Using Physical Context in a Mobile Social Networking Application for Improving Friend Recommendations

Bin Xu, Alvin Chin, Hao Wang, Hao Wang
Mobile Social Experiences Group
Nokia Research Center
Beijing, China
{ext-bin.7.xu, alvin.chin, hao.ui.wang}@nokia.com, alexwhu@gmail.com

***Abstract*— In online social networks such as Facebook, people receive friend recommendations that are based usually on common friends or similar profile such as having the same interest or coming from the same company. However, people receive friend spam in which they do not know why they should add this friend. If we can record the physical context then we can determine how you met that person, and use that for recommending that person to you. In this paper, we create a friend recommendation system using proximity encounters and meetings as physical context called EncounterMeet. We conduct a user study to examine whether physical context-based friend recommendations is better than common friends. Results show that averagely, the EncounterMeet algorithm recommended more friends to participants that they added, more recommendations were ranked as good, compared with the common friend algorithm. The results can be used to help design context-aware recommendations in physical environments.**

Keywords-Friend recommendation, physical context, proximity, mobile social network

I. INTRODUCTION

In online social networks such as Facebook, friend recommendations help to expand a user's social circle. Most friend recommendations are based on common friends that you and your potential friend have in common, as well as similar profile and shared content. However, very often we receive a lot of friend spam where we do not know why those "friends" have been recommended. In fact, Miller asks how many of your Facebook friends are really your friends (that you have a close friendship with)? [1] In addition, there are many instances where you have to manually remember the new friends that you have made and then add them later to your online social networks. The problem exists in that social networking sites only know the context of how two people know each other if they explicitly indicate that, but often this explicit relationship has not been recorded because there is no integration with the real world activities.

How do we solve this? How can we get physical context and interactions? We need to create a system and platform for recording physical context and social interactions in the real world, which can be used as a trigger for opportunistic social networking. We create Find & Connect as this platform, and create a method for recording physical context in an indoor environment such as encounters and meetings.

In our previous work [28], we found that more interactions in physical context result in a higher possibility in friendship formation. In this paper, we create EncounterMeet and EncounterMeet+ friend recommendation algorithms that use encounters and meetings as the physical context in an office indoor environment. We hypothesize that the quality of friend recommendations based on physical context will be better than those based on common friends. To demonstrate if this is true, we perform a user study between the EncounterMeet and common friend recommendation algorithms.

Results show that the EncounterMeet algorithm recommended more friends to participants that they added (50% compared to 38% for common friends), more recommendations were ranked as good (44% compared to 32% for common friends), and more had previous acquaintance with these recommendations (69% vs. 59% for common friends), compared with the common friend algorithm. We also present a friend recommendation interface based on our novel EncounterMeet+ algorithm (combination of common friends, similar profile, shared content, and EncounterMeet) to solicit participants' feedback which was positive. Though the study is conducted with a small sample size (10), the results demonstrate the potential usefulness of physical context-based friend recommendations such as EncounterMeet.

Our contributions are the following. First, we build a friend recommendation system that is based on indoor physical context and people interactions (indoor proximate encounters) captured in real time. Second, we conduct a detailed friend recommendation study where we not only examine the quality of the recommendations based on the accepted recommendations, but also ask questions related to acquaintance of the recommended individual and reasons for accepting this friend, therefore making it more clear on who users want to be friends with and why they have the preferences. Third, we present a new friend recommendation interface that allows the user to select the desired weight score for each feature in our recommendation algorithm, to personalize the results of the recommendations tailored to the user.

The paper is organized as follows. Section 2 describes background and related work on state-of-the-art in friend recommendations and motivates the need for physical context-based friend recommendations. Section 3 presents our platform that we built for supporting physical context-based friend recommendations. In Section 4, we describe how we use encounters as the physical context for friend recommendations and describe our EncounterMeet+ algorithm from which EncounterMeet is a base case. In Section 5, we explain the user study for comparing friend recommendations based on common friends with friend recommendations based on encounters and meetings (EncounterMeet). Section 6 shows our results from the user study and survey. Finally, Section 7 concludes the paper and discusses future work.

II. BACKGROUND AND RELATED WORK

Most social network and social media sites have friend recommendations to encourage users to increase their social circle [27]. In this section, we look at the state-of-the-art in friend recommendations which can be based on common friends, similar profile and shared content; and physical and social context.

A. Friend Recommendations Based on Common Friends, Similar Profile and Shared Content

Current friend recommendation systems in Facebook, LinkedIn and MySpace [23] known as "People You May Know", are mainly based on common friends and similar profile characteristics [17] such as being in the same network or same company. However, the system may still recommend people whom the user does not know so she will not add them as a friend. Therefore, being able to recommend more known people is important to improve the quality of the recommendations [5] and one way of doing so is to use shared content and interactions. For example, in the enterprise, co-authored papers, patents, and comments have been used as the shared content and interactions, and an interface has been designed to explain the reasons for why you should add this person as a friend and to provide recommendation feedback [14, 15].

To calculate the friend recommendation score in order to rank the friend list, weights are assigned to each feature in the recommendation. The weights can be assigned empirically by the system based on user behavior [10] or manually entered by the user. However recommendation systems ignore the physical interactions to associate how you may know that person [11, 18, 26]. Our work differs from the above in that we add physical context (encounters and meetings) and social interactions (messages, and question and answer posts) in addition to similar profile (common interests) and common friends, and our new algorithm assigns weights based on user input.

B. Friend Recommendations Based on Physical and Social Context

With a rapid rise in location-based social network applications such as Foursquare, location based on GPS and/or WiFi is the shared physical context which has become the basis for social interaction and friend recommendations. Previous work [13, 14, 16] shows that the more social network information and sources integrated, the richer the result and the closer the returning people are to the ideal friend list. Thus, physical interactions within proximity may be utilized to recommend similar-minded people [8, 9, 24]. In fact, Cranshaw [9] shows how physical location can be used to recommend friends in online social networks that are nearby and this has been implemented by applications such as SONAR [25]. Froehlich et al [12] show that there is a positive correlation between physical place preference and visiting frequency and visit time.

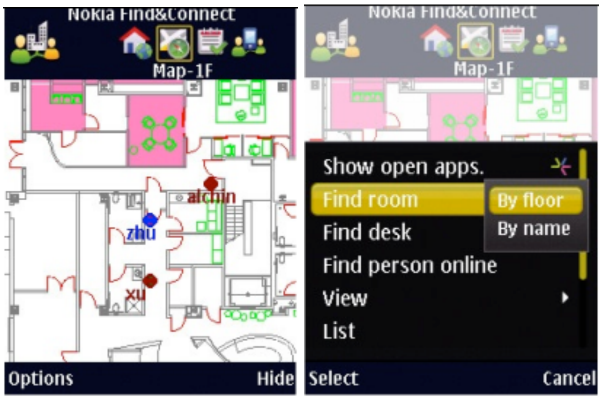
Context is usually added as an additional feature in collaborative filtering based recommendation algorithms [30] and physical context is usually captured from positioning systems such as GPS or RFID [20]. In addition, statistics of logged data can be used to infer spiritual friendship based on similar behavior and social friendship based on explicit user relationships, in order to create friend recommendation scores [19]. Similarly, we do have user input context (profile for comparing interests) and we do create friend recommendation scores. However, in our work, priority weight is given to physical proximity when recommending the potential friends, since the people nearby that you may see and listen to, may be talking with you. Physical context in our work is not absolute location like GPS or RFID location coordinates, but rather semantic where we use encounter as the combination of two types of physical context (location and time) to represent mobility interaction and use WiFi as the positioning system. Encounter is defined later in this paper.

Social context can also include social interactions between users. For example, Lo and Lin [22] use weighted minimum message ratio to determine friends, and our previous work uses social network analysis on the conversation graph extracted from message interactions [6]. Besides context and content, domain knowledge can also be included to improve content interest [31] using ontologies, often used with service recommendations [7]. In our work, for social interactions, we use the number of messages and number of question and answer posts recorded by our system. We now explain our system for supporting physical context-based friend recommendations in the next section.

III. PLATFORM FOR PHYSICAL CONTEXT-BASED FRIEND RECOMMENDATIONS: FIND & CONNECT

To obtain the physical context needed for integrating with friend recommendations, and determine whether physical context influences friend recommendations, we build a platform for ephemeral social networking and friend recommendations in an indoor environment called Find & Connect.

Before becoming friends, there needs to be a mechanism for finding where potential friends could be and where there are meetings or activities for finding these friends. As a result the 'Find' part of Find & Connect allows users to find resources such as meeting rooms, then 'Connect' allows users to initiate an interaction with them such as becoming friends, for example, at the end of a meeting.



(a) **(b)**

Figure 1. Finding resources in Find & Connect by (a) locating resources on a map (here, locating users on a floor) and (b) searching a resource with some criteria (here, searching for a room by a certain floor).

A. Finding Resources

In Find & Connect, we define a resource as a physical object in an environment where social interaction and ephemeral social networking can take place. Just like we have social interaction centered around objects such as photos and videos (known as object-centred sociality [29]), in the indoor environment, we have social interaction around which we already naturally do every day. The resources in an office environment include rooms, office desks, and people, and in a conference include papers and sessions. The resources can then be found and located on a map using our phone application as shown in Figure 1 (a) or can be searched as in Figure 1 (b). For locating people on a floor, we use a WiFi positioning system where the phone's WiFi signal strength to the nearest WiFi access points are compared to a WiFi positioning model that records a radio map of the WiFi signals on the floor, in order to estimate the user's location [2,4].

B. Connecting to People

After a resource has been found (eg., a meeting room), then you can see whether there are people there and decide to whether initiate a connection with a person. This shows how the "connecting" is integrated with the "finding" of the resource. Some of the features that we build for connecting to people include adding a friend, following someone, sending a message to someone or to a group, posting a message to a session, posting a status update, and recommending friends and people to follow. Figure 2 shows an interface for friend recommendations, where we provide the reasons for adding this friend (which are lacking in current social networks). This provides more information to the user to help them decide whether she should add this friend or not.

IV. USING ENCOUNTERS FOR EPHEMERAL SOCIAL NETWORKING AND FRIEND RECOMMENDATIONS

We integrate the physical context captured by Find & Connect (namely encounters and meetings) into the friend

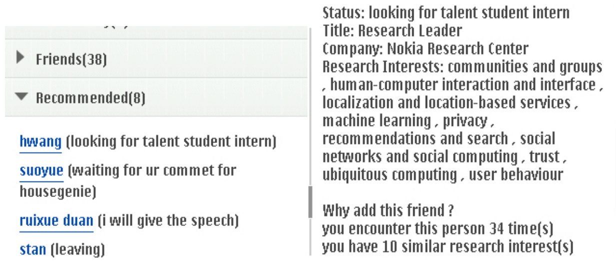


Figure 2. Friend recommendation interface in Find & Connect with (a) list of recommended friends and (b) profile of the recommended friend and the reasons for adding this friend.

recommendations. The encounter information forms the basis for ephemeral social networking and integration into the friend recommendations.

A. Definition of Ephemeral Social Network

We define an ephemeral social network as a social graph of nodes and edges where the nodes are individuals and the edges indicate that the individuals connected have encountered each other at a specific time for at least a specific duration defined by an encounter duration threshold. In some respects, this is similar to an opportunistic network [21]. This ephemeral social network provides a network of opportunity where individuals can form social relationships like being friends. For example, people at a meeting can form an ephemeral social network.

B. Encounter Graph

For creating an ephemeral social network in an indoor environment, we need to create an encounter graph by first mining each user's position and calculating the distance between that user and all other users that are on the same floor. We then create a graph $G_{en}(V, E)$ where V is the set of nodes (v_i | $1 \leq i \leq N$), N is the number of nodes and E is the set of edges (e_{ij} | $1 \leq i \leq N$, $1 \leq j \leq N$, $i \neq j$) and

- The node v_i is user i and node v_j is user j and the edge e_{ij} is a link when two users (v_i and v_j) encounter each other where the encounter distance threshold is defined as ΔD
- The edge (e_{ij}) has a timestamp attribute to define when the encounter happens called $T_{en}(e_{ij})$

Figure 3 illustrates an example of an encounter that occurs in our definition. ΔD is the defined threshold, and t_1, t_2, t_3, t_4, t_5 are adjacent time points. At time t_1, the distance D_1 between user v_i and user v_j is larger than ΔD, then at t_2, they move closer to a distance D_2 smaller than ΔD. We record t_2 as the start time of this encounter denoted as T_{en}. Then they keep moving closer until t_5, where they move apart and their distance D_5 is larger than ΔD.

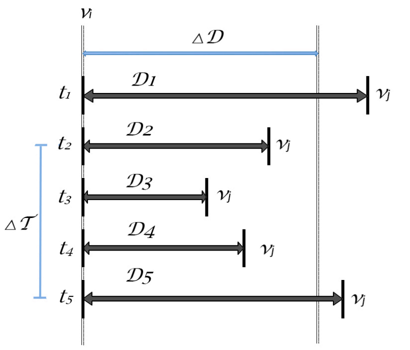


Figure 3. Defining an encounter between two users v_i and v_j.

We record t_5 as the end time of the encounter. Thus, $\Delta T_{en}(e_{ij}) = t5 - t2$ is the encounter duration, and we record the last distance less than ΔD (distance $D4$ between v_i and v_j at t_4) as the encounter distance $D_{en}(e_{ij})$.

C. EncounterMeet+: Encounter-based Friend Recommendation Algorithm

We use the encounter graph as the one kind of the evidence for showing people that you might want to make friends with because you were together with them and perhaps were talking with them. Note that Find & Connect does not know if you are actually talking with them but only records the possibility that you might be due to your encounter distance and the encounter duration.

EncounterMeet+ algorithm uses various evidence together to recommend potential friends to users. Here are the kinds of evidence on which our recommendations are based.

Common interests (ci): In the user's profile of Find & Connect, users can specify their interests from a pre-defined list and if you and user *B* have the same interest then user *B* may likely be a recommended friend. This is similar to the SONAR algorithm of Guy et al [16].

Common friends (cf): This is the common friends algorithm used in popular social networking sites such as Facebook and LinkedIn's "People You May Know" feature, where if you and user *B* both have the same friend, then user *B* can likely be a recommended friend.

Common meetings (cm): Based from meetings that users can book in Find & Connect and from our meeting room reservation system.

Encounters (e): If you and user *B* encounter each other several times a day and for several days, then user *B* may likely be a recommended friend you want to add because of your previous proximity interactions.

Pass by (p): If you and user B pass by each other several times a day and for several days, then user B may likely be a recommended friend. Pass by differs from encounters in that in a pass by, users approach each other from opposite directions and meet for a very short period of time (less than the time for an encounter) before passing each other and going in opposite directions.

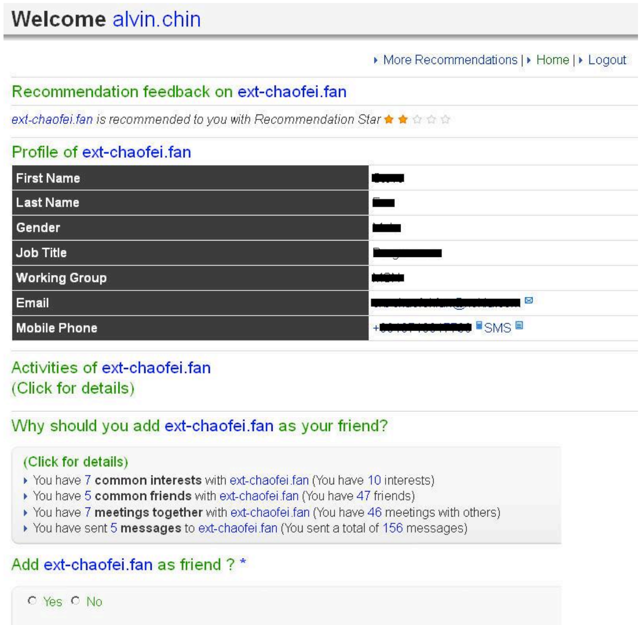


Figure 4. User interface for friend recommendations from the EncounterMeet+ recommendation algorithm. The interface is exactly the same for EncounterMeet and common friends except the content is different.

Mobile Q&A (qa): In Find & Connect, a user can send a question to a group of people and if any one of those people is online, they can answer that question. If user *B* answers several questions from you, then user *B* may likely be a recommended friend that you want to add because of your previous social interactions. This is a new feature.

Messages (m): In Find & Connect, users can send a message to another user. If user *B* sends several messages to you, then user *B* may likely be a recommended friend. This is a new feature.

From all these features, we describe the EncounterMeet+ algorithm below.

Define a weight vector w_i:

$$w_i = \{w_{ci}, w_{cf}, w_{cm}, w_e, w_p, w_{qa}, w_m \mid w_{ci} + w_{cf} + w_{cm} + w_e + w_p + w_{qa} + w_m = 1, 0 \leq w_f \leq 1\} \quad (1)$$

for each user U_i that EncounterMeet+ will recommend potential friends to user U, where depending on the importance of each feature to the friend recommendation, we attach a weight w_f to each feature f.

Next, define the relevance vector R_i:

$$R_i = \{R_{ci}, R_{cf}, R_{cm}, R_e, R_p, R_{qa}, R_m\} \quad (2)$$

for user U_i's relevance to user U in each feature f, where user U_i is not in the friend list of user U. Relevance R_f is measured by the Jaccard similarity of that feature f between U_i and U as

$$R_f = | N_f(U_i \cap U) | / |N_f(U_i \cup U)| \quad (3)$$

where N_f refers to the frequency usage or appearance count for that feature. However, other similarity measurements such as Pearson's Coefficient and Cosine Similarity, can be used to define the relevance between U_i and U in each feature space.

We define the recommended score FR_i for recommended friend U_i to user U as:

$$FR_i = w_i \cdot R_i = \{w_{ci}, w_{cf}, w_{cm}, w_e, w_p, w_{qa}, w_m\} \cdot \{R_{ci}, R_{cf}, R_{cm}, R_e, R_p, R_{qa}, R_m\}^T \quad (4)$$

Find & Connect then use this score to determine whether a user will be recommendation to the other in a user pair.

Here our main objective is to determine from a user perspective whether friend recommendations based on physical context (encounters and meetings) are better than friend recommendations based on common friends. To test this hypothesis, we conduct a friend recommendation user study in the office using our Find & Connect office system [4] and we present this in the next section below.

V. FRIEND RECOMMENDATION USER STUDY

A. Study Setup

We recruited 10 employees in the office who used Find & Connect frequently for booking meetings and had many position updates in the system, of these 8 are male and 2 are female. The study took 1 hour to complete and participants were asked to perform two tasks on a phone. The first task was to evaluate up to 10 friend recommendations based on common friends, whereas the second task was to evaluate up to 10 friend recommendations based on physical context where we used encounters and meetings (EncounterMeet algorithm). We used EncounterMeet rather than EncounterMeet+ because we did not want other features other than encounters and meetings to affect the recommendation feedback from the participants, so as not to skew our results. We ranked each friend recommendation with a score and presented the rank to the user with the number of recommendation stars. For each friend recommendation in each task, subjects were shown the profile of the suggested friend and their activities (eg. meetings they attended), as well as the reasons for adding the suggested friend as shown in Figure 4.

In order to evaluate the quality of the algorithms, subjects were asked to complete a set of questions about the recommended friend to determine if this was a person they knew (in real life, in online social networks, or in their phonebook) and whether the recommendation was good, followed by the option of adding that person as a friend, similar to questions asked in [6]. Finally, we provided the third recommendation algorithm EncounterMeet+ which was a combination of common friend, similar profile, shared content and EncounterMeet as an interface shown in Figure 4 and observed the user's behavior in using this interface. Note, the user interface for the two friend recommendation tasks in the user study, are similar to Figure 4, except the content is different. We did not incorporate the EncounterMeet+ algorithm into the tasks as it would not have provided a direct comparison, but just added it to the study for users to explore its recommendations.

The reasons for adding the suggested person as a friend (displayed as a list) are similar to the recommendation user interface by Guy et al [17] where they use similar content viewed or published, instead of activities and interactions.

TABLE I. RESULTS FROM FEEDBACK IN FRIEND RECOMMENDATION UI.

	Common friend	**EncounterMeet**
# of total recommendations	81	83
Average # of recommendations presented per user	8.1	8.3
% of good recommendations	32.1	44.6
% of recommended persons already known	24.7	37.3
% of recommended persons in phonebook	9.8	13.3
% of recommended persons in SNS	14.8	16.9
% of recommendations accepted	38.3	50.1

After the two tasks, subjects answered a series of questions from a survey to collect their feedback on the usefulness and usability of the EncounterMeet recommendation algorithm compared to current friend recommendation algorithms based on common friends and similar interests. For the friend recommendation study, we set k = 10 for the top k friend recommendations and in order to compare with the common friend algorithm, we do not take into account similar profile, content, or common relationships, but rather only just encounters and meetings (EncounterMeet), therefore w_{ci}, w_{cf}, w_{qa} , w_p and w_m = 0 in the EncounterMeet+ algorithm in Equation 4.

VI. RESULTS

From the friend recommendation study, we present the results. Table 1 shows the aggregated results from the feedback answered by each user for each recommended friend for each task. For EncounterMeet, we used equal weights where $w_m = w_e = 0.5$.

From Table 1, we can clearly see that for having nearly almost the same number of recommendations presented for both tasks (common friend and EncounterMeet friend recommendations). From the EncounterMeet recommendation, users rated a greater number of good recommendations (44.6% vs. 32.1%), knew more of the recommended people (37.3% vs. 24.7%), had a fairly higher number of recommended people already in their phonebook (13.3% vs. 9.8%), had a slightly larger percentage of recommended people in their online SNS (16.9% vs. 14.8%), and accepted a larger percentage of people as friends (50.1% vs. 38.3%), than from the common friend recommendation. This indicates that overall EncounterMeet provided better recommendations to the users than the common friend recommendation. .

A. Reasons for Recommending This Friend

We analyze the comments for each recommendation want to obtain a user perspective as to why a recommended friend was a good recommendation.

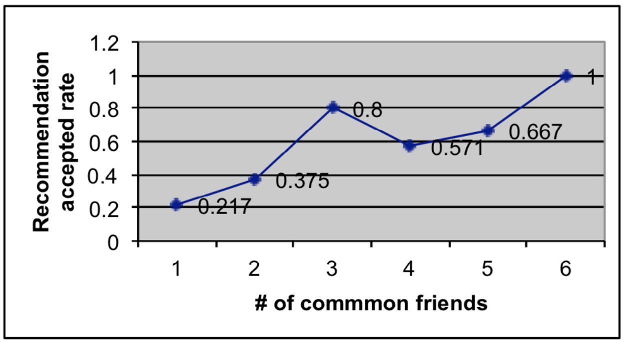


Figure 5. Recommendation acceptance rate vs. number of common friends for the common friend recommendation algorithm

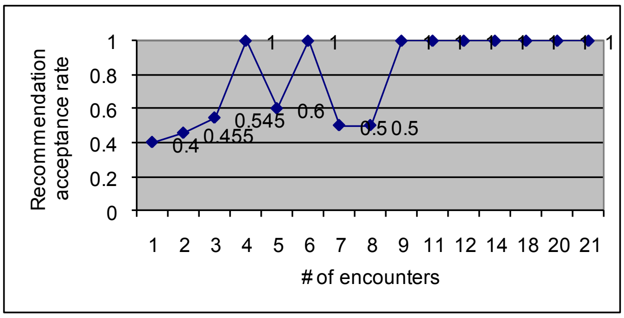


Figure 6. Recommendation acceptance rate vs. number of encounters for

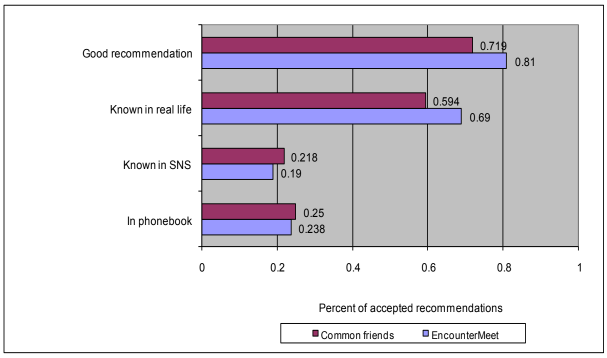


Figure 7. Percent of accepted friend recommendations based on acquaintance for common friends and EncounterMeet recommendation algorithms.

For the common friend algorithm, 45.5% of the users provided reasons for why the suggested friend was a good or bad recommendation. Some of the good reasons were: *"I know him from my friend", "We met in a meeting before", I may have been at a dinner evening where she was present", "She's my neighbor and colleague on the same floor", and "We are in the same group"*. The reasons specified indicate that similar profile, social relationships, co-location and physical proximity are factors in providing good recommendations. For the EncounterMeet algorithm, 32.6% of the users provided reasons for why the suggested friend was a good or bad recommendation. Some of the good reasons were: "*I am more interested in knowing what type of encounters, and even common interests*", "*I know he's from the MSN team, which is a team I work with a lot*", "*my interactions with X shows the actual amount of time. This is important because X is already my friend and I trust his judgment.*" We can see that the reasons here are due to previous meetings, same group, and common content. Therefore, we can clearly see that physical encounters and meetings are important in addition to common friends and similar content and profile, in recommending friends.

B. *Reasons for Not Recommending This Friend*

For the common friend recommendation task, a strong majority (88%) of the people mentioned that why the recommended friend was not good was because they simply did not know that person. In particular, comments such as "*I have no idea why this person is recommended to me? What are the common interests?*" and "*I have no idea who she is. And there is very little info in her profile.*" were common responses. For the EncounterMeet friend recommendation, 46.7% of the people mentioned that the recommended friend was not good because they also did not know that person. Some of the reasons for not recommending the friend include: "*It should be a better recommendation. I know X already and enjoy interacting with him, but the system doesn't know that*", "*interaction distance is too long. Actual interaction time very short. Don't know why recommended*", "*I know her already, and often interact with her but she doesn't have a high recommendation.*" This means that in order to make recommendations more acceptable, the system still needs to record the social interactions, activity and social context as the evidence for recommendations.

C. *Acceptance Rate vs. Frequency and Acquaintance*

We here sought to understand how frequency and acquaintance affects recommendation acceptance. For frequency, we look at the number of common friends and the number of encounters. For acquaintance, we look at how the participant knows that recommended person (which we asked as questions for each friend recommendation). We hypothesize that a participant will accept a friend recommendation more if she has a greater number of common friends, or a greater number of encounters.

For the common friend algorithm, 71.9% of the recommendations that are considered good recommendations are accepted. The more common friends that a user and the recommended friend have then the greater the recommendation acceptance rate as shown in Figure 5. Generally speaking, we observe that users that have more common friends tend to accept more friend recommendations (0.462 Pearson correlation) of the peers, while the number of recommended person's friends and common friends have a similar effect on accepting friend recommendations (0.421 and 0.427 Pearson correlation respectively).

For the EncounterMeet recommendation algorithm, the greater the number of encounters with the recommended friends, then the recommendation acceptance rate of adding that friend is higher (from Figure 6). In addition, we discover that the reason for accepting a friend recommendation is mostly correlated with the number of encounters (0.329), followed by total encounter duration time with the recommended person (0.307), total encounter duration time with others (0.226), number of user's encounters with all others (0.222), and percent of all encounter duration time with the recommended person (0.106).

From Figure 7, a large majority of accepted friend recommendations are due to knowing the recommended person in real life (not unexpected), followed by an almost equal percentage known in SNS and in the user's phonebook.

We also see that from the accepted friend recommendations in Figure 7, the EncounterMeet algorithm provided a higher percentage of good recommendations than the common friends algorithm (81% vs. 72%) and a greater percentage of accepted recommendations are known in real life (69% vs. 59%). Both algorithms provided an equal percentage of accepted recommendations that are known in SNS and are in the user's phonebook.

By looking at the percentage of recommended friends that the user knows, we see that for the most part, good recommendations as rated by the user, are accepted as friends. A high majority of recommended friends that are known in real life and in a user's phonebook are accepted and added as friends, and the percentage is nearly the same for each friend recommendation algorithm.

Therefore, acquaintance does affect whether a user will add this recommended friend, with real connections (contacts in phonebook and known in real life) providing the strong incentive, and is regardless of the type of friend recommendation algorithm. In addition, frequency (common friends, number of friends, number of encounters) does affect the probability of adding the recommended person as a friend.

D. *Good Recommendations vs. Accepted Recommendations*

The next question that we address is whether a good recommendation will influence the probability of accepting this recommendation. From the results, the correlation coefficient between good recommendations and accepted recommendations are strongly positive related with 0.688 Pearson correlation for common friends and 0.741 for EncounterMeet. Therefore, if users think that a person is a good recommendation, then there is a very high probability that she will add this recommended person as a friend. The key challenge, of course, is to design the algorithm that provides the highest number of good recommendations.

VII. SURVEY RESULTS

In this section, we collect the feedback from a survey that we provided to the users of the friend recommendation study, with regards to their experience with friend recommendations and their comments from using our friend recommendation algorithm.

Most participants (80%) found our friend recommendation interface (Figure 4) fairly easy to use. For those who found the interface difficult to use, the reasons included the following: too little information provided about the recommended friend, there was a need to highlight the common friend, and there was a need to add photos. 80% of the participants have used social networking sites (SNS) and 70% add the recommended friend from the SNS some of the time, therefore most are familiar with social networking sites. For those recommendations which they did not add as friends, the reasons included "*I do not know this person*" and "*I was too busy to use SNS*". When

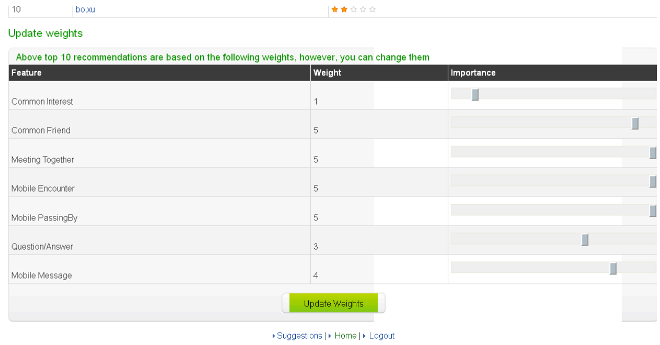


Figure 8. Friend recommendation interface where you can select the weight scores for each feature in the friend recommendation algorithm.

comparing the Find & Connect friend recommendation interface with a popular friend recommendation interface such as Facebook, 44% of them liked our friend recommendation interface, while 33% liked both the Facebook and Find & Connect friend recommendation interface. In deciding whether to add a person as a friend, 90% said physical interactions like encounters and meetings help them to decide, 50% said profile similarity and 40% said common friends.

We finally presented the participants with an interface where they could selectively choose the weight score s_f for each feature (from 1 to 5) as shown in Figure 8, in order to vary the output quality of the friend recommendations.

This algorithm is EncounterMeet+ which we defined earlier. For example, one user may find that she will add friends if she has met them before, whereas another will add friends only if they have common friends. Thus, EncounterMeet+ is able to provide personalized friend recommendation based on user's preferences on the importance of each feature. Each weight w_f for each feature in Equation 1 is calculated as:

$$w_f = s_f / (s_{cf} + s_{ci} + s_m + s_e + s_p + s_{qa} + s_{cm}) \quad (5)$$

From the user study, we wanted to see if participants would like to control the importance of various features as input into the friend recommendation algorithm whereas current algorithms are static. Overall, 80% of the participants did like this, while 10% said they did not like this.

VIII. CONCLUSION

In this paper, we describe a friend recommendation algorithm based on encounters and meetings called EncounterMeet+ to address the problem of improving recommendations based on physical context compared with existing friend recommendations such as common friends. From our user study comparing common friends and EncounterMeet, we discover that EncounterMeet outperformed the common friend algorithm, by providing more good recommendations and more accepted recommendations. We also present our friend recommendation interface using the EncounterMeet+ algorithm combining common friend, similar profile, and shared content with EncounterMeet, and discover that users enjoyed using this algorithm and interface to select their own weight scores for each feature.

For future improvements, we plan to use the participants' feedback for providing the EncounterMeet+ interface along with feature weights and include more known people in the recommendations based on your phonebook and social acquaintance in the enterprise. For future work, we will investigate how to create the ephemeral social network (ESN) from the encounter graph (that we mentioned earlier) and how to integrate the ESN into the recommendation algorithm. In addition, we will modify the recommendation algorithm so that it will learn from the user's past history of recommendations and recommendation feedback, thus automatically adjusting the feature weights without the need for a feature weight interface for user's preferences, as well as adding content-based features. Finally, we will conduct a friend recommendation study with more users in a conference type of environment, and measure its performance. Even though the sample size in this user study is small, nonetheless, these results present an optimistic view that physical context-based friend recommendations such as EncounterMeet and EncounterMeet+ can be used to design improved friend recommendations and thus reduce friend spam.

AKNOWLEDGEMENTS

We thank all the users in our friend recommendation study for their participation and feedback, as well as the Find & Connect team for developing the system to conduct this research.

REFERENCES

[1] Are Your Facebook Friends Really Your Friends? http://www.associatedcontent.com/article/1983858/are_your_facebook_friends_really_your.html.

[2] Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. Group formation in large social networks: membership, growth, and evolution. In *Proc. SIGKDD 2006*, ACM Press (2006), 44-54..

[3] Bahl, P. and Padmanabhan, V.N. RADAR: an in-building RF-based user location and tracking system. In *Proc. INFOCOM 2000*, IEEE Press (2000) ,775-784 vol.2

[4] Chang, L., Chin, A., Wang, H., Zhu, L., Yin, F., Wang, H., and Zhang, L. Enhancing the Experience and Efficiency at a Conference with Mobile Social Networking: Case Study with Find & Connect. Accepted In *Proc. HumanCom 2011*

[5] Chen, J., Geyer, W., Dugan, C., Muller, M. and Guy, I. Make new friends, but keep the old: recommending people on social networking sites. In *Proc. ACM CHI*, ACM Press (2009), 201-210.

[6] Chin, A and Wang, H. Using Cohesive Subgroups for Analyzing the Evolution of the Friend View Mobile Social Network. In Proc. IEEE UIC2010, Springer 2010.

[7] Costa, A., Guizzardi, R., Guizzardi, G. and Filho, J. COReS: Context-aware, Ontology-based Recommender system for Service recommendation. In *Proc. UMICS 2007 (CAISE 2007)*, 1-15.

[8] Counts, S. and Geraci, J. Incorporating physical co-presence at events into digital social networking. In *Proc. ACM CHI 2005*, ACM Press (2005), 1308-1311.

[9] Cranshaw, J., Toch, E., Hong, J., Kittur, A. and Sadeh, N. Bridging the gap between physical location and online social networks. In *Proc. ACM UbiComp 2010,* ACM Press (2010), 119-128.

[10] Debnath, S., Ganguly, N. and Mitra, P. Feature weighting in content based recommendation system using social network analysis. In *Proc. ACM WWW 2008*, ACM Press (2008), 1041-1042.

[11] Eagle, N. and Pentland, A. Social serendipity: Mobilizing social software. *IEEE Pervasive Computing*, *4*, 2 (2005), 28-34.

[12] Froehlich, J., Chen, M., Smith, I. and Potter, F. Voting with your feet: An investigative study of the relationship between place visit behavior and preference. In *Proc. ACM UbiComp 2006*, ACM Press (2006), 333-350.

[13] Gilbert, E. and Karahalios, K. Predicting tie strength with social media. In *Proc. ACM CHI 2009*, ACM Press (2009), 211-220.

[14] Guy, I., Jacovi, M., Shahar, E., Meshulam, N., Soroka, V. and Farrell, S. Harvesting with SONAR: the value of aggregating social network information. In *Proc. ACM CHI 2008*, ACM Press (2008), 1017-1026.

[15] Guy, I., Zwerdling, N., Carmel, D., Ronen, I., Uziel, E., Yogev, S. and Ofek-Koifman, S. Personalized recommendation of social software items based on social relations. In *Proc. ACM RecSys 2009*, ACM Press (2009), 53-60.

[16] Guy, I., Jacovi, M., Perer, A., Ronen, I. and Uziel, E. Same places, same things, same people? mining user similarity on social media. In *Proc. ACM CSCW 2010*, ACM Press (2010), 41-50.

[17] Guy, I., Ur, S., Ronen, I., Perer, A. and Jacovi, M. Do You Want to Know? Recommending Strangers in the Enterprise. In *Proc. ACM CSCW 2011*, ACM Press(2011), 285-294.

[18] Jones, Q., Grandhi, S. A., Terveen, L. and Whittaker, S. People-to-people-to-geographical-places: the P3 framework for location-based community systems. In *Proc. ACM CSCW 2004*, ACM Press (2004), 249-282.

[19] Kwon, J. and Kim, S. Friend Recommendation Method using Physical and Social Context. *International Journal of Computer Science and Network Security 10,* 11 (2010), 116-120.

[20] Liiv, I., Tammet, T., Ruotsalo, T. and Kuusik, A. Personalized Context-Aware Recommendations in SMARTMUSEUM: Combining Semantics with Statistics. In *Proc. SEMAPRO 2007*, IEEE Computer Society (2007), 50-55.

[21] Lindgren, A. and Hui, P. The Quest for a Killer App for Opportunistic and Delay Tolerant Networks. In Proc. ACM MobiCom Workshop on Challenged Networks (CHANTS 2009), ACM Press (2009), 59-66.

[22] Lo, S. and Lin, C. WMR--A Graph-Based Algorithm for Friend Recommendation. In *Proc. IEEE/WIC/ACM WI 2006*, IEEE Computer Society, 121-128.

[23] Moricz, M., Dosbayev, Y.. and Berlyant, M. PYMK: friend recommendation at myspace. In *Proc. SIGMOD* 2010, ACM Press (2010), 999-1002.

[24] Quercia, D. and Capra, L. FriendSensing: recommending friends using mobile phones. In *Proc. ACM RecSys 2009*, ACM Press (2009), 273-276.

[25] SONAR. http://www.sonarapp.com/.

[26] Terry, M., Mynatt, E. D., Ryall, K. and Leigh, D. Social net: using patterns of physical proximity over time to infer shared interests. In *Proc. ACM CHI 2002*, ACM Press (2002), 816-817.

[27] Terveen, L. and McDonald, D. W. Social matching: A framework and research agenda. *ACM Transactions on Computer-Human Interaction (TOCHI)* 12 (2005), 401-434.

[28] Wang, H., Chin, A., and Wang, H. Interplay between Social Selection and Social Influence on Physical Proximity in Friendship Formation. In *SRS 2011 workshop* in conjunction with CSCW 2011

[29] Why some social network services work and others don't- or: the case for object-centered sociality. http://www.zengestrom.com/blog/2005/04/why-some-social-network-services-work-and-others-dont-or-the-case-for-object-centered-sociality.html.

[30] Woerndl, W. and Groh, G. Utilizing Physical and Social Context to Improve Recommender Systems. In *Proc. IEEE/WIC/ACM WI-IATW 2007*, ACM Press (2007), 123-128.

[31] Xie, X. Potential Friend Recommendation in Online Social Network. In *Proc. IEEE/ACM GREENCOM-CPSCOM 2010,* IEEE Computer Society, 831-835.