SocialDining: Design and Analysis of a Group Recommendation Application in a Mobile Context

Mike Gartrell, Khaled Alanezi, Lei Tian, Richard Han, Qin Lv, Shivakant Mishra
University of Colorado Boulder
{mike.gartrell, khaled.alanezi, lei.tian}@colorado.edu

ABSTRACT

Mobile social networks are rapidly becoming an important new domain showcasing the power of mobile computing systems. These networks combine mobile location information with social networking data to enable fully context-aware environments. This paper describes SocialDining, a system that fuses mobile and social data to power novel context-aware recommendation services that provide recommendations to small groups of users who want to meet together for food or drink at local restaurants. We report our analysis on the data collected from 31 users for the SocialDining application over the course of 15 weeks.

1. INTRODUCTION

Recommendation algorithms are an integral part of today's Internet browsing and shopping experience, having been integrated into popular Web sites like Amazon, Netflix, Pandora and Google News [5, 1, 6, 2]. These recommendations are typically targeting individuals, and employ techniques such as collaborative filtering and content-based recommendation, based upon the individual's viewing, purchasing, or rating history.

More recently, recommendation algorithms in the research community have extended beyond the individual to encompass groups of individuals [8, 11, 13, 23, 31, 38, 41, 44, 43]. This is challenging because the different tastes of individuals must somehow be combined using a group consensus function, and it has been shown that this function needs to incorporate such additional contextual factors as the social relationships among group members, their expertise, and the similarity/dissimilarity of tastes.

In this work, we extend this group recommendation research one step further by developing and analyzing a mobile application called SocialDining that recommends restaurants to a mobile group of users. We seek to understand what mobile contextual factors such as location can further influence group decision-making, in addition to the earlier mentioned contextual factors. We describe in the following how we designed SocialDining as an Android application, as shown in Figure 1, that provides a workflow to different mobile groups of users so that they can easily create an invitation for a dining event and then vote on both the time and place of the event - namely at which restaurant they would like to dine. This application was then deployed to over ten different groups of users, who created over a hundred invitations. IRB approval was received for this study. We then analyzed the invitations to explore how factors like sociability, mobility, geographic spread, restaurant proximity, and the host's role in creating the invitation influenced group decision-making, such as the willingness to accept recommendations or complete invitations.

The contributions of this paper are as follows:

- SocialDining represents the first field-based study of group recommendation behavior in the context of a deployed mobile application, which has returned substantial original data.
- The paper describes the design and implementation of a novel mobile group application and system, which provides a workflow that enables different mobile groups of users to easily create a dining invitation and vote on the date and restaurant for a group dining event.
- This work's group recommendation approach constitutes the first integration of social network data with a group consensus function to provide real-time recommendations in a deployed mobile application.
- A novel analysis is presented that examines the impact of such factors as sociability, mobility, restaurant proximity and user preferences on the group decision-making process (e.g. willingness to accept recommendations). A correlation analysis of these factors is also performed.

In the following, we begin by describing related work. Section 3 presents the design and implementation of the SocialDining mobile application and system. Section 4 provides a detailed evaluation of the data obtained from the deployed SocialDining application, which was used by eleven user groups, who created over a hundred dining invitations. We then conclude the paper and describe future work.

2. RELATED WORK

At a high level, recommender systems can be divided into individual-based recommender systems that provide recommendations to individual users and group-based recommender systems that provide recommendations to groups of users.

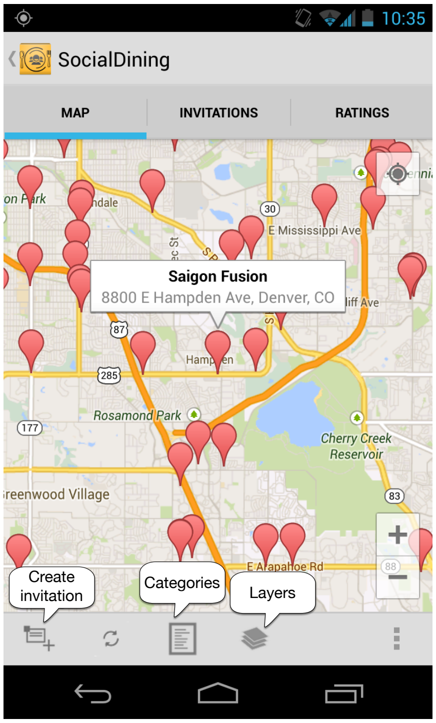

Figure 1: SocialDining main screen.

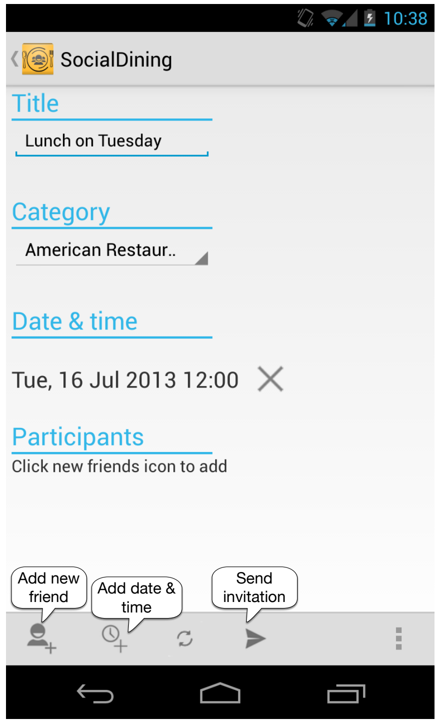

Figure 2: SocialDining invitation creation screen.

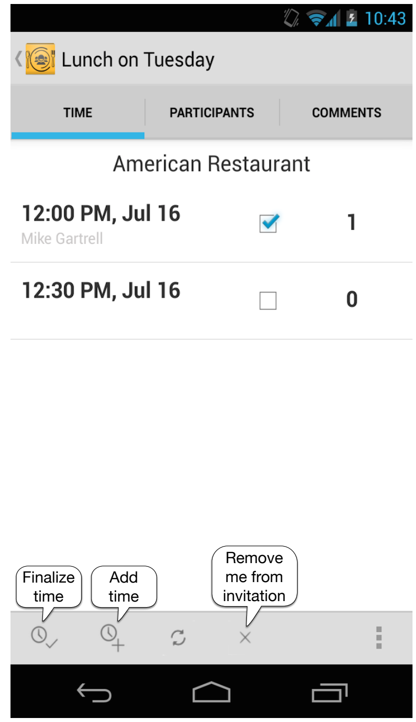

Figure 3: SocialDining invitation time voting screen.

Individual-based recommender systems are generally implemented using one of two approaches: content filtering and collaborative filtering. The content filtering approach builds profiles that describe both users and items. One popular example of content filtering is the Music Genome Project [4] used by Pandora [6] to recommend music. Collaborative filtering relies only on past user behavior (e.g., users' purchase history or ratings on items) without using explicit user and item profiles.

There are two primary approaches to collaborative filtering: neighborhood methods and latent factor models. Neighborhood methods involve computing relationships between items or between users. Item-based neighborhood approaches [14, 25, 37] predict a user's rating for an item based on ratings of similar items rated by the same user. User-based neighborhood approaches [12, 24] predict a user's rating for an item based on the ratings of similar users on the item. Latent factor models on the other hand characterize users and items in terms of factors inferred from patterns in ratings data. Some of the most successful recommender systems that use latent factor models are based on matrix factorization approaches [34, 35, 36, 39, 40].

More recently, a number of approaches to social-based recommender systems that consider relationships between users in social networks have been proposed and evaluated. For example, neighborhood-based approaches to recommendation in OSNs [29, 21, 19, 45] explore the social network and compute a neighborhood of users trusted by a specified user. Using this neighbor, these systems provide recommendations by aggregating the ratings of users in this trust neighborhood. Some latent factor models for OSN-based recommendation have also been proposed [26, 27, 28, 22, 42]. These methods use matrix factorization to learn latent features for user and items from the observed ratings and from users' friends (neighbors) in the social network.

The problem of group recommendation has been investigated in a number of works [8, 11, 13, 23, 31, 38, 41, 44, 43]. Most group recommendation techniques consider the preferences of individual users and propose different strategies to either combine the individual user profiles into a single group (or pseudo user) profile, and make recommendations for the pseudo user, or generate recommendation lists for individual group members and merge the lists for group recommendation. Jameson and Smyth's three main strategies for merging individual recommendations are *average satisfaction*, *least misery*, and *maximum satisfaction* [23]; these form the bedrock of group recommendations [8, 13, 30]. Different weights (like weights of family members) have also been used in aggregation models [10]. A more involved consensus function that utilizes the dissimilarity among group members on top of average satisfaction and least misery strategies, is also plausible [8]. Social connections and content interests can equally be utilized in heuristic group consensus functions [18].

With the prevalence of mobile devices, recommender systems have started to incorporate features such as users' current locations and history of recent places that the users have visited. Current user location has been used as a key criterion for generating restaurant recommendations for individual users in [20]. Current location, time and weather have been used in computing recommendations for individual users in [32]. AGReMo is a recommender system that provides ad-hoc groups of users who want to watch a movie together with shared on-demand recommendations on mobile devices taking into account their current locations [9]. Similarly, a restaurant recommender systems that considers the preferences of group users in a mobile environment is proposed in [33]. However, both of these mobile group-based recommender systems have been evaluated only via in-lab, survey-based user studies, unlike SocialDining, which has been deployed in the field.

3. SYSTEM DESIGN AND ARCHITECTURE

In this section we describe the design, architecture, and implementation of SocialDining. We begin by describing the typical user workflow through a series of use cases, and then present the architecture and implementation of SocialDining.

3.1 Use Cases

3.1.1 A host invites several friends to meet for lunch at an American Restaurant

In the following use case, we describe the actions that a user takes in inviting some friends to meet for lunch at an American restaurant. We call this user the host.

1. The host user begins on the main screen of the SocialDining mobile client, as shown in Figure 1. On this screen, three tabs are apparent. The "Map" tab allows the host to browse nearby restaurants, indicated by red place markers. The user can tap on a place marker to see more information about that restaurant or friend, as shown in the popup balloon for the Saigon Fusion restaurant. If the user taps on the popup balloon for a restaurant, the Foursquare profile Web page for that restaurant will be displayed. The user can pan the map and zoom in and out as desired. The user's location is usually displayed as a small blue triangle, although this is not shown in Figure 1's screen capture. Also, if the appropriate layer is enabled, nearby friends are indicated by blue place markers (not shown).

   The "layers" button on the action bar at the bottom of the screen in Figure 1, shown as a stack of three sheets, can be used to selectively enable or disable display of restaurants and/or friends. The "categories" button immediately to the left of the layers button, shown as a sheet with a list of line items, can be used to show only those restaurants of a particular category, such as Asian restaurants or brewpubs. Finally, the "create invitation" button on the action bar, shown in the bottom left corner of the screen, is used to create and send a new invitation. In this use case, the host user proceeds to create a new invitation by tapping on the "create invitation" button.

2. After tapping on the button to create a new invitation, the screen shown in Figure 2 appears. The host can enter a title for the invitation, specify a restaurant category for the invitation, specify one or more possible dates and times for the invitation, and specify one or more friends that should be included as participants in the invitation. Finally, when the host is satisfied with the invitation settings, the host taps the "send invitation" button, shown as a triangular symbol in the action bar at the bottom of the screen, to send the invitation to all of the selected participants.

3.1.2 A user receives an invitation to meet several friends for lunch at an American Restaurant

In the following use case, we describe the actions that a user takes when he receives an invitation to meet several friends for lunch at an American restaurant.

1. First, the user receives a notification from the SocialDining application on his smartphone indicating that he has received a new invitation. The user responds to this notification and the SocialDining application opens with the time voting screen displayed for this invitation, as shown in Figure 3. The user can express his preferences for the date and time for the invitation by voting on one or more possible options. The proposed dates and times specified by the host during invitation creation are displayed initially. In this use case, the user votes for 12:00 PM on July 16. Any user may add a new proposed date and time to the list of options to vote for by tapping the "add time" button, shown at the bottom of the screen as a clock with a plus sign. Once a user has added a new proposed date and time, this new option is automatically made visible to all other invitation participants for voting. Voting continues until the host finalizes the date and time for this invitation by tapping the "finalize time" button, shown in the bottom left corner of the screen as a clock with a check mark. Only the host is permitted to finalize the date and time for an invitation.

   The user can open the "Participants" tab, shown in Figure 3, to view the list of participants for this invitation. The user can remove himself from the participant list by tapping on the "x" button shown at the bottom of the screen; doing so discontinues further participation from this user in the invitation. Also, the host can add a new user to the invitation from the Participants tab. The list of possible new users to add is populated from the list of that user's Facebook friends who have installed SocialDining.

   In the "Comments" tab shown in Figure 3, a list of comments sent by the participants in this invitation is visible. When the user writes a comment message in the text field at the bottom of this tab (not shown) and submits the comment, the comment is sent to all invitation participants. Each comment is displayed with the name of the user who sent the comment, the comment message, and the time at which the comment was sent.

   In this use case, the host finalizes the date and time for the invitation after several participants have voted.

2. After the host has finalized the date and time for the invitation, each user participating in this invitation receives a notification regarding this action. Upon responding to this notification, the SocialDining application opens with the restaurant voting screen, as shown in Figure 4. The user can express his preferences for the restaurant of the invitation by voting on one or more options. The proposed list of restaurants are initially populated by the group recommendation engine on the SocialDining server, which considers the list of participants for this invitation and the restaurant category specified by the host during invitation creation. This recommended list of restaurants are

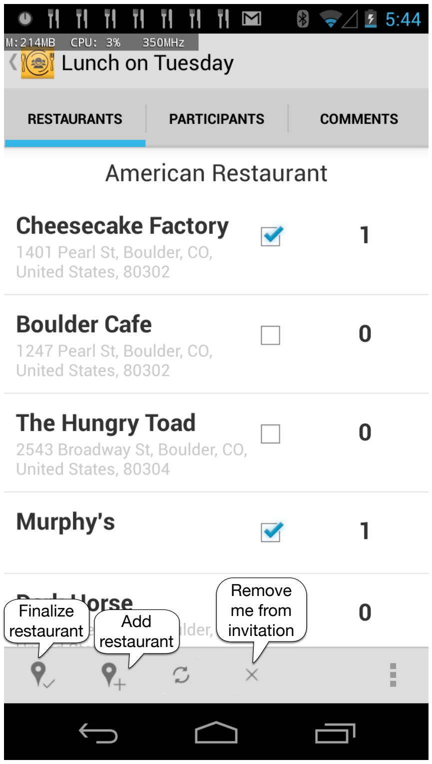

Figure 4: SocialDining invitation restaurant voting screen.

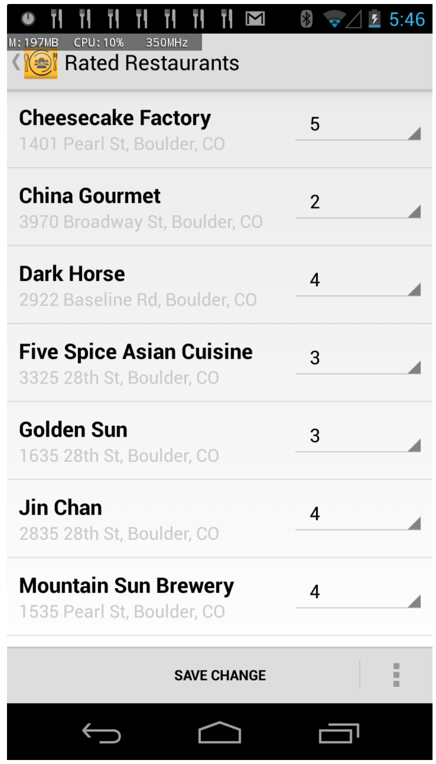

Figure 5: SocialDining restaurant rating screen showing the list of restaurants currently rated by this user.

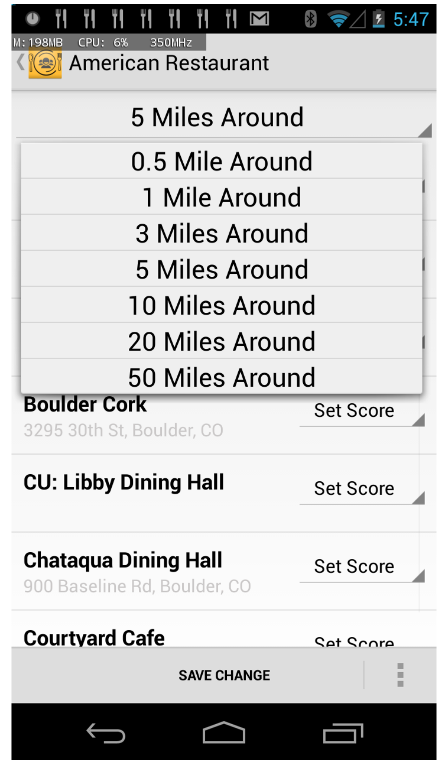

Figure 6: SocialDining restaurant rating screen - search by restaurant category and promity to user location.

ranked in descending order of predicted preference for this group of invitation participants, with restaurant having the highest predicted group preference rating shown at the top of the list. The user may tap on the name of a restaurant to view the Foursquare profile Web page for this restaurant. In this use case, the user votes for the Cheesecake Factory and Murphy's restaurants. Any user may add a new restaurant to the list of voting options by tapping the "add restaurant" button, shown at the bottom of the screen as a place marker with a plus sign. Once a user has added a new proposed restaurant, this new option is automatically made visible to all other invitation participants for voting. Voting continues until the host finalizes the restaurant for this invitation by tapping the "finalize restaurant" button, shown in the bottom left corner of the screen as a place marker with a check mark. Only the host is permitted to finalize the restaurant for an invitation.

In this use case, the host finalizes the restaurant for the invitation after several participants have voted. Three hours after the finalized date and time for the invitation, the SocialDining application prompts the host to enter the "group decision" for this invitation, including information on the name of the restaurant that the group went to for this outing and the group consensus preference rating for this restaurant.

3.1.3 A user provides information on his individual preferences by rating restaurants

In the following use case, we describe the actions that a user takes when he wants to provide information on his individual restaurant preferences by rating restaurants. The SocialDining recommendation engine uses this information when computing restaurant recommendations for invitations.

1. From the main SocialDining screen shown in Figure 1, the user can tap on the "Ratings" tab to open the screen for providing individual restaurant ratings. From this screen, the user can browse restaurants by restaurant category and proximity to the user's current location, or search for restaurants by name, as shown in Figures 6 and 7, respectively. The user can also view his list of currently rated restaurants and modify these ratings, as shown in Figure 5. After setting restaurant ratings, the user taps on the "Save Change" button shown at the bottom of the screen to post the new/modified ratings to the SocialDining server.

3.2 Architecture and Implementation

The SocialDining mobile client is implemented as an Android application, and communicates with the remote SocialDining server, which is implemented as a Java Web application using the Spring application framework [7]. All required functionality to the client is exposed through the server's REST [16] APIs. MongoDB [3] is used to store and manage all data on the server.

The SocialDining recommendation engine is implemented on the server and utilizes the individual and group-based recommender systems described respectively in [17] and [18] to compute restaurant recommendations for each invitation,

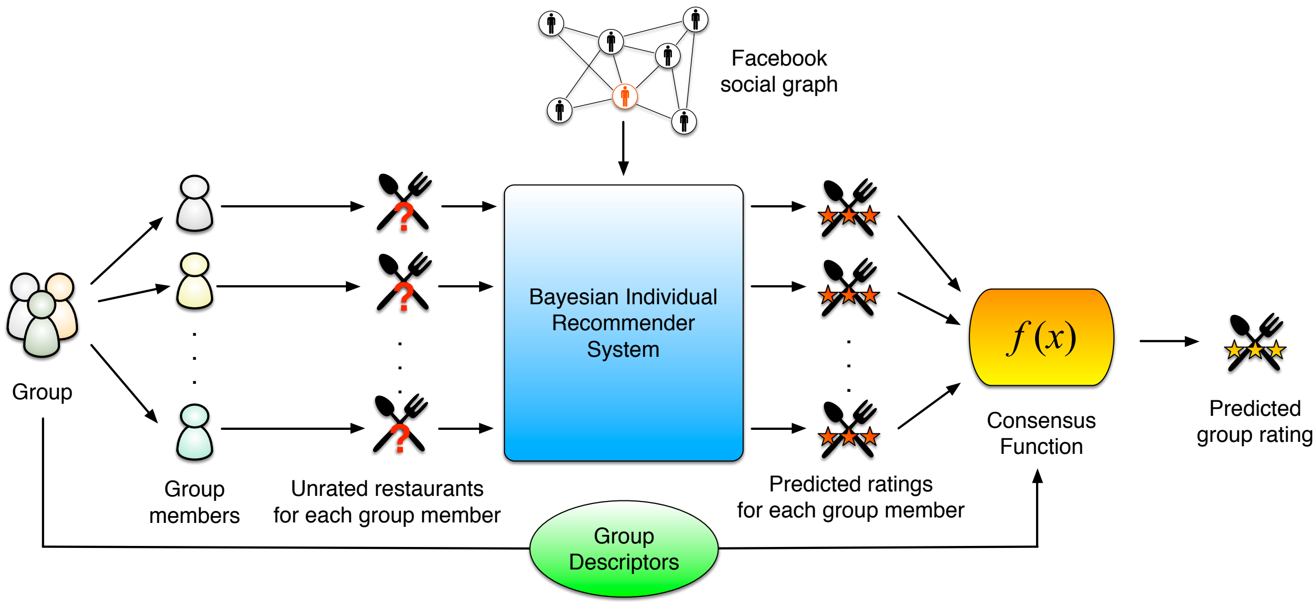

Figure 8: SocialDining recommendation system architecture.

Figure 7: SocialDining restaurant rating screen - search by restaurant name.

based on the group of invitation participants and the specified restaurant category. We show the architecture of the SocialDining recommendation system in Figure 8. The Social Likelihood Bayesian model from [17] is used to compute recommendations for individuals, and the heuristic group consensus function based on average satisfaction from [18] is used to compute recommendations for groups. As described in [17], the Social Likelihood model employs the social graph to enhance recommendation quality. When launching the SocialDining application for the first time after the application is installed, the user is prompted to log in using his Facebook account. This Facebook account information is used to populate each SocialDining user account, including the user's name and Facebook friend list. This friend list is used to populate the social graph maintained internally within SocialDining.

For the group recommendation component, we chose to use the heuristic group consensus function based on average satisfaction since we did not have a measure of social strength between each member of the groups in our user study, and therefore assumed that most group members would share social connections of moderate strength. The expertise and average pairwise dissimilarity groups descriptors are used by the heuristic group consensus function to compute group recommendations; details on these group descriptors are provided in [18].

The information about each restaurant in SocialDining is obtained from Foursquare, including the restaurant name, category, street address, and latitude/longitude coordinates.

4. SOCIALDINING DATA EVALUATION

To investigate the quality of SocialDining recommendations and obtain user feedback on the SocialDining application, we conducted a user study for 15 weeks, from August – December 2012. We recruited 11 groups to participate in this study: eight groups of mutual friends and three romantic-couple groups. Each group was composed of 2 – 5 individuals, with some individuals participating in two groups; a total of 31 individuals participated in this study. Each group participated in the study for a duration of 3 – 5 weeks. The participants were undergraduate stu-

Week	Median Display Rank	Number of completed invitations	Number of completed invitations with group decision recommended	Number of completed invitations where group decision is a user-added restaurant
1	2	7	3	1
2	2	9	4	1
3	2.5	13	5	4
4	2.5	19	8	4
5	3	25	10	6
6	3	33	14	9
7	3	45	20	14
8	3	53	25	17
9	3	57	26	20
10	3	69	35	23
11	3	77	40	26
12	3	87	47	27
13	3	90	47	29
14	2.5	95	48	33
15	2	104	53	37

Table 1: Historical data on completed invitations and recommendations

dents, graduate students, and university staff. Data on approximately 500 restaurants in Boulder was obtained from Foursquare and used to populate the SocialDining restaurant database, and participants were restricted to selecting from these restaurants when using SocialDining in this study. Due to the number of data points collected from this study in terms of invitations, individual user ratings, group ratings, and individual user location traces, all of the results presented in the following sections are statistically significant.

We describe some observations from the data collected during this user study in the next several sections. In the following discussion, a "completed invitation" refers to an invitation where the host user for the invitation has submitted information to SocialDining regarding the restaurant that the group went to for food/drink for this invitation. Recall that the SocialDining application queries the host user for this information three hours after the finalized event date and time for an invitation has passed; this information submitted by the host for an invitation, including the group consensus rating, is referred to as a "group decision" below.

4.1 Application Changes Based on User Feedback

During the preliminary stages of our SocialDining user study we collected feedback from our users via a survey. This survey was administered after users' initial experience with an early version of the SocialDining application that implemented a relatively primitive user interface (UI), lacked functionality for voting on invitation event dates and times, and lacked functionality for formally finalizing the restaurant for an invitation. This survey requested feedback on users' experiences with the UI, workflow of the application, and suggestions regarding new application functionality. We received significant feedback requesting a more refined and elegant UI, and requesting new functionality for scheduling an invitation. Based on this feedback we developed a new version of the SocialDining mobile client with an enhanced UI, new functionality for voting on and finalizing the date and time for an invitation, and new functionality for finalizing the restaurant for an invitation. The use cases described in Section 3.1 reflect the changes made to the SocialDining UI and workflow based on user feedback.

4.2 SocialDining Restaurant Recommendations

Table 1 shows historical data for the invitations completed over the course of our user study, including the number of invitations completed, the number of invitations where the group decision matches a recommended restaurant, and the number of invitations where the group decision matches a restaurant that has been added to the invitation by one of the invitation participants. This data shows that the group decision matches a restaurant recommendation provided by SocialDining for approximately 50% of completed invitations. Of the remaining 50% of completed invitations, the group decision matches a restaurant added to the invitation by a participant about 70% of the time. Therefore, for about 15% of completed invitations, users appear to use a communication channel that does not involve explicit voting on restaurants in the SocialDining app when determining the group decision. This channel may involve comments within the app, or some other mechanism such as SMS text messages, email, etc.

In Table 1 we define "display rank" as the position in the ranked list of SocialDining restaurant recommendations where the group-decision restaurant is found, for an invitation where the group decision matches one of the recommendations. Therefore, we see from this table that for those invitations where the group decision matches a recommendation, the group decision is generally found near the third most highly ranked recommendation, which suggests that the SocialDining recommendation engine performs reasonably well in surfacing relevant recommendations.

4.3 Impact of Restaurant Proximity on Group Decisions

The SocialDining client application posts the user's current location to the server every five minutes, if the application is running in the background, or every 30 seconds,

Group decision matches a recommendation?	User type	Median distance in km between closest user location cluster and group-decision restaurant
Yes	All users	1.335
	Host users	1.726
	Non-host users	1.495
No	All users	1.146
	Host users	1.360
	Non-host users	1.608

Table 2: Impact of restaurant proximity to user location on invitation group decisions

Group decision matches a recommendation?	Median maximum distance in km between users during the 2 hours preceding each invitation
Yes	3.33
No	2.75

Table 3: Impact of user spread on invitation group decisions

Group decision matches a recommendation?	Mean number of group meetings per user per week
Yes	9.67
No	4.44

Table 4: Impact of group meeting frequency on invitation group decisions

if the app is running in the foreground. The user may disable location tracking at any time in the application, which prevents the application from posting location data to the server.

We leverage the user location data recorded on the server to investigate the impact of user location on group decision behavior in SocialDining. First, we apply the DBSCAN clustering algorithm [15] to find spatial clusters in the temporally-ordered location trace data for each user who participated in our study. The DBSCAN ϵ parameter is set to 1.0 km, and the parameter for the minimum number of points required to form a dense region is set to 40. We found that these DBSCAN parameter settings find sensible clusters in our location trace based on a visual inspection of these clusters plotted on a map, and these clusters appear to correspond to locations frequented by our study participants, such as work, school, home, etc. Next, for each completed invitation, and for each invitation participant, we identify the participant user location cluster that is closest to the group decision restaurant for that invitation, with the requirement that this cluster must contain a point with a timestamp that occurs within two hours before or after the finalized event date and time for the invitation.

Table 2 shows the median distance between the closest invitation participant location cluster and the group decision restaurant for those initiations where the group decision either matches or does not match one of the recommendations provided by SocialDining. Note that some invitations were omitted, due to lack of user location clusters satisfying the requirements described above. We see from this table that for all users, the median distance between the closest user location cluster and group decision restaurant is approximately 14% smaller for those invitations where the group decision does not match a recommendation as compared to when the group decision matches a recommendation. Therefore, we infer that for those invitations where the invitation participants do not elect one of the recommendations, restaurant proximity to the user's location may be an important factor. For example, we would intuitively expect that users would prefer to go to lunch at restaurants that are close to their place of work or school at certain times, such as a weekday afternoon. Furthermore, we see from Table 2 that when considering only the host user of each invitation, the median distance delta between the closest host user location cluster and group decision restaurant is 21% smaller for those invitations where the group decision does not match a recommendation. However, when considering only *non-host* invitation participants, the effect is reversed, and median distance between the closest non-host participant location and the group decision restaurant location is 7% farther away when a recommendation is not used for the group decision compared to when a recommendation is used. Therefore, we can conclude from this analysis that when a group decides not to use a recommendation and selects a nearby restaurant, the group generally decides to select a restaurant closer to the host and not closer to any of the other participants. This suggests that the host may carry more influence in the group decision-making process when the group decides to not use a recommendation.

4.4 Impact of User Spread on Group Decisions

Having investigated the impact of restaurant proximity on group behavior, we now turn to an analysis of the impact of the distance between users prior to an invitation, which we term the "user spread". Table 3 shows how the maximum distance between invitation participants in the two hours preceding an invitation impacts group decisions. We see that when groups do not use a recommendation for the group decision, users tend to be somewhat closer to each other or less spread than when groups decide to use a recommendation. This suggests that when users are separated by smaller distances/spreads, they are less likely to use recommendations provided by SocialDining.

4.5 Impact of User Sociability on Group Decisions

Next, we examine the "sociability" of our users, which we define as the number of times that users within a group physically meet with each other within a specified time period. We estimate this sociability metric by using the user location clusters to infer meetings between group members. We use the following criteria to define a group meeting:

1. More than 50% of the members of a group must be present.
2. The group members must be at approximately the same location for at least 15 minutes.
3. The location clusters for each group member must be within 10 meters of each other.

We see from Table 4 that for invitations where the group decision matches a recommendation, participants in these invitations meet with each 118% more often than participants in invitations where a recommendation is not used for the group decision. This suggests that users who meet more frequently are more likely to use recommendations in SocialDining. Table 5 shows that for invitations that are completed (i.e., a group decision is submitted), participants in these invitations meet with each other 99% more frequently than those who participate in invitations that are not completed. This result suggests that users who meet more often are more likely to complete an invitation.

4.6 Impact of User Mobility on Group Decisions

We now turn to examining the impact of user mobility on group behavior in SocialDining. Here we define user mobility as the total distance traveled by a user in the 24-hour period preceding an invitation. Table 6 shows how user mobility impacts whether or not the group decision matches a recommendation. We see that for host users, mobility is 37% higher for invitations where the group decision matches a recommendation. However, for non-host users, slightly higher mobility is associated with groups who do not select a recommendation. We see from Table 7 that user mobility is substantially higher for groups who complete invitations compared to groups who do not complete invitations. These results indicate that highly mobile host users may be influencing groups toward using recommendations when group decisions are made.

4.7 Impact of User Preference on Group Decisions

Next we compare individual user preferences to group preferences in SocialDining. Recall that each submitted group decision contains information on which restaurant the group went to for an invitation, along with a group consensus rating for that restaurant. We conduct this analysis by computing the root mean square error (RMSE) between the group preference rating for a restaurant and the individual participant preference ratings for this restaurant, for each completed invitation where this rating data is available. The results shown in Table 8 indicate that host user preferences are generally closer to group preferences, which suggests host users may have higher influence than non-host users on the group decision making process. Host user preferences are closest to group preferences when a recommendation is not used for the group decision, which indicates that hosts may carry more influence in this scenario and override a recommendation in favor of a restaurant that they strongly prefer.

	Mobility	Sociability	Proximity	User Spread	Preference
Mobility		-0.10	-0.08	-0.08	-0.01
Sociability	0.37		-0.09	0.22	-0.35
Proximity	0.32	-0.08		-0.01	-0.05
User Spread	-0.06	0.02	0.06		-0.03
Preference	0.18	0.24	0.11	0.30	

(light shading)	Used Recommendation
(dark shading)	Did not use Recommendation

Table 9: Correlation Matrix

4.8 Correlation Between Factors

Table 9 shows a correlation matrix between many of the factors we have discussed previously, computed using the Pearson product-moment correlation coefficient for completed invitations where the group decision either matches or does not match a recommendation provided by SocialDining. In this table, "proximity" in this table refers to restaurant proximity as discussed in Section 4.3, "mobility" refers to user mobility as discussed in Section 4.6, "user spread" refers to the distance between users as discussed in Section 4.4, and "preference" refers to the RMSE between individual user preferences and group preferences as discussed in Section 4.7. We see that the two highest positive correlations are between sociability and mobility and between proximity and mobility when a recommendation is not used. The relatively high positive correlation between sociability and mobility implies that users who meet more often with other group members (higher sociability) tends to be associated with traveling greater distances (higher mobility), when a group decision does not match a recommendation. This association is fairly causally transparent, since we might expect that users who meet more frequently with others would often have to travel more to arrive at those meetings. The relatively high positive correlation between restaurant proximity and mobility implies that greater distances between invitation participants and the group decision restaurant is associated with traveling greater distances, when a recommendation is not used. This association is also fairly causally transparent, since we would expect that greater distances between users and group decision restaurants would require those users to travel greater distance to reach those restaurants.

5. CONCLUSIONS AND FUTURE WORK

In this work we have presented the design, implementation, and evaluation of SocialDining, a novel mobile group application that provides a workflow for groups to manage invitations for dining events. Our analysis of data collected

Invitation completed?	Mean number of group meetings per user per week
Yes	7.10
No	3.57

Table 5: Impact of group meeting frequency on invitation completion

Group decision matches a recommendation?	User type	Median distance in km traveled by users during the 24 hours preceding each invitation
Yes	All users	11.41
	Host users	6.13
	Non-host users	8.30
No	All users	10.27
	Host users	4.49
	Non-host users	8.48

Table 6: Impact of user mobility on invitation group decisions

during a field-based user study provides substantial evidence that host users have a significant influence on the group decision making process in SocialDining. We see evidence of the host's influence in restaurant proximity to user location, user mobility, and user preference. Additionally, restaurant proximity to users and user proximity to other users in a group play an important role in the group decision making process. These are all key factors impacting group behavior dynamics that should inform the design of group recommendation systems in the SocialDining application domain.

Further analysis is required to investigate temporal patterns in the data we have collected. We would also like to conduct user studies at large scale, involving at least hundreds of participants and thousands of completed invitations. The data collected from such a large-scale study would allow us to further investigate individual and group preference behavior in this domain, particularly regarding how location and other contextual factors impact preferences. Additionally, a larger dataset would facilitate the development and evaluation of a probabilistic model for group recommendation customized for this application. We intend to pursue the development of a new group recommendation system that leverages the insights regarding mobile group behavior that we have obtained from our work.

6. REFERENCES

[1] Amazon. `http://www.amazon.com`.

[2] Google news. `http://news.google.com`.

[3] Mongodb. `http://www.mongodb.org/`.

[4] The music genome project. `http://www.pandora.com/mgp.shtml`.

[5] Netflix. `http://www.netflix.com`.

[6] Pandora. `http://www.pandora.com`.

[7] Spring framework. `http://projects.spring.io/spring-framework/`.

[8] S. Amer-Yahia, S. B. Roy, A. Chawla, G. Das, and C. Yu. Group recommendation: Semantics and efficiency. In *Proc. of VLDB 2009*.

[9] C. Beckmann and T. Gross. AGReMo: providing ad-hoc groups with on-demand recommendations on mobile devices. In *Proceedings of the 29th Annual European Conference on Cognitive Ergonomics*, 2011.

[10] S. Berkovsky, J. Freyne, and M. Coombe. Aggregation trade offs in family based recommendations. In *Proc. of the 22nd Australasian Joint Conference on Advances in Artificial Intelligence*, 2009.

[11] L. Boratto, S. Carta, A. Chessa, M. Agelli, and M. L. Clemente. Group recommendation with automatic identification of users communities. In *Proc. of WI-IAT '09*, 2009.

[12] J. Breese, D. Heckerman, C. Kadie, et al. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th conference on Uncertainty in Artificial Intelligence*, pages 43–52, 1998.

[13] L. de Campos, J. Fernández-Luna, J. Huete, and M. Rueda-Morales. Managing uncertainty in group recommending processes. *Journal of User Modeling and User-Adapted Interaction*, 19, 2009.

[14] M. Deshpande and G. Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.

[15] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, volume 96, pages 226–231, 1996.

[16] R. T. Fielding and R. N. Taylor. Principled design of the modern web architecture. *ACM Transactions on Internet Technology (TOIT)*, 2(2):115–150, 2002.

[17] M. Gartrell, U. Paquet, and R. Herbrich. A bayesian treatment of social links in recommender systems. CU Technical Report CU-CS-1092-12, 2012.

[18] M. Gartrell, X. Xing, Q. Lv, A. Beach, R. Han, S. Mishra, and K. Seada. Enhancing group recommendation by incorporating social relationship interactions. In *Proceedings of the 16th ACM international conference on Supporting group work*, GROUP '10, pages 97–106, New York, NY, USA, 2010. ACM.

[19] J. Golbeck. *Computing and Applying Trust in Web-based Social Networks*. PhD thesis, University of Maryland College Park, 2005.

[20] T. Horozov, N. Narasimhan, and V. Vasudevan. Using location for personalized poi recommendations in mobile environments. In *Proceedings of the International Symposium on Applications and the Internet*, 2006.

[21] M. Jamali and M. Ester. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 397–406, New York, NY, USA, 2009. ACM.

[22] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys 2010, pages 135–142, New York, NY, USA, 2010. ACM.

Invitation completed?	User type	Median distance in km traveled by users during the 24 hours preceding each invitation
Yes	All users	10.54
	Host users	5.89
	Non-host users	8.30
No	All users	1.66
	Host users	0.57
	Non-host users	3.15

Table 7: Impact of user mobility on invitation completion

Group decision matches a recommendation?	User type	Mean RMSE
Yes	All users	0.8663
	Host users	0.7714
	Non-host users	0.7985
No	All users	0.6871
	Host users	0.5
	Non-host users	0.7122

Table 8: Impact of the difference between individual and group consensus preferences

[23] A. Jameson and B. Smyth. Recommendation to groups. In *The adaptive web: methods and strategies of web personalization*, 2007.

[24] R. Jin, J. Chai, and L. Si. An automatic weighting scheme for collaborative filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 337–344. ACM, 2004.

[25] G. Linden, B. Smith, and J. York. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.

[26] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR 2009, pages 203–210, New York, NY, USA, 2009. ACM.

[27] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceeding of the 17th ACM conference on Information and knowledge management*, CIKM 2008, pages 931–940, New York, NY, USA, 2008. ACM.

[28] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM 2011, pages 287–296, New York, NY, USA, 2011. ACM.

[29] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, RecSys '07, pages 17–24, New York, NY, USA, 2007. ACM.

[30] J. Masthoff. Group modeling: Selecting a sequence of television items to suit a group of viewers. *Journal of User Modeling and User-Adapted Interaction*, 14(1), 2004.

[31] M. OConnor, D. Cosley, J. A. Konstan, and J. Riedl. Polylens: A recommender system for groups of users. In *European Conference on Computer-Supported Cooperative Work*, 2001.

[32] M.-H. Park, J.-H. Hong, and S.-B. Cho. Location-based recommendation system using bayesian user's preference model in mobile devices. *Lecture Notes in Computer Science*, 4611:1130–1139, 2007.

[33] M.-H. Park, H.-S. Park, and S.-B. Cho. Restaurant recommendation for group of people in mobile environments using probabilistic multi-criteria decision making. *Lecture Notes in Computer Science*, 5068:114–122, 2008.

[34] J. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, pages 713–719. ACM, 2005.

[35] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning*, pages 880–887. ACM, 2008.

[36] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20:1257–1264, 2008.

[37] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.

[38] D. Sprague, F. Wu, and M. Tory. Music selection using the partyvote democratic jukebox. In *Proc. of AVI 2008*, 2008.

[39] N. Srebro and T. Jaakkola. Weighted low-rank approximations. In *Proceedings of 20th International Conference on Machine Learning*, volume 20, pages 720–727, 2003.

[40] D. H. Stern, R. Herbrich, and T. Graepel. Matchbox: large scale online bayesian recommendations. In *Proceedings of the 18th international conference on World wide web*, pages 111–120. ACM, 2009.

[41] E. Vildjiounaite, V. Kyllï£¡ï£¡nen, and T. Hannula. Unobtrusive dynamic modelling of tv programme preferences in a finnish household. *Journal of Multimedia Systems*, 15, 2009.

[42] L. Yu, R. Pan, and Z. Li. Adaptive social similarities for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, RecSys 2011, pages 257–260, New York, NY, USA, 2011. ACM.

[43] Z. Yu, Z. Yu, X. Zhou, and Y. Nakamura. Handling conditional preferences in recommender systems. In *Proc. of IUI 2009*.

[44] Z. Yu, X. Zhou, Y. Hao, and J. Gu. Tv program recommendation for multiple viewers based on user profile merging. *Journal of User Modeling and User-Adapted Interaction*, 16(1), 2006.

[45] C. Ziegler. *Towards Decentralized Recommender Systems*. PhD thesis, University of Freiburg, 2005.