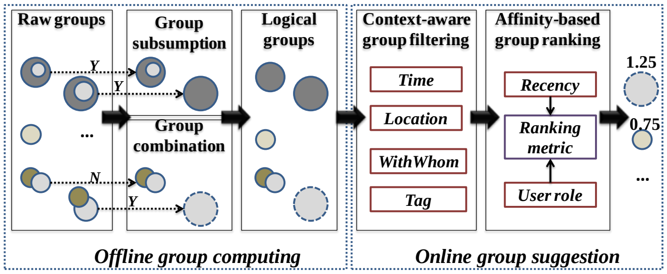


Fig. 3. The approach for private activity preparation.

(1) The social graph model. To study the grouping patterns or the rules of private activities, we first build a model to characterize the network of private activity participation. The social graph model is then proposed. In this model, the edges are formed by initiating or participating activities. We employ the egocentric network method used in [3], in which a message sent by a user to a group of contacts is regarded as one that forms a single edge (a *hyperedge*). The edge is directed, represented as either *in* or *out* edges, corresponding to user-initiated or user-participated activities, respectively. All user-initiated group activities form a set called GA_{out}, while all user participated activities form the GA_{in} set. The nodes to each hyperedge refer to the members of a certain activity and they form a *raw group* (see Section III.B).

Figure 4 is an example of *Bob*'s social graph, where four raw groups are involved (e.g., *G1* to *G4*). For instance, *G1* is formed by an activity which *Bob* initiates and *B-D* are members. *G2* is associated with several activities, where *Bob* is either the initiator (the *out* edge) or a participant (the *in* edge).

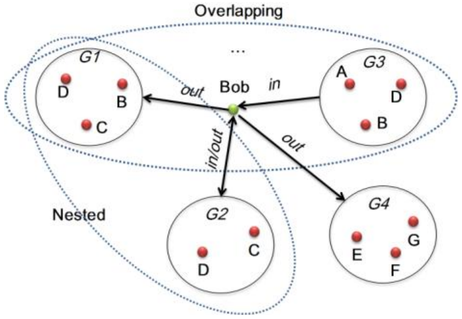


Fig. 4. A group activity graph example.

(2) Private group computing. People can participate in various group activities, and different activities usually link different group instances. This relation results in a large number of groups in the initial social graph. This is a challenge for initiating private group activities, considering that there are many overlapping or nested groups. For instance, *A* can be involved in both a sport team and a project team, and the two groups overlap. This is also obvious from the social graph model. As shown in Fig. 4, the social graph of *Bob* consists of overlapping (e.g., groups *G1* and *G3*) and nested groups (e.g., groups *G1* and *G2*). *G4* is more specific and has little relation with the other three groups. Regarding different themes of private group activities, a user's social ties can be clustered into different sub-communities, which is called the *implicit grouping pattern*. For example, *A* usually has lunch together with *B*, *C* and *D*, except for one day that *B* does not come for some reason; this results in two raw groups: *{B, C, D}* and *{C, D}*. From the implicit grouping pattern perspective, the two group instances should be merged as a logical group: *{B, C, D}*. It is not difficult to derive that recommendations based on raw group instances can result in missings data or errors.

To discover the implicit grouping patterns and improve the performance for private group suggestion, the *group abstraction* concept is used to eliminate minor subsets of groups by merging highly nested or overlapping groups into logical groups. We refer to the merging of nested groups as *group subsumption* and the merging of overlapping groups as *group combination*.

Group subsumption. Given two nested groups to a user *u*, *G1* and *G2* ($G2 \subset G1$). The two groups can be subsumed if they are highly nested. We refer to MacLean *et al.*'s *information leak* metric for group nesting evaluation [2]. The value of information leak is determined by two factors: *similarity of the two groups* and *the ratio of the number of activities held by each group*. We thus define a new parameter *subrate* to measure whether two groups can be subsumed, expressed in Eq. (8).

$$subtrate(G1, G2) = \frac{|G1| - |G2|}{|G1|} \times \frac{num(G2)}{num(G1)}, \quad when\ G2 \subset G1 \tag{8}$$

where $|Gi|$ refers to the number of members of group *Gi*, and $\frac{|G1| - |G2|}{|G1|}$ defines the similarity of two groups; *num(Gi)* refers to the number of activities held by *Gi*. Note that both *in* and *out* relations between *u* and *Gi* are used. Suppose *G1= {B,C,D}*, *G2={B,C}*, and 100 and 5 records are relevant to *G1* and *G2* in

ARC respectively, we have $subtrate(G1,G2)=\frac{3-2}{3}\times\frac{5}{100}=1/60$. If the value is below a predefined threshold (Th_{sub}), the two groups can be subsumed.

Group combination. Two overlapping groups can be combined if they are very similar. To measure the similarity between two groups, we use the *Jaccard* metric, expressed in Eq. (9). The two groups can be combined if their similarity exceeds a threshold (Th_{com}).

$$comrate(G1,G2)=\frac{|G1\cap G2|}{|G1\cup G2|},\ when\ overlap(G1,G2) \tag{9}$$

Group abstraction results in a set of logical groups, which facilitates the management of groups (considering that the number of raw groups can be rather large).

(3) Context-Aware Group Filtering. One basic principle for group cueing is to suggest relevant groups according to user needs. Various contexts can be used to filter out irrelevant logical groups when users initialize activities.

- ***Spatial-temporal contexts***. It includes the location and time context regarding to the activity being initiated. Location can be obtained by *in-phone* GPS positioning or Wi-Fi indoor positioning techniques [36]. The initiation time context is specified as one of the four logical periods, such as *morning* (6:00 to 11:00) and *noon* (11:00 to 13:00).
- ***WithWhom***. Nearby friends who are often co-initiators or members of an activity. We use *WithWhom (i)* to indicate that a number of *i* contacts are together with the initiator. This context can be obtained using the Bluetooth ID of user mobile phones.
- ***Tag***: a category tag specified by the initiator often shows the type of the activity being organized.

The rule for group filtering is performed in this way: for each context C_i that is obtained when a new activity is organized, a logical group G_j is considered irrelevant and thus filtered if it has no historical record that matches C_i.

(4) Group Ranking. After context-aware filtering, there can be still one more group remaining. To suggest the most relevant group to the user, we developed a group affinity ranking method to rank the remaining groups by the strength of the link between a user and the user's logical groups. We employ the method used in [3], which was originally used for link-strength measurement in email networks. The link strength between two entities is computed based on their interaction history.

In MobiGroup, it is measured based on the group activity history between a user and a logical group. In addition to interaction frequency (i.e., the number of associated activities), *recency* and *user role* are also considered in group affinity ranking. Human relationships are dynamic over time, and we use *recency* to denote that the recent data is more important than the old data. For *user role*, the activities in which the user is the initiator (i.e., the *out* relation) are more important than those in which the user is merely a participant (i.e., the *in* relation). We define the affinity rank between user *u* and the logical group G_k as *affRank(u, G_k)*, which can be calculated by Eq. (10). Given *u*, the implicit group with the highest rank will be recommended.

$$affRank(u,G_k)=$$
$$\omega_{out}\sum_{A_i\in GA_{out}\wedge G(A_i)=G_k}\left(\frac{1}{2}\right)^{d_{Now}-d_{pub}(A_i)}+\omega_{in}\sum_{A_i\in GA_{in}\wedge G(A_i)=G_k}\left(\frac{1}{2}\right)^{d_{Now}-d_{pub}(A_i)} \tag{10}$$

where ω_{out} and ω_{in} represent the weights of the user roles in social activities, with the former being larger to represent the importance of initiator roles. We empirically use 1.5 and 1.0 in the current implementation. *$d_{pub}(A_i)$* refers to the initiation date of activity A_i.

We use an example to demonstrate the group affinity ranking method. As shown in Table I, for user *u* and a logical group *G1*, there are three historical activities (*A1*-*A3*) associated with them. We can also find the metadata to each activity, such as user role (*in/out* relation) and the activity date. To illustrate our method, three cases are given. The only difference between case 1 and 2 is that the user role to *A2* changes (from *out* to *in* relation). We find that the rank value of case 1 is higher than that of case 2, which indicates that the *out* relation (i.e., the activity initiator role) is weighted higher in our method. Similarly, the only difference between case 2 and 3 is that the activity date changes. We find that the rank value of case 2 is higher than that of case 3, which reveals the effect of the recency factor.

TABLE I. AN EXAMPLE FOR AFFINITY-BASED GROUP RANKING

Case ID	Associated Activities (*in*/*out*, date)	affRank Value
1	A1(*out*, Nov. 15), A2(*out*, Nov. 16), A3(*in*, Nov. 14)	1.5*(0.5^2+0.5)+ 0.5^3=1.25
2	A1(*out*, Nov. 15), A2(*in*, Nov. 16), A3(*in*, Nov. 14)	1.5*(0.5^2)+ (0.5+0.5^3)=1
3	A1(*out*, Nov. 15), A2(*in*, Nov. 15), A3(*in*, Nov. 14)	1.5*(0.5^2)+ (0.5^2+0.5^3)=0.75

VI. RUNNING ACTIVITY RECOGNITION AND SUGGESTION

In addition to augmenting the preparation process for group activities, we also provide support for running activities. Specifically, if users permit sharing their current activities, MobiGroup can recommend the activities to relevant people based on social/physical contexts. It is particularly helpful for unplanned activities, which do not have the preparation process.

A. Ambient Sound-based Activity Recognition

To suggest a running activity to other people, it is important to first identify the category of the activity. Previous studies on human activity recognition [7, 37] mostly focused on individual activities. Some explore group activity recognition using mainly computer vision techniques [8, 38]. Nevertheless, the vision based method is computationally intensive and intrusive. We have developed a less-intrusive approach to group activity understanding, based on the analysis of sensed ambient sounds from smartphones. There have been studies on ambient sound-based group identification [8, 39] or place categorization [40], and our work is designed for a different purpose.

To reduce the cost when running on resource-constrained devices, a two-stage recognition method is used. Currently our work used the smart campus as the test bed and identified three most common activity types, namely *talk*, *relaxation*, and *other activities*. The relaxation activities are associated with music events (e.g., in a party, in a shopping mall). The data recognized as *other activities* will be further processed in the second stage. We identify three popular activities that university students undertake on a daily basis: street roaming, sporting, and in a quiet environment (e.g., in a library). The recognition algorithm is given below.

(1) Preprocessing. The raw audio data must be preprocessed because it cannot be recognized directly. Ambient sound in the form of audio streams is captured by the mobile phone's microphone sensor. The audio stream data is segmented into frames for feature extraction. The sampling rate is 8KHz, and each frame length is set to 64ms to enable lower duty cycle on the phone. In speech recognition, researchers often set the frame overlap between two continuous frames to enhance smooth analysis. The frame overlap is generally set to 32 or 64ms. As mobile devices have limited resources, we do not set frame overlap in this study. To compensate the high-frequency part that is suppressed during the sound production process, the input signals should be pre-emphasized [41]. Finally, because the change of audio signal in the time domain is difficult to distinguish, the frames are processed using the Fast Fourier Transformation method (FFT) for further analysis.

(2) First-stage activity categorization. According to the requirements of MobiGroup, three features are selected, namely Zero Crossing Rate (ZCR) [42, 43], Spectral Flux (SF) [44], and Haar [45]. Each selected feature is useful for distinguishing the associated acoustic events. For instance, comparing with music, speech signals often show a higher variation in ZCR[43]. Haar-like filtering is traditionally used for image processing, but it is also effective for ambient sound recognition [45]. Based on the features, we build a classifier using the J48 decision tree algorithm, which is efficient when running on resource-constrained devices [45].

(3) Second-stage activity recognition. For the second stage recognition, the goal is to classify the ambient sounds identified as 'other' in the first stage into three activity types. The 24 dimensional MFCC features [41] are used, and the DTW algorithm [46] is applied to classify the three types of activities. MFCC features are more powerful but computationally intensive. We avoid using them over the large number of raw audio clips in the first stage.

B. Context-Aware Running Activity Recommendation

When a user shares her status online, it indicates that she wants to have opportunistic participants to join her activities. The host user may have different preferences or constraints for the participants. As depicted in Section III.A, we attempt to address two varied constraints: one is to invite close friends (the *static intimacy*), and the other is to suggest the ones that have frequent interaction recently (the *dynamic intimacy*). Several other contexts should also be considered, such as preferences of friends, and the distance between the activity venue and the host user's location. This is achieved through the fusion of social interaction data gleaned from online and offline communities. *Online community* data are used to characterize the social intimacy (static or dynamic) between users, while the *offline* data refers to the user contexts in the real world. In other words, we are trying to generate hybrid groups based on online third-party data and associated contexts in real-world scenarios.

(1) Preference. We use $P(u_i, A_j)$ to denote user u_i's preference degree to activity category A_j. It can be calculated based on the user's historical activity records in ARC. Since preferences can change over time, we use the observation data of the last 30 day for preference learning. It is formulated in Eq. (11).

$$P(u_i, A_j) = \frac{NumAct(A_j, u_i, ARC(d))}{NumTotalAct(u_i, ARC(d))} \tag{11}$$

where *ARC(d)* refers to the subset of historical data over the past *d* days, set to 30 in current study; *NumAct* and *NumTotalAct* refer to the number of activity instances a user participates for activity category A_j and the total number of activity instances the user participates, individually.

(2) Distance. It denotes the distance between the venue of an activity and a user. In general, within a certain range of the distance Th_d, the impact of the distance on user's choice is low and the distance factor changes slowly. Beyond the threshold, though, the distance will become more and more important as its value increases. Based on this consideration, we construct the distance parameter *Dis* in Eq. (12):

$$Dis = \begin{cases} (\sqrt[Th_d]{2})^{distance}, & distance < Th_d \\ (\sqrt[Th_d]{2})^{distance} \cdot (\sqrt{2})^{distance - Th_d}, & distance \geq Th_d \end{cases} \tag{12}$$

Th_d is set to 3km in the current study. If the distance is over *3km*, the growth rate of *Dis* will become increasingly higher.

(3) The dynamic intimacy (DI). DI supports the formation of interaction-based groups using third-party data. It is measured by the interaction frequency in OSNs between a user and her friends, based on the interaction data (e.g., comments, likes) on each other's post wall. DI characterizes the dynamic relation among users because the interaction frequency among them may change over time. In Eq. (13), we define the interaction factor of user u_i to u_j based on u_i's comments to u_j's posts during the past 30 days (considering the dynamic nature of interaction).

$$I_{<u_i, u_j>} = \sum_{k=1}^{n} (\frac{1}{2})^{\Delta t(k)} \cdot CR, \ \Delta t(k) \leq 30 \tag{13}$$

where n is the total number of comments of u_i to u_j. $\Delta t(k)$ denotes the recency, i.e., the number of days between the comment date of k and the current date. *CR* denotes the ratio of user u_i's comments to u_j over all the comments to u_j. For the dynamic intimacy between two users (u_i, u_j), we should consider both directions of interaction: u_i to u_j and u_j to u_i. The dynamic intimacy of user u_i to u_j is defined in Eq. (14).

$$DI_{<u_i, u_j>} = w_1 \cdot I_{<u_i, u_j>} + w_2 \cdot I_{<u_j, u_i>} \tag{14}$$

We consider that a user's comments to her friends are more important for recommending activities to the user, so it should have a higher weight. In the current study, the two weights w_1 and w_2 are set to 1.0 and 0.5.

(4) The static intimacy (SI). It refers to the relatively long-term or static relationship among users. Two measures — mutual intimacy and coverage intimacy — are calculated from the social tie graph for static intimacy measurement.

According to Mitchell [47], if there are more mutual friends between two users, and there are more social ties among the mutual friends, the two users are highly likely close friends. We have proposed a method to calculate static user intimacy based on this finding. As shown in Fig. 5 (a), the grey nodes represent a friend of *A* but not a friend of *B*, or a friend of *B* but not a friend of *A*, whereas the black nodes represent the mutual friends between *A* and *B*. According to the relationship between common friends, the *mutual intimacy (MI)* ratio between u_i and u_j can be defined as Eq. (15):

$$MI(u_i,u_j)=\frac{2*e}{M*(M-1)} \qquad (15)$$

where *e* denotes the actual number of edges among mutual friends, and *M* refers to the number of mutual friends of two users. For example, the *MI* ratio between user *A* and *B* (as shown in Fig. 5a) equals to 5/6.

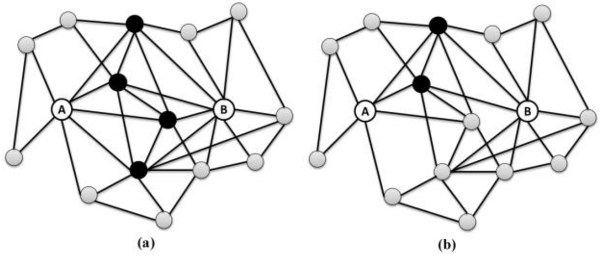


Fig. 5. Examples of social tie connection between User A and B

If two users only have two mutual friends and there is one edge between the two friends, the value of *MI* has the maximum value (i.e., *MI=1*), as shown in Fig. 5b. However, we cannot derive that the two users are very close. For example, the relationship between *A* and *B* seems closer in Fig. 5a than in Fig. 5b, but the *MI* value does not indicate this. In addition to *MI*, we define another parameter, i.e. the *coverage intimacy (CI)* ratio for *SI* measurement. *CI* measures the proportion of the number of mutual friends between u_i and u_j to the total number of u_i's friends, which can be formulated in Eq. (16):

$$CI(u_i,u_j)=\frac{M}{F(u_i)} \qquad (16)$$

where $F(u_i)$ denotes the total number of u_i's friends.

We define the static intimacy *SI* of user u_i to u_j based on the above two intimacy ratios, as formulated in Eq. (17):

$$SI_{<u_i,u_j>}=MI(u_i,u_j)\cdot CI(u_i,u_j) \qquad (17)$$

According to Eq. (18), the static intimacy of *A* to *B* in Fig. 5(a) and (b) is calculated as 10/21 (5/6*4/7) and 1/3 (1*2/6), respectively. It indicates that the static intimacy between *A* and *B* in Fig. 5(a) is higher than that of Fig. 5(b).

Finally, the ranking score for recommending u_j's activity A_k to u_i is defined in Eq. (18), where α is a constant value.

$$Rank_{DI/SI}(u_i,u_j,A_k)=\alpha*\frac{P(u_i,a_m)\cdot(DI_{<u_i,u_j>}\ or\ SI_{<u_i,u_j>})}{Dis} \qquad (18)$$

The activity information can be recommended to the list of users in terms of the order of their ranking scores.

VII. Evaluation

A. Prototype Implementation

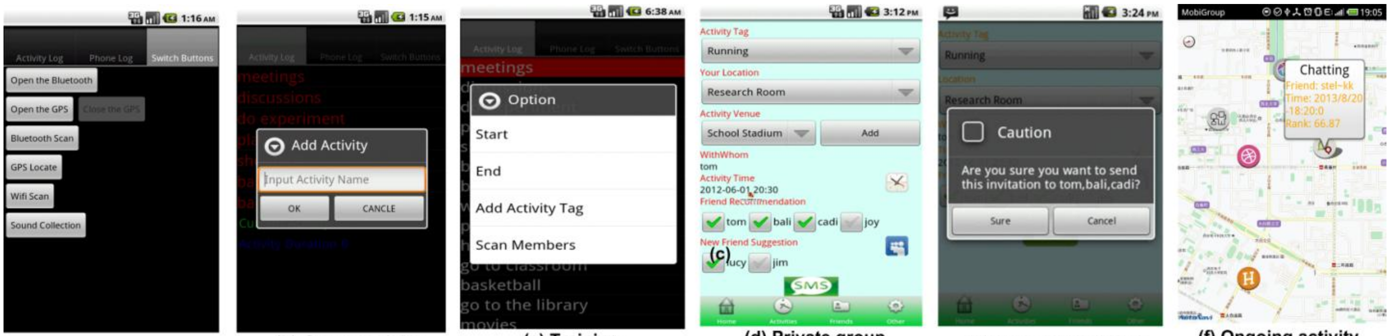
We have implemented MobiGroup on the Android platform, using SQLite as the database for ARC data storage. Users can control the working status of the relevant sensors used in MobiGroup (Fig. 6a). The interfaces for group activity creation and training activity data collection are shown in Fig. 6(b) and Fig. 6(c) respectively. Figure 6(d) displays the private group activity initiation page through which users can specify the activity venue and choose attendees from a recommended pool (relatively fixed group members). Pressing the "SMS" button sends a text message to the selected contacts (Fig. 6e). Figure 6(f) shows the interface for ongoing activity suggestion from which users can view the recommended activities on the Google map. Users can click on an activity icon for detailed information.

B. Experiment Settings and Methods

(1) Public activity advertising. The first experiment investigated whether users would benefit from our heuristic-based approach to public activity advertising. We used the dataset of activity posters in our university collected from 38 student volunteers [32]. There were many bulletin boards on campus with advertisements on different types of public events. We asked the volunteers to capture poster ads that were interesting or important to them. This simulates the post-saving function in MobiGroup, as each captured "poster" can be viewed as a 'post' in Mobigroup. Based on the data we can measure crowd-post interaction and suggest the relevant posts to the appropriate users. The data collection process lasted around eight weeks. We also acquired the social relation information from the 38 participants through a questionnaire. We detected five groups using this information with a method proposed in [48], e.g. students taking a common course and undergraduates from a student club. We applied the group detection results to 'social' post suggestion.

We conducted a user study comparing users' ratings to the 'hot' or 'social' posts recommended by our heuristic-rule-based method and a baseline method. For a user *u*, MobiGroup suggests the top-*k* 'hot' or 'social' posts (ranked by their group size). Meanwhile a named *random* method randomly selects *k* posts for *u*, which serves as the baseline. We combined the recommendations from the two methods into a unique set. In this study, *k* was set to 10, following the setting used by many recommendation services. We set Th_{hi} and Th_{ss} to 2 and 0.3 respectively in the experiments based on the dataset in [32]. More specifically, the ratio of the average size of the top 25% post groups (ranked by the group size) to the medium 50% ones are around 2 (for Th_{hi}), and the average group density of the top 10% post groups ranked by *LinkD* is around 0.3. Twenty data contributors of the poster dataset were invited to participate in this study. They were asked to rate their degree of interest to the recommended posts on a five-point Likert Scale. We can thus measure the effectiveness of the recommendation result.

(2) Private activity suggestion. The second experiment investigated whether the group computing and context-based methods are effective to support private activity group formation and event suggestion. Private activity suggestion relies on mining and analyzing historical group interaction and activity organization data. We employed two methods to collect such data, namely *smartphone logging* and *online blogging*. The former is to report user activities whenever and wherever they happen using the user interfaces presented in Section VII.A. The latter requires users to manually record their daily activities on an online blogging page. We recruited 45 students from our department, including 23 undergraduate students from the Mobile Computing course and 22 graduate students by e-mail and open calls. The data collection lasted six months. Among the 45 participants, 29 used smartphone logging and 16 used online blogging. For online blogging, since it was effortful, we asked users to record important social activities in their daily life. We collected about 1,500 data records which corresponded to22 predefined activity categories. The activities were broadly classified into three types: working (e.g., meetings, discussions), relaxation (e.g., parties), and sporting. The five most popular



Are you sure you want to send this invitation to tom,bali,cadi?

(a) Sensor control

(b) Group activity creation

(c) Training data collection

(d) Private group activity initiation

(e) Member invitation

(f) Ongoing activity suggestion

Fig. 6. User interfaces of MobiGroup.

activities were lunch (35.3%), lesson (14.8%), discussion (14.4%), sports (8.3%), and meeting (7.5%). The four most popular initiation places were lab, student dormitory, classroom, and library.

We use *Precision* and *Recall* to assess the effectiveness of the group cueing algorithm. *Precision* is measured by the ratio of the correct number (*Rig*) to the total number of recommendations (*Tot*). Recall is determined by the ratio of *Rig* to the number of people who are actually invited (*Act*), i.e., the *ground truth*. For instance, supposing that *A*, *B*, and *C* participate in an activity, and MobiGroup yields the recommendation result *A*, *B*, *D*, and *E*. The *Rig*, *Tot*, and *Act* are 2, 4, and 3, respectively. The *Precision* is thus 50% and the *Recall* is 67%. In the experiments, we chose 800 ARC records as the training set, and 400 as the test set. The *MemList* in the test records was regarded as the ground truth. Th_{sub} and Th_{com} (as defined in Section IV.C), were set empirically to 0.2 and 0.3.

The Friend Suggest (FriSug) method [3] was used as the baseline for comparison. It proposes a method for recipient suggestion in email composing using the affinity measurement metric (i.e., *AffRank*) and the seed user context (i.e., *WithWhom*). MobiGroup, however, additionally uses group computing and other contexts for real-world activity group suggestion.

(3) Ongoing Activity Recognition and Suggestion. The purpose of the following experiments is two-fold: *(1) to validate whether the activity recognition method based on mobile sound sensing is effective; (2) to test the usefulness of the hybrid community-based approach to running activity suggestion.*

We adopted our campus as the test bed and chose the five typical activities associated with student communities (as discussed in Section VI.A). A total of 600 audio clips were collected for evaluation. There were 200 speech clips including talks, chats, and meetings, 200 music as well as 200 environmental sounds from libraries/labs (80), streets (60), and sport courts (60). 75% were used for training, and the remaining for testing. The length of each audio clip is around 20 seconds. We applied our two-stage method for ambient-sound-based group activity recognition, with the three selected features in the first-stage and MFCCs in the second-stage. We compared this method with Movi [8], which only used MFCC features for ambient sound classification.

For the experiments on friend intimacy calculation (static or dynamic), we recruited 13 active users from the Sina Weibo (http://www.weibo.com) micro-blogging service. We call them *A*-*M*. *A* is an undergraduate student from the Mobile Computing course, and *B*-*G* are all friends of *A*; *H* is a graduate student of our lab, and *I*-*M* are all friends of *H*. We collected their online interaction records in the form of threaded comments for one month. We also obtained the whole list of their social ties on *Sina Weibo*, resulting in a total of 351 friends. During the test period, a total of 565 comments were collected from the 13 users' post walls.

C. *Experiment Results*

(1) Public activity advertising. Users' average rating scores on recommendations by each method are shown in Fig. 7, where we obtain the following findings:

- People prefer the 'hot' and 'social' posts suggested by our method more, given that these posts normally have higher rating than the randomly chosen posts.
- The average scores of 'social' posts are normally higher than 'hot' recommendations. It indicates that people in the same detected group do share some common interests, and group preference-based recommendation is more effective than item popularity in loosely-linked communities.

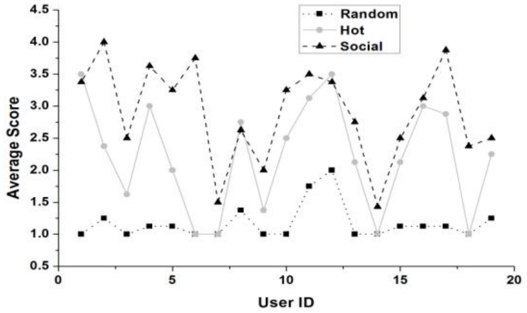


Fig. 7. Results for tagging with heuristic rules.

(2) Group-Aware Private Activity Suggestion.

The effects of contexts. One major difference between MobiGroup and other group tools is that MobiGroup provides group suggestions in real-world settings, and uses various contexts derived by smartphone sensing to filter irrelevant groups. To evaluate the effects of different contexts, we have chosen four types of contexts, with *Time* and *I-Loc (initiation place)* as the basic and *Tag*, *WithWhom(1)*, and *WithWhom(2)* as additional criteria for group filtering. The results in Table II indicate that rich contextual information can enhance recommendation performance and that the *WithWhom* context performs better than the *Tag* context does.

The effects of group abstraction. Group abstraction (GA) is another contribution to group management, which can eliminate noise in group detection and merge relevant raw groups into bigger logical units. The results in Table II suggest that group abstraction can characterize the social graph of a user and enhance the performance of private group suggestion.

TABLE II. EFFECTS OF CONTEXTS TO GROUP SUGGESTION

Contexts	*Precision*	*Recall*
AR (AffRank) + WithWhom (1) [3]	35%	40%
MobiGroup – Context	29%	31%
MobiGroup – GA	40%	44%
Time + I-Loc	57%	75%
Time + I-Loc + Tag	70%	81%
Time + I-Loc + WithWhom (1)	70%	93%
Time + I-Loc + WithWhom (2)	84%	98%

Using *spatio-temporal contexts* and *group abstraction*, the performance of MobiGroup is much higher (e.g., the precision increases from 35% to 70%) than that of FriSug [3]. It indicates that the two factors are useful for real-world group suggestion.

(3) Ongoing Activity Recognition and Suggestion. For activity recognition, the first stage is to classify the three general activities: *talk*, *relaxation*, and *other activities*. We constructed a decision tree with the J48 algorithm based on the selected features extracted from audio clips. From the results in Fig. 8, the *F1*-measure score was above 75%. *Precision* and *recall* of the first-stage classification based on the three selected features are higher than using MFCCs directly, while the computational cost is reduced by more than 50% (5230 vs. 2035 multiplication calculations and 5912 vs. 2841 addition calculations per frame for MFCCs and MobiGroup). A Chi-square test did not indicate any statistical relationship between proportion of correct results for MobiGroup as compared to MFCC for activity recognition.

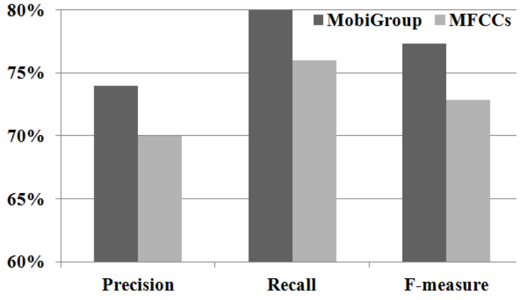


Fig. 8. Experiment results for the first-stage activity recognition.

TABLE III. THE CONFUSION MATRIX FOR ACTIVITY RECOGNITION

Class	*Sporting*	*Quiet*	*Street Roaming*
Street Roaming	17.0%	13.8%	69.2%
Sporting	73.4%	14.2%	12.4%
Quiet	6.4%	93.6%	0%

We list the confusion matrix of three fine-grained activities recognized by the second-stage MFCC-based method in Table III. Because of the diversity of the background sounds, the recognition rate of street roaming activity only reaches 69.2%, often mistaken for sporting events. In contrast, the features of the ambient sound in quiet places are very stable, thus the recognition rate is much higher.

Based on the activity recognition results, MobiGroup recommends relevant activity to users in an opportunistic manner. Two methods are proposed in Section V.B to meet host users' distinct needs on participants, namely static intimacy (SI) and dynamic intimacy (DI). Several other parameters are also considered, such as user preferences, distance, and so on. We assumed that the preference degree were the same (all set to 0.5), and their intimacy can be computed according to the collected dataset. The distances among users were randomly generated within the range of 0-8*km*. The recommendation results with the two methods are shown in Table IV and V.

TABLE IV. RUNNING ACTIVITY RECOMMENDATION (α =200)

User	*Friend*	*Intimacy (Dynamic/Static)*	*Distance (km)*	*Ranking Value (Dynamic/Static)*
A	*B*	0.56 / 0.44	0.68	46.1 / 36.0
	C	0.07 / 0.75	0.11	6.8 / 73.0
	D	0.87 / 0.28	1.32	66.9 / 21.5
	E	0.10 / 0.26	0.57	8.8 / 22.8
	F	1.04 / 0.66	4.13	27.1/ 17.2
	G	0.09 / 0.47	0.98	5.5 / 28.7
H	*I*	0.47 / 0.44	0.36	43.2 / 40.5
	J	0.92 / 0.61	6.11	7.67 / 5.08
	K	0.32 / 0.68	0.66	27.5 / 58.4
	L	0.25 / 0.54	2.45	14.2 / 30.7
	M	0.18 / 0.50	1.05	14.1 / 39.2

TABLE V. SUGGESTING RESULTS FOR RUNNING ACTIVITIES (SIZE=3)

User	*Dynamic Intimacy (DI)*	*Static Intimacy (SI)*
A	{D, B, F}	{C, B, G}
H	{I, K, L}	{K, I, M}

In Table IV dynamic and static intimacy do not always follow the same variation trend. For example, though the DI of friend *C* to *A* (and *K* to *H*) is low, but the SI is the highest. In the DI recommendation, the activity distance of friend *B* and friend *E* is almost the same, but the DI of friend *E* is lower than that of friend *B*, so the recommendation value of friend *B* is higher than that of friend *E*. Although the DI of friend *I* is lower than that of friend *J*, the activity distance of friend *I* is much lower than that of friend *J*, so the recommendation value of *I* is higher. In the SI recommendation, the activity distances of friend *B* and *E* are almost the same, but the SI of friend *B* is higher than that of friend *E*, so the ranking of *B* is higher than *E*.

Assume that three friends are to be invited (i.e., group size = 3), the results for *A* and *H* are shown in Table V. It is clear that the selected members and their order differ in the suggested DI or SI groups for the same user. There are also common members in DI and SI groups (one for *A* and two for *H*), indicating that some people are both strong in structural connection and dynamic interaction.

The above results indicate that the two social contexts (DI and SI) recommend activities from different viewpoints, and the spatial context also impact the decision process. In other words, group suggestion for ongoing activities is highly context-dependent, to meet diverse user needs and tastes.

D. Discussion

Group activities are very complex and people often have different requests. This work mainly addresses two needs of users, namely *planned activity preparation* and *ongoing activity suggestion*. Other needs are to be studied in the future.

Identification of groups is important to social activity sharing and suggestion, especially for ongoing activities. In the current study, we mainly use the proximity information to identify members of existing groups. However, sometimes it is not easy to differentiate whether two nearby people are involved in the same group activity. We will explore new sensing techniques to address this problem [8, 49].

There have been numerous studies about item recommendation. For instance, collaborative filtering [50] has been used for user-similarity-based recommendation. Gartrell et al. [51] proposed a group recommendation approach (e.g., recommending a TV program to a crowd) based on both content and social interests. Our work focuses on context-based ongoing activity recommendation, where the factors such as user-activity distance and interaction dynamics are considered. We intend to improve our work in the future by integrating promising recommendation techniques in other application domains.

Privacy is always a sensitive issue to smartphone sensing applications. People are usually serious and cautious about collecting their location and audio data. To address this, MobiGroup limits the length of an audio clip to 20 seconds for activity recognition, and the feature extraction is undertaken on local mobile clients.

VIII. Conclusion

In this paper we have presented our efforts for group activity organization and recommendation in the real-world settings. We present a generic conceptual model for characterizing the diversity and regularity of group activities. Built upon this model, a combination of group computing, intelligent tagging, and context learning approaches are proposed to facilitate planned group activity preparation. For running activities, we have leveraged a combination of mobile sound sensing and cross-community mining techniques to provide context- and group-aware suggestions for those who meet certain constraints. Experiments indicate that people show high interesting degree to the ‘hot’ or ‘social’ tagged public activities. Findings also demonstrate that by using various contexts and the group abstraction process, the performance of private group suggestion is improved. In addition, by leveraging cross-space community data, we can meet diverse social/physical constraints in running activity suggestion.

References

[1] A.C. Ellis, S.J. Gibbs, and G. Rein, “Groupware: some issues and experiences,” *Commun. ACM*, vol. 31, no. 1, pp. 39-58, 1991.

[2] D. MacLean, et al., “Groups without tears: mining social topologies from email,” in *Proc. 16th Int. Conf. Intell. User Interface*, 2011, pp. 83-92.

[3] M. Roth, et al., “Suggesting friends using the implicit social graph,” in *Proc. 16th ACM SIGKDD Int. Conf.*, 2010, pp. 233-242.

[4] A. Saleema, J. Fogarty, and D. Weld, “Regroup: Interactive machine learning for on-demand group creation in social networks,” in *Proc. SIGCHI Conf.*, 2012, pp. 21-30.

[5] M. Eslami, et al., “Evaluation of automated friend grouping in online social networks,” in *Proc. of SIGCHI Conf.*, 2014, pp. 2119-2124.

[6] S. Whittaker, et al., “ContactMap: Organizing communication in a social desktop,” *ACM Trans. on Comput. Hum. Int.*, vol. 11, no. 4, pp. 445-471, 2004.

[7] E. Miluzzo, et al., “Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application,” in *Proc. 6th ACM Conf. Emb. Network Sensor Syst.*, 2008, pp. 337-350.

[8] X. Bao and R.R. Choudhury, “MoVi: mobile phone based video highlights via collaborative sensing,” in *Proc. 8th ACM Conf. Emb. Network Sensor Sys.*, 2010, pp. 357-370.

[9] E.G. Boix, et al., “Flocks: enabling dynamic group interactions in mobile social networking applications,” in *Proc. 2011 ACM Symp. Appl. Comput.*, 2011, pp. 425–432.

[10] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, “Opportunistic IoT: Exploring the Harmonious Interaction between Human and the Internet of Things,” *J. Netw. Comput. Appl.*, vol. 36, no. 6, pp. 1531-1539, 2013.

[11] D. Zhang, B. Guo, and Z. Yu, “The Emergence of Social and Community Intelligence,” *Comput.*, vol. 44, no. 7, pp. 21-28, 2011.

[12] B. Guo, et al., “Mobile Crowd Sensing and Computing: The Review of an Emerging Human-Powered Sensing Paradigm,” *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1-31, 2015.

[13] B. Guo, et al., “GroupMe: Supporting Group Formation with Mobile Sensing and Social Graph Mining”, in *Proc. Mob. Ubiq. Syst.*, 2012, pp. 200-211.

[14] A. Jameson and Barry Smyth, “Recommendation to groups,” in *Proc. Adaptive Web*, Springer, 2007, pp. 596-627.

[15] B. Guo, Z. Yu, D. Zhang, and X. Zhou, “Cross-Community Sensing and Mining”, *IEEE Commu. Mag.*, vol. 52, no. 8, pp. 144-152, 2014.

[16] H.Z. Kim and K.S. Eklundh, “Reviewing practices in collaborative writing,” *Comput. Supp. Coop. W. J.*, vol, 10, no. 2, pp. 247-259, 2001.

[17] Z. Yang, et al., “Enabling multi-party 3D tele-immersive environments with ViewCast,” *ACM Trans. Multi. Comput.*, vol. 6, no. 2, pp. 1-30, 2010.

[18] E. Cho, S. A. Myers, and J. Leskovec, “Friendship and Mobility: user movement in location-based social networks,” in *Proc. 17th ACM SIGKDD Int. Conf.*, 2011, pp. 1082-1090.

[19] J. Tang, et al., “Inferring Social Ties across Heterogeneous Networks,” in *Proc. 5th ACM Int. Conf. Web Search Data Mining*, 2012, pp. 743-752.

[20] B. Guo, D. Zhang, D. Yang, Z. Yu, and X. Zhou, “Enhancing Memory Recall via an Intelligent Social Contact Management System,” *IEEE Trans. on Hum. Mach. Syst.*, vol. 44, no. 1, pp. 78-91, 2014.

[21] N. Cross and A.C. Cross, “Observations of teamwork and social processes in design,” *Design studies*, vol. 16, no. 2, pp. 143-170, 1995.

[22] B.A. Nardi, “Context and consciousness: activity theory and human computer interaction,” *MIT Press*, 1996.

[23] D. Easley and J. Kleinberg, “Networks, Crowds, and Markets,” *Cambridge University Press*, 2010.

[24] S. Wakamiya, R. Lee, and K. Sumiya, “Urban area characterization based on semantics of crowd activities in twitter,” in *Proc. GeoSpatial Semantics*, 2011, pp. 108-123.

[25] Q. Chuan, et al., “Tagsense: a smartphone-based approach to automatic image tagging,” in *Proc. 9th ACM Conf. Emb. Network Sensor Sys.*, 2011, pp. 1-14.

[26] H.V. Enrique, et al., “A consensus model for group decision making with incomplete fuzzy preference relations,” *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 5, pp. 863-877, 2007.

[27] M.E.J. Newman, “Modularity and community structure in networks,” *Proc. Natl. Acad. Sci.*, vo. 103, no. 23, pp. 8577-8582, 2006.

[28] J. Staiano, et al., “Friends don’t lie: inferring personality traits from social network structure,” in *Proc. ACM Conf. Ubi. Comp.*, 2012, pp. 321-330.

[29] A.K. Dey, “Understanding and Using Context,” *Pers. Ubiquit. Comput.*, vol. 5, no. 1, pp. 4-7, 2001.

[30] C. Bizer, et al., “DBpedia-A crystallization point for the Web of Data,” *J. Web Semantics*, vol. 7, no. 3, pp. 154-165, 2009.

[31] Q. Yuan, et al., “Time-aware Point-of-interest Recommendation,” in *Proc. 36th Int. ACM SIGIR Conf.*, 2013, pp. 363-372.

[32] B. Guo, et al., “FlierMeet: Cross-Space Public Information Reposting with Mobile Crowd Sensing,” in *Proc. ACM Conf. Ubi. Comp. Adjunc.*, 2014, pp. 59-62.

[33] K. Cheverst, K. Mitchell, and N. Davies, “Investigating context-aware information push vs. information pull to tourists,” in *Proc. Mobile HCI.* 2001, pp. 1-6.

[34] J. Yang and J. Leskovec, “Patterns of temporal variation in online media,” in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 177-186.

[35] R. Kumar, J. Novak, and A. Tomkins, “Structure and Evolution of Online Social Networks,” *Link mining: models, algorithms, and applications*, Springer, 2010, pp. 337-357.

[36] Q. Yang, et al., “Estimating Location Using Wi-Fi,” *IEEE Intell. Syst.*, vol. 23, no.1, pp. 8-13, 2008.

[37] E. Tapia, et al., "Activity recognition in the home using simple and ubiquitous sensors," in *Proc.Int. Conf. Per. Comp.*, 2004, pp. 158-175.

[38] N. Ihaddadene, et al., "Crowd behaviour monitoring," in *Proc. 16th ACM Int. Conf. Multi.*, 2008, pp. 1013-1014.

[39] X. Bao and R.R. Choudhury, "VUPoints: collaborative sensing and video recording through mobile phones," *ACM SIGCOMM Comput. Comm. Review*, vol. 40, no.1, pp. 100-105, 2010.

[40] M. Azizyan, I. Constandache, and R.R. Choudhury, "SurroundSense: mobile phone localization via ambience fingerprinting," in *Proc. of 15th ACM Int. Conf. Mobi. Comput. Netw.*, 2009, pp. 261-272.

[41] R. Jang. (Jun. 2008). Audio signal processing and recognition. [Online]. Available: http://mirlab.org/jang/books/ audioSignalProcessing.

[42] T. Zhang and C. Kuo, "Hierarchical classification of audio data for archiving and retrieving," in *Proc. Int. Conf. Acoustic, Speech, and Signal Processing*, 1999, pp.3001-3004.

[43] H. Lu, et al., "SoundSense: scalable sound sensing for people-centric applications on mobile phones," in *Proc. 7th ACM Conf. Emb. Network Sensor Sys.*, 2009, 165-178.

[44] G. Tzanetakis and P. Cook, "Sound analysis using MPEG compressed audio," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 2000, pp. 11761-11764.

[45] Y. Zhan and T. Kuroda, "Wearable sensor-based human activity recognition from environmental background sounds," *J. Amb. Intel. Hum. Comput.*, vol. 5, no. 1, pp. 77-89, 2014.

[46] D. J. Berndt and J. Clifford., "Using Dynamic Time Warping to Find Patterns in Time Series," in *Proc. ACM KDD Workshops*, vol. 10., no. 16, 1994, pp. 359-370.

[47] M. Mitchell, "Complex Systems: Network Thinking," *Artif. Intell.*, vol. 170, no. 18, pp. 1194-1212, 2006.

[48] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast Unfolding of Communities in Large Networks," *J. Stat. Mech. Theory E.*, no. 10, pp. 1-12, 2008.

[49] D. Gordon, et al., "Group affiliation detection using model divergence for wearable devices," in *Proc. 2014 ACM Int. Symp. Wear. Comput.*, 2014, pp. 19-26.

[50] B. Sarwar, et al., "Item-based collaborative filtering recommendation algorithms," in *Proc. ACM 10th Int. Conf. World Wide Web*, 2001, pp. 285-295.

[51] M. Gartrell, et al., "Enhancing group recommendation by incorporating social relationship interactions," in *Proc. of ACM 16th Int. Conf. Sup. Group Work*, 2010, pp. 97-106.

Bin Guo is a professor from Northwestern Polytechnical University, China. He received his Ph.D. degree in computer science from Keio University, Japan in 2009 and then was a post-doc researcher at Institut TELECOM SudParis in France. His research interests include ubiquitous computing, mobile crowd sensing, and HCI.

Zhiwen Yu is currently a professor and vice-dean at the School of Computer Science, Northwestern Polytechnical University, China. He has worked as an Alexander Von Humboldt Fellow at Mannheim University, Germany from Nov. 2009 to Oct. 2010, a research fellow at Kyoto University, Japan from Feb. 2007 to Jan. 2009. His research interests cover pervasive computing and HCI.

Liming Chen is Professor of Computer Science and the Head of the Context, Intelligence and Interaction Research Group (CIIRG) in the School of Computer Science and Informatics at De Montfort University, United Kingdom. He received his B.Eng and M.Eng from Beijing Institute of Technology, Beijing, China, and his Ph.D in Artificial Intelligence from De Montfort University, UK. His research interests include activity recognition, intelligent systems, smart environment and assisted living.

Xingshe Zhou is a professor from the School of Computer Science, Northwestern Polytechnical University, China. He is the director of Shaanxi Key Laboratory of Embedded System Technology. His research interests include embedded computing and pervasive computing.

Xiaojuan Ma is an assistant professor of Human- Computer Interaction (HCI) at the Department of Computer Science and Engineering, Hong Kong University of Science and Technology (HKUST). She received the Ph.D. degree in Computer Science at Princeton University. Before joining HKUST, she was a researcher of Human-Computer Interaction at Noah's Ark Lab, Huawei Tech. Investment Co., Ltd. in Hong Kong. Her research interests include human-computer interaction, crowdsourcing, ubiquitous computing, and computational linguistics.