PROCEEDINGS OF THE 11th ANNUAL CONFERENCE ON WORLD WIDE WEB APPLICATIONS

2-4 September 2009
Port Elizabeth
South Africa

Editor:
P.A. van Brakel

Publisher:
Cape Peninsula University of Technology
PO Box 652
Cape Town
8000

Proceedings published at
http://www.zaw3.co.za

ISBN: 978-0-620-45215-1

TO WHOM IT MAY CONCERN

The full papers were refereed by a double-blind reviewing process according to South Africa's Department of Education (DoE) refereeing standards. Papers were reviewed according to the following criteria:

- Relevancy of the subject to Web applications
- Explanation of the research problem & investigative questions
- Quality of the literature analysis
- Appropriateness of the research method(s)
- Adequacy of the evidence (findings) presented in the paper
- Standardised referencing style.

The following reviewers took part in the process of evaluating the full papers of the 11th Annual Conference on World Wide Web Applications, held from 2 tot 4 September 2009 in Port Elizabeth:

Prof J Brits
Faculty of Information Sciences (Dean)
University of Wisconsin
Milwaukee
USA

Prof T Carr
Centre for Higher Education Development
University of Cape Town
Cape Town

Prof J Cronjé
Faculty of Informatics and Design (Dean)
Cape Peninsula University of Technology
Cape Town

Mr Craig de Beer
Web Management Services
Massey University
Palmerstone
New Zealand

Prof M Erasmus
Management Information Systems
Erasmus Associates
Lynnwood
Pretoria

Dr AM El-Sobky
Educational Sciences
RITSEC
Cairo
Egypt

Dr MWH Labour
Laboratory of the Sciences of Communication
University of Valenciennes and Hainaut Cambrésis
Valenciennes Cedex
France

Prof S Mutula
Department of Information Science
University of Botswana
Botswana

Dr David Raitt
European Space Agency
The Hague
The Netherlands

Prof A Singh
Department of Business Information Systems
University of KwaZulu-Natal
Durban

Prof P Weimann
Economics and Social Sciences
University of Applied Sciences
Berlin
Germany

Further enquiries:

Prof PA van Brakel
Conference Chair: Annual Conference on WWW Applications
Cape Peninsula University of Technology
Cape Town
+27 21 469 1015 (landline)
+27 82 966 0789 (mobile)

A model for a socially-aware mobile tourist recommendation system

J. Leonard
Academy for Information Technology,
University of Johannesburg
Johannesburg, South Africa
digidot@telkomsa.net

M. Coetzee
Academy for Information Technology,
University of Johannesburg
Johannesburg, South Africa
marijkec@uj.ac.za

Abstract

People, whether for business or pleasure, often travel or visit locations they are unfamiliar with. To overcome this, a mobile tourist recommendation system can be used. This system is able to generate tourist attraction recommendations for a person, based on a number of criteria such as time and location. One criterion often overlooked though, is a person's social environment. This is an important consideration as touring is often done with more than one person. By not considering the other people within a person's social environment, recommendations will only be limited to the preferences of a single person. The purpose of this paper is to propose a model for a mobile tourist guide recommendation system that considers a persons social environment when generating tourist attraction recommendations. This is done by studying other tourist recommendation systems, eliciting key requirements for a socially aware tourist guide system from them, and using these requirements as input for a mobile tourist recommendation system model that can harness a person's social environment when recommending tourist attractions.

Keywords: mobile tourist guide, group recommendation, trust based recommender.

1. Introduction

Tourists often visit new places knowing very little about the best tourist attractions within a location. This means tourists often miss out on tourist attractions which would make their visit more enjoyable. A possible solution to this problem is to make use of a mobile tourist guide recommendation system. Such systems have the ability to recommend tourist attractions to a user or a group of users given specific criteria. Common examples of criteria include location, time, and personal preferences. For instance, if a tourist were to query the system for all restaurants within walking distance (location), open in the evenings (time), and serving Italian cuisine (personal preference), the system should provide a list of all restaurants matching that criteria.

The problem with many tourist recommendation systems is that recommendations are usually based on the inputs of a single tourist without considering their social environment (Ardissono, Goy, Petrone, Segnan & Torasso, 2003; Buriano, 2006). The concern with this is that touring is often done with other people, and these other people greatly affect system recommendations (Buriano, 2006). For example, a restaurant recommendation would be different if you were going to dine with friends than if you were going to dine with your wife. Therefore, the consideration of a tourist's social environment in a mobile tourist guide system is an important one.

The purpose of this paper is to propose a model for a socially-aware mobile tourist guide system that focuses on the recommendation of tourist attractions to groups of people. In doing this, the paper is ordered as follows. Section 2 explains the various types of recommendation systems and motivates which one is favourable for a socially-aware mobile tourist guide system. Section 3 looks at related work with the purpose of eliciting requirements for the system model. Section 4 explores the key requirements of these systems with reference to a proposed architecture for a socially-aware mobile tourist guide system. Section 6 concludes the paper.

2. Recommendation systems

RSs (Recommendation Systems) are defined as systems which aid the user in evaluating an item, based on suggestion or influence from the system (Hinze & Junmanee, 2005). Typically, most RSs fall into one of three generic categories: collaborative filtering, content-based filtering, or trust-based recommendations systems. This section discusses each type of RS and concludes by specifying the most suitable for a socially-aware mobile tourist RS. Each type of RS is briefly introduced with each being defined and having their main advantages and disadvantages being presented.

Collaborative filtering RSs produce recommendations by comparing a user profile (an information repository containing a user's details and preferences) with one or more other user profiles (Avesani, Massa & Tiella, 2005; Hinze & Junmanee, 2005; Golbeck & Hendler, 2006). A recommendation is then returned based on how closely a user profile matches another user profile. On the other hand, a *content based RS* uses the user profile differently. Instead of using user profile similarity to generate recommendations, a user profile is compared with information about an item (Hinze & Junmanee, 2005). It's important to note that, in this system, the user profile is formed differently. Here a profile's formed by harnessing the information from items previously rated by the user. Therefore, a recommendation is returned based on how closely an item matches items already rated by the user.

Trust-based RSs are different from both collaborative filtering and content based RSs in that no profile matching is performed. Instead, recommendations are generated by harnessing trust relationships within social networks (Avesani, et al., 2005; Golbeck & Hendler, 2006). The premise of these systems is that users are more likely to trust recommendations given by those users they know and trust, than those users they do not (Avesani, et al., 2005; Golbeck & Hendler, 2006). Consequently, the recommendations returned by the system will all be dependant upon how much you trust the raters of those recommendations.

Below is Table 1 summarising this discussion, with the advantages and disadvantages of each system listed.

Table 1: Advantages and disadvantages of recommendation systems

Type of Recommendation System	Description	Advantages	Disadvantages
Collaborative Filtering	Produces recommendations by comparing a user profile with one or more other user profiles (Avesani, Massa & Tiella, 2005; Hinze & Junmanee, 2005; Golbeck & Hendler, 2006)	Recommendations can be produced even if a user has never assigned a rating to any item before as profile similarity is being determined (Hinze & Junmanee, 2005)	A large base of user ratings must exist. If not, then system recommendations will be limited and of minimal benefit to the user (Hinze & Junmanee, 2005)
Content Based Filtering	Produces recommendations by comparing a user profile with information about an item (Hinze & Junmanee, 2005).	Items can be recommended to the user without having an existing user ratings base (Hinze & Junmanee, 2005)	Recommendations will usually be limited to those items previously rated by the user (Hinze & Junmanee, 2005)
Trust Based	Generates recommendations by harnessing trust relationships within social networks (Avesani, et al., 2005; Golbeck & Hendler, 2006)	Greater accuracy with recommendations can be produced when trust is used to determine profile similarity (Avesani, et al., 2005)	Inferred trust is dependant upon the fact that there is at least one path between the users. If there is none, trust can't be calculated (Golbeck & Hendler, 2006).

Motivating a recommendation system for a socially-aware mobile tourist guide

In terms of a socially-aware mobile tourist guide system, the type of RS suited best would be a combination of a collaborative filtering and a trust-based filtering RS. The advantage of a collaborative filtering RS is that recommendations are based on profile similarity and not rating history (Hinze & Junmanee, 2005). With regards to a socially-aware mobile tourist guide system, this is important for a number of reasons. Firstly, in this context, you want all system tourist attractions to be considered, independent of whether a user has reviewed one or one hundred. With content based filtering RSs, this can't be done because recommendations are constrained by rating history (Hinze & Junmanee, 2005). Secondly, with regards to the formation of recommendation groups, it is also important to consider a user profile independent of their rating history. By doing this, more accurate and diverse recommendations can be calculated by the system because each individual's needs are analysed (Coyle, Mccarthy, Mcginty, Nixom, Salamo & Smyth, 2006; Hinze & Junmanee, 2005).

One weakness of collaborative filtering RSs is the assumption that profile similarity will produce an accurate recommendation. The problem with this assumption is that it is exclusive of people who aren't similar to your profile, even though they may possess valid recommendations. Trust is, therefore, a better indication of a good recommendation than profile similarity as the people we trust may not be the ones most similar to us (Avesani, et al., 2005; Golbeck & Hendler, 2006).. Therefore, by combining trust and collaborative filtering we have a more complete RS.

In the next section, a discussion on the key requirements for a collaborative, trust-based system is covered. These requirements are elicited in the next section where a study of other related systems is done to identify key implementations and any oversights. The findings are then used as the basis for the requirements of a socially-aware mobile tourist guide system. This study is performed in the next section.

3. Related work

This section overviews three tourist guide systems with the purpose of eliciting requirements for a mobile tourist guide system. In the following sections, three systems are discussed: INTRIGUE (Interactive Tourist Information Guide) (Ardissono, et al., 2003), Travel Decision Forum (Jameson, 2004), and Moleskiing (Avesani, Massa & Tiella, 2005). Once a study of these systems has been completed, requirements for a proposed mobile tourist guide system model are then elicited. The section concludes with these findings.

3.1 INTRIGUE (interactive tourist information guide)

INTRIGUE is a content based mobile tourist RS for people visiting the Italian city of Torino (Ardissono, et al., 2003). Although INTRIGUE offers other tourist related features, its main focus is on the provision of recommendations to groups with conflicting and differing interests (Ardissono, et al., 2003). The implementation of this feature is the focus of this section.

INTRIGUE achieves the generation of group recommendations by separating a group of users with conflicting interests into subgroups of users with similar interests (Ardissono, et al., 2003). This is achieved by following a number of steps. Firstly, the host user manually places each user into a particular sub group. The users in each sub group are then given a registration form to fill in. Once completed, a sub group profile is formed, by using the information from the registration form, and comparing it to a set of stereotypes. INTRIGUE's typical stereotypical models are age, visual impairment, and disabilities. The stereotype matching the sub group profile closest is then chosen. When each sub group's profile has been formed, each one is merged and used in calculating a group recommendation (Ardissono, et al., 2003).

3.2 Travel decision forum

The Travel Decision Forum is a prototype system that recommends holiday destinations to groups of users (Jameson, 2004). The system focuses on enabling a group of users to come to a common consensus in terms of a system proposal. Therefore, the point of it is not that the system will recommend a particular location, but that a joint preference group model will be formed through negotiation and debate (Jameson, 2004).

Within the system, it is easy for other group members to see how happy or not each group member is with the proposal (Jameson, 2004). This is achieved by providing a facility for group members to disagree with a particular proposal. In doing this, users have a platform to either convince other group members as to why a proposal should be accepted or rejected or to explain a user's line of reasoning. Therefore, when a disagreement comes about, the user can either change their own personal preferences to bring them into line with the group, or they could suggest a new recommendation to the group. A final group recommendation is generated when the group come to an agreement on a holiday destination (Jameson, 2004).

3.3 Moleskiing

Moleskiing is an online skiing RS, developed by Avesani, Massa & Tiella (2005) that makes use of trust to generate system recommendations. Contextually, the system is used by skiers who consult the system before skiing. By identifying ski routes reviewed by trustworthy skiers, other skiers are able to determine whether a specific route is safe or not (Avesani, et al., 2005).

Typically, a skier would consult the system the day before they decide to choose a specific route (Avesani, et al., 2005). The review of that route would include things like the snow conditions and safety in general. The trustworthiness of the rater is also determined, to verify that the review is a valid one. Trustworthiness is determined by the trust ratings of the reviewer's peers. Upon reading the review, the skier would then decide whether they should ski on the specific route or not. If they decide to ski on that route, they would then write a review themselves and update the rating for that route. Therefore, a final recommendation is agreed upon by the user based on a global trust level of the raters of that item, in this case a ski route (Avesani, et al., 2005).

3.4 Requirements for a socially aware mobile tourist guide system

In view of the above systems, a number of considerations have been identified. To form a set of requirements for a mobile tourist guide system model. In this section, each requirement is briefly explained with the key requirements being further discussed in the next section. These requirements follow.

- **The importance of trust.** Trust is a metric often overlooked in many RSs. While providing recommendations for users, systems often don't verify the reliability of the recommendation. Even within a group context, it is important that the recommendations are verified to be trustworthy and reliable (Avesani, et al., 2005; (Golbeck & Hendler, 2006). Trust is a metric that can be used for this purpose.

- **Group recommendations.** As noted already, touring is an activity often done in a group context with other people (Ardissono, et al., 2003; Buriano, 2006). Therefore, RSs should be able to cater for the provision of recommendations to groups. This, however, is not an easy task. The main difficulty is the fact that groups are made up of users with different needs and preferences (Ardissono, et al., 2003; Coyle, Mccarthy, et al., 2006). In order to provide group recommendations though, social context must be considered (Buriano, 2006). This typically involves a relevant group aggregation algorithm where each individual's preferences are considered and weighted within the group (Ardissono, et al., 2003; Coyle, et al., 2006). This is similar to INTRIGUE, which weighs factors such as disabilities and visual impairment (Ardissono, et al., 2003).

- **Dynamic group formation**. A mobile tourist RS should be able to form groups dynamically with minimal user intervention. This will be less obtrusive for group members and there will be no cumbersome manual user selection process required by the group creator. In considering this, it is also important that the group creator should also have the ability to explicitly define groups should they wish to do so (Backlund-Norberg, M., Hallberg, J., Kristiansson J., Nugent, C., & Synnes, K., 2007).

- **Recommendation transparency**. When a user receives a recommendation, it is important that the system explains to the user how it came to that recommendation (Ardissono, et al., 2003; Tintarev, 2007). Recommendations are complex and if users can see what logic the system used to come to the given recommendation, a greater trust in the system is developed (Ardissono, et al., 2003; Tintarev, 2007). Especially in a group context, recommendation transparency also enables group members to come to a consensus on recommendations because each knows why a recommendation suits certain members better than others (Jameson, 2004; Mccarthy, Mcginty, Salamo, & Smyth, 2006).

- **Recommendation consensus**. Linked to recommendation transparency, it is important for groups to have a public facility whereby each is able to see one another's satisfaction rating (Jameson, 2004; Mccarthy, et al., 2006). This rating is usually an indication as to how much each user will or won't enjoy a recommended destination. In addition, group members should then have a platform where they can voice their concerns and come to a consensus on a group recommendation (Mccarthy, et al., 2006). If group members are face to face, this would usually be done by talking to one another, but if group members are not close by, a platform should be available for group members to discuss the recommendation (Jameson, 2004).

In the next section, the elicited requirements are applied to a proposed, socially-aware mobile tourist guide system model. The main requirements are discussed in greater detail so that they can be better understood. This discussion takes place from an implementation perspective, with the purpose of seeing how each requirement is harnessed by the system model to generate group recommendations that are both trustworthy and reliable.

4. Proposed system architecture

In this section, a proposed model for a socially-aware mobile tourist guide system is discussed. This model implements the requirements of the previous section with the purpose of generating reliable group recommendations. Due to the complexity of this, the group requirements generation process is split into a number of steps:

- **Group formation**. In the first step, the group for which a recommendation will be generated, needs to be formed. This process is discussed in section 4.2.

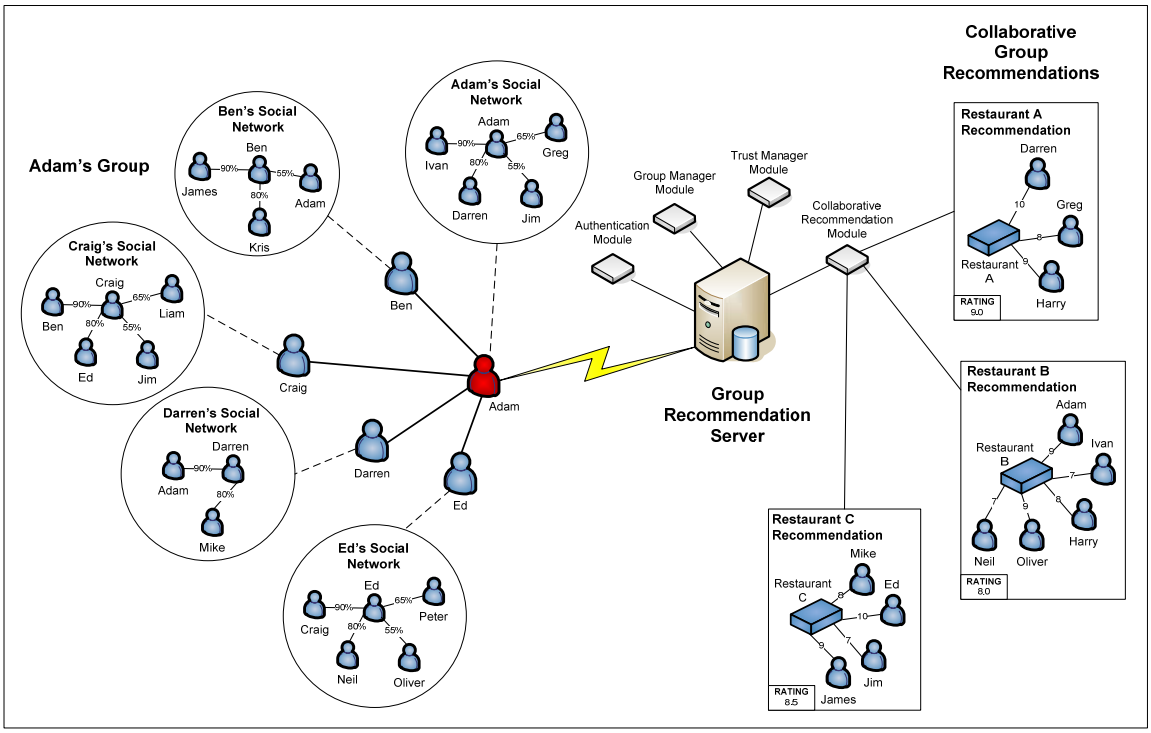
- **Collaborative group recommendation**. In the second step, a group recommendation is generated. It is explained how a group profile is used to do a collaborative search and generate a list of group recommendations. This is discussed in section 4.3.

- **Determine rating group trustworthiness**. In the third step, the reliability of the group recommendation is calculated. This is discussed in section 4.4.

- **Generate a trust-based group recommendation**. In the last step, the final group recommendation is generated after determining the reliability of the group members themselves. This is discussed in section 4.5.

Because there are so many steps in generating a reliable group recommendation, a scenario is used to describe the workings of each section. This scenario is presented in section 4.1 and is used to reflect an instance of the proposed model. The model, with reference to the scenario, is shown below in figure 1.

Figure 1: Proposed system architecture with reference to a case study



4.1 Scenario

Consider the following scenario. Adam (the highlighted user in figure 1) has just come out of a movie after watching it with his friends Ben, Craig, Darren, and Ed. As they talk about the movie amongst themselves, Adam notices that it's quite late in the evening and, therefore, offers to take his friends out to dinner as a treat. He consults his mobile tourist guide system for a restaurant recommendation at which they can dine.

Adam reaches into his pocket and takes out his mobile device. He selects the application and logs in. Adam then chooses the recommendation section and the option to generate a new system recommendation. Once he has selected the option to generate a new recommendation, he finds that he's taken to the group formation screen.

4.2 Group formation

The first step in generating a reliable group recommendation is to form a group for which that recommendation can be generated. This is an important step as, any group recommendation must reflect the preferences of the individual members of the group (Coyle, et al., 2006; Hinze & Junmanee, 2005). Considering one user too many or too few within a group can influence the validity of a group recommendation.

Groups can be formed in two generic ways: implicit and explicit group formation (Backlund-Norberg, M., et al., 2007). Implicit group formation is the formation of a group with minimal user intervention. Conversely, explicit group formation is the formation of a group through direct user invention. In the above architecture both methods are considered as each is relevant in different

circumstances. Normally, there are two cases to consider. The first case is when group members are face-to-face. The second case is when group members aren't physically in the same location (Backlund-Norberg, M., et al., 2007).

When group members are face-to-face, it's far easier for a group administrator to use an implicit form of group formation. In the architecture, this is done using Bluetooth (a short-range, wireless communication medium common too many modern mobile devices) (Backlund-Norberg, M., et al., 2007). From an implementation perspective, Bluetooth is used as a communication medium with all nearby group members having their user profiles downloaded to the group administrator's mobile device. This user profile is usually a lightweight file (normally in XML) containing a user's personal information and preferences. An example user profile is shown below. This file can only be downloaded by the group administrator upon approval by the relevant group member. Once there though, it is processed with the group members name being extracted and placed in a list of valid group members. When completed, the list can either be confirmed or edited by the group administrator.

Figure 2: Sample user profile for a restaurant recommendation

```
<GroupRecommender>
    <UserProfileExample>

        <Name>Adam</Name>
        <Surname>Anderson</Surname>
        <DOB>3 August 1980</DOB>

        <RestaurantRecommendation>
            <PreferenceOne>Italian</PreferenceOne>
            <PreferenceTwo>Chinese</PreferenceTwo>
            <PreferenceThree>Portuguese</PreferenceThree>
            <Atmoshphere>Relaxed</Atmosphere>
            <Cost><R500</Cost>
            <ForKids>True</ForKids>
            <LiveEnertainment>N/A</LiveEnertainment>
            <OccassionType>Informal</OccassionType>
            <OutsideSeating>N/A</OutsideSeating>
        </RestaurantRecommendation>

    </UserProfileExample>
</GroupRecommender>
```

In situations where group members aren't nearby, a different approach to the above is required as Bluetooth is restricted to short distances. Therefore, an explicit form is required where a group administrator would manually add group members to a list (Backlund-Norberg, M., et al., 2007). In the architecture, this is catered for by allowing the group administrator to do a search of all registered users on the database. Once the group member has been found, it's added to the list of valid group members. When all the necessary names have been added, the list can then be confirmed.

In terms of the scenario with Adam, the first approach would be used. Adam would ensure each group member had their Bluetooth functionality enabled and then request a search for all nearby system users. In doing this, all of Adam's friends are prompted as to whether they will allow him access to their user profile. Because each person knows Adam, they accept. After a

short wait, the system picks up all nearby users and forms a dynamic group list. Adam double checks that all his friends have been added and then confirms the list. This is now Adam's group (as shown in figure 1).

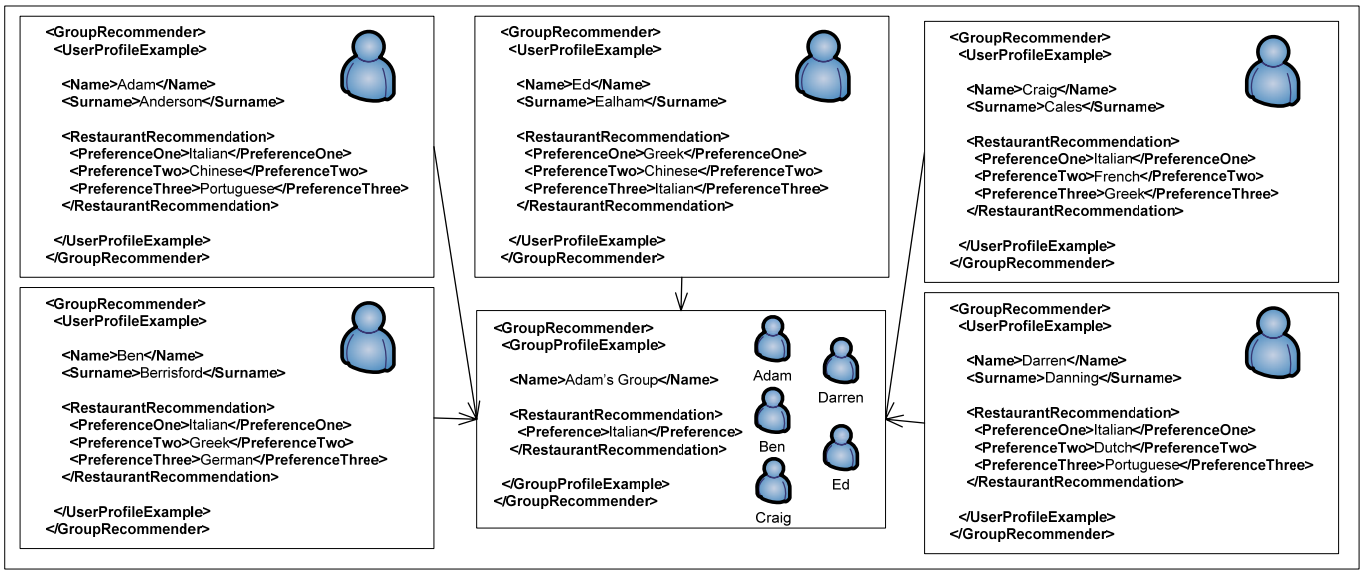
4.3 Collaborative group recommendation

Once a list of group members has been accepted, the next step is to form a group profile and use this profile as the basis for a collaborative group recommendation (Coyle, et al., 2006). The purpose of this collaborative search is to generate a base list of recommendations that can be verified by a group. Verification is performed in the next section where a group makes use of trust to determine how reliable the generated list of recommendations is. To do this, though, a recommendation list that meets the preferences of the group needs to firstly be performed. This process is discussed in this section.

As previously mentioned, the difficulty in generating group recommendations is managing differing user preferences (Ardissono, et al., 2003; Coyle, et al., 2006). Therefore, it is a difficult task to ensure each group member has at least some of their preferences considered (Ardissono, et al., 2003; Coyle, et al., 2006). This is why it is important for each group member's user profile to be individually analysed with special attention given to those who need to specifically be catered for such as children and disabled people (Ardissono, et al., 2003; Coyle, et al., 2006; Hinze & Junmanee, 2005). A suitable aggregation algorithm is consequently required to consider such aspects. The algorithm should consider the user profiles of each user and merge the relevant fields for that specific recommendation, such as the merging of cuisine preferences for a restaurant recommendation (Coyle, et al., 2006). This process is best shown with reference to the scenario.

Assume Adam requires a group recommendation for his group. In the background, unknown to him, a group profile is being formed that will be sent to the server. Because he requests a restaurant recommendation, only the relevant fields of each group member's profile is considered. Consider the user profiles in figure 3 for example. Because the group requests a restaurant recommendation, only the cuisine preferences of each group member are considered when forming a group profile. Consequently, when a group profile is formed, only the relevant fields are merged. Figure 3 shows how a group profile for Adam's group is generated. In it, the user profiles of Adam's friends, Ben, Craig, Darren, and Ed are aggregated to form a group profile. The user profiles of each are shown in figure 3.

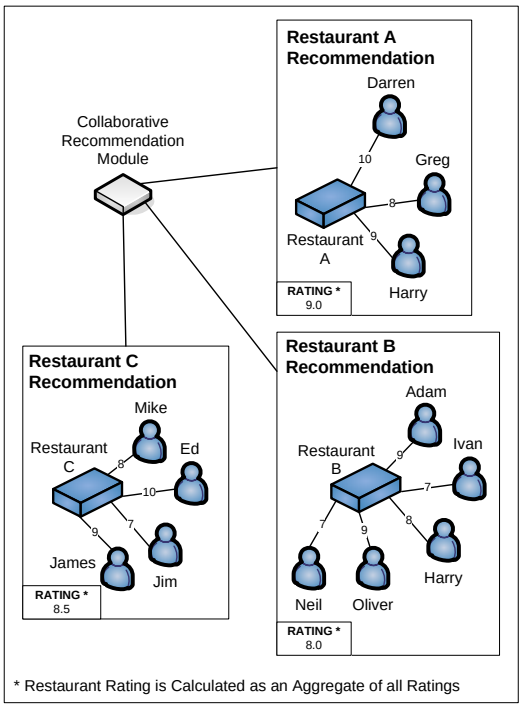
Figure 3: Aggregation of profiles to form a group profile



The difficulty in aggregating the user profiles is catering for differing preferences (Ardissono, et al., 2003; Coyle, et al., 2006). For instance, Ed's cuisine preference is for Greek, while everyone else prefers Italian. The difficulty is determining how to consider Ed in the group profile. In this instance, it's not too difficult to do this. Because Ed's second preference is Italian, the cuisine preferences of each group member can be aggregated to have Italian as the group preference. Therefore, the compromise is small. The final, aggregated group profile, after merging the user profile's of Adam's friends, is shown in figure 3.

When fully processed, the group profile is sent, from the group administrator's mobile device to the server, for a collaborative search to be done. This is the duty of the collaborative recommendation module shown in figure 1. The purpose of this module is to take a group profile as input, perform a collaborative search, and return a prioritised list of recommendations. The collaborative recommendation module does this by determining profile similarity between the group profile and all system users. When the group profile is calculated to be similar to another user's profile, the best recommendations submitted by that user are analysed. The same is done for each system user. Therefore, once the collaborative search is complete, the extracted recommendations from similar users to the group are merged to form a comprehensive recommendations list. Each recommendation in this list will contain the recommended item, the list of raters for that item, and their associated ratings. For example, Adam's group profile would generate a list like the one shown in figure 4.

Figure 4: Collaborative restaurant recommendation list for Adam's group



This figure shows the top three restaurant recommendations (restaurant A, B, and C) returned after a collaborative search done for Adam's group. Consider the recommendation for restaurant A. This recommendation contains three users who assigned a rating to restaurant A. Darren assigned the restaurant a rating of 10, Greg a rating of 8, and Harry a rating of 9. These users aren't necessarily similar to the group, but they are the users who assigned the highest ratings for restaurant A. The highest raters are considered as restaurant A was calculated to meet the preferences of Adam's group profile best when the collaborative search was performed. The same logic is applied to the recommendation for restaurant B and restaurant C.

4.4 Trustworthiness of the rating group

In the previous step, the formation of a collaborative group recommendation list was discussed. This is the first step in generating a reliable group recommendation. The second step is to verify the trustworthiness of the returned recommendations list. The purpose of this is to determine how reliable the collaborative group recommendations are. In other words, are the returned recommendations really the best recommendations for the group? In this section, the methodology of performing this step is discussed.

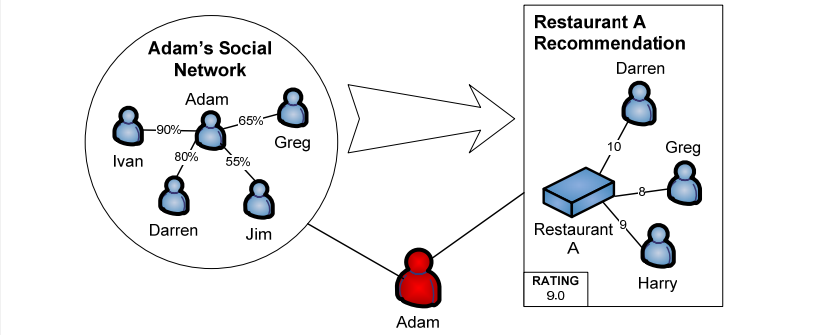
Once the collaborative search has been completed, the recommendations list is returned to the group administrator. Upon receiving this, each group member is forwarded the same recommendation list. This is done so that each group member is able to check the reliability of the recommendation based on their own personal preferences (Avesani, et al., 2005; Golbeck & Hendler, 2006).

To determine the trustworthiness of a recommendation, the trust levels between the individual group member and each individual rater contained within that recommendation (i.e. each

member of the rating group) is determined (Avesani, et al., 2005; Golbeck & Hendler, 2006). This is done by leveraging the group member's social network (Avesani, et al., 2005; Golbeck & Hendler, 2006). This social network is contained on the user's mobile device and could simply be a list of contacts or a social graph, as shown in figure 5 below. To explain the use of trust in determining the reliability of a recommendation, the scenario is once again referenced.

Assume Adam and all his friends have the list of restaurant recommendations on their mobile devices. In line with figure 4, Adam and his friend's would have recommendations for restaurant A, B, and C. Because restaurant A's recommendation has the highest rating, its reliability is determined first by each member of the group. For the purposes of this scenario, the reliability calculation will only be considered from Adam's perspective, though each group member will follow the same logic. Therefore, consider figure 5 below.

Figure 5: Determining the trustworthiness of rating group A (from Adam's perspective)



Trustworthiness of Rating Group from Adam's Perspective

User	Trust Level	Recommendation Rating	Calculated Rating (Trust Level * Recommendation Rating)
Darren	80% (Explicit)	10	8.0
Greg	65% (Explicit)	8	5.2
Harry	90% (Inferred)	9	8.1

As illustrated, two groups can be seen. The first group is the social network of Adam. This group contains all system users Adam has explicitly assigned a trust rating to. These users are Darren, Greg, Jim, and Ivan. The second group is the collection of users who have assigned a rating to restaurant A. This group contains Darren, Greg, and Harry. The first step in determining reliability is to calculate trust between Adam and each member of the rating group, which contains Darren, Greg, and Harry. The architecture does this in two steps. It first checks for any explicit levels of trust using the social network (Golbeck & Hendler, 2006). Then, for rating group members that don't have explicit trust levels defined, one is inferred (Avesani, et al., 2005; Golbeck & Hendler, 2006). For instance, in the above diagram Adam has defined explicit trust levels for Darren (80%) and Greg (65%), both within his social network. For Harry though, no trust level exists. Therefore, this will have to be inferred.

Inferred trust is a calculation performed on the server by the trust manager shown in figure 1. Using an inference algorithm, the trust manager takes in two inputs: the social network of a user and the user for which the trust rating will be inferred (Golbeck & Hendler, 2006). The trust manager then uses a relevant trust inference algorithm and calculates a trust level (Avesani, et al., 2005; Golbeck & Hendler, 2006). When completed, this rating is returned to the user who

requested the inferred rating. Once a trust level exists between the user and each member of the rating group, the next step in determining recommendation reliability can be performed. In the scenario shown in figure 5, Harry has been inferred a rating of 70%. Because Adam now has all the necessary trust ratings, the next step can thus be performed.

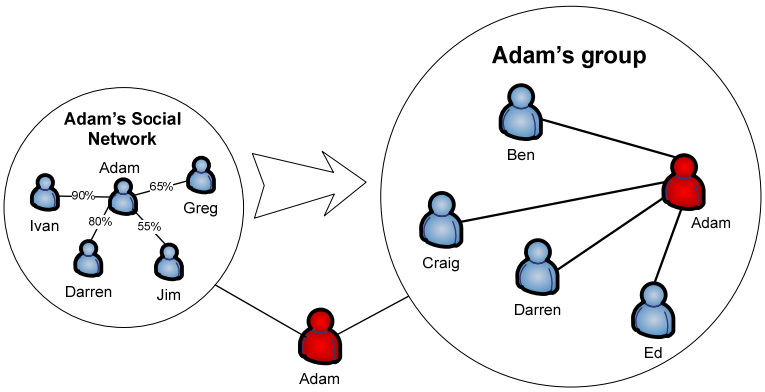
In order to determine recommendation reliability, the trust ratings of each rating group remember is used to generate a calculated rating for the relevant user (Avesani, et al., 2005; Golbeck & Hendler, 2006). This rating is essentially a statement saying, 'based on my trust level for these raters, I would assign the following rating to this item' (Golbeck & Hendler, 2006). This calculated rating is compared with the overall rating given for that item. If there is a large difference between them, it can be assumed that, based on the user's trust levels for that rating group, they probably wouldn't enjoy the recommended item. An example calculation can be shown with reference to a sample algorithm. Let's assume a simple calculation like: calculated rating per user = Trust Level * Rating. In order to determine the calculated for A, you would apply the algorithm to each member of the rating group and average the results to determine a final rating. Therefore, for Darren, Greg, and Harry, you'd take Adam's trust levels for each of them and multiply it by their rating for restaurant A. The calculated ratings for each are shown in figure 5. By averaging the results, you would determine a calculated rating for A. Therefore, you'd take Darren, Greg, and Harry's calculated ratings and divide it by 3 to get Adam's calculated rating. Therefore, Adam's final rating would be (8.0 + 5.2 + 8.1) / 3 = 6.5. This rating means that, based on Adam's trust levels for Darren, Greg, and Harry, he would most likely assign a rating of 6.5 to restaurant A.

4.5 Trust-based group recommendation

In the previous two steps, a prioritised collaborative group recommendation list was formed with its reliability being verified by each group member using trust. In the final step of the recommendation process, the trust level of the group members themselves are taken into account. This is the consideration that the people you go out with have a large bearing on your own experience. Therefore, after this step, the recommendation would finally be calculated to be reliable or not. In terms of the architecture therefore, reliability means that after considering the reliability of the recommendation's rating as well as the company you're going out with, how far apart is the recommendation of each group member when compared with the recommendation's rating. The final step of the recommendation process, from the architecture's perspective, is discussed below.

In order to account for the effect of the group itself, a similar logic as the previous section is applied (Avesani, et al., 2005; Golbeck & Hendler, 2006). The only difference being that the logic's applied to the Adam's group (Ben, Craig, Darren, and Ed) and not the rating group for restaurant A consisting of Darren, Greg, and Harry. Therefore, to determine how much you'd enjoy the recommendation with the group, you need to see how reliable the calculated ratings (as performed in the previous step) for each group member are. Again, this calculation would be done by taking the calculated rating for Ben, Craig, Darren, and Ed, and multiplying it with Adam's trust level for each one. The results for this are shown in figure 6. In order to determine Adam's final rating, he would average all the group calculated ratings (as in the previous section). Therefore, his final rating would be (6.5 + 4.32 + 4.875 + 6 + 6.4) / 5 = 5.619. This final rating means that if Adam were to go to restaurant A with his friends, based on his trust for those who have rated restaurant A and based on the trust he has for his friends, he would give restaurant A, a rating of 5.619. Therefore, for Adam, restaurant A would be a poor recommendation as it was originally given an overall recommendation of 9.0, but Adam's rating is only 5.619.

Figure 6: Determining the trustworthiness of Adam's group (from Adam's perspective)



Trustworthiness of Adam's Group from Adam's Perspective

User	Trust Level	Calculated Rating	Group Calculated Rating (Trust Level * Recommendation Rating)
Adam	100% (Default)	6.5	6.5
Ben	60% (Inferred)	7.2	4.32
Craig	75% (Inferred)	6.5	4.875
Darren	80% (Explicit)	7.5	6
Ed	80% (Inferred)	8.0	6.4

Once the final rating for all group members has been calculated, they are publicly displayed on the mobile device for each member to see (Jameson, 2004; Mccarthy, et al., 2006). This is so that each group member will be able to see how satisfied they would be going to the recommended tourist attraction and also to see how satisfied the rest of the group would be (Jameson, 2004; Mccarthy, et al., 2006). This allows group members to deliberate about recommendations and try to convince other group members to either shy away or pursue a recommended item (Backlund-Norberg, M., et al., 2007; Jameson, 2004). When the group finally comes to a consensus, it's the responsibility of the group administrator to reject the recommendation and look at the next one on the list, or approve it. Should they reject it, the next recommendation is sent to the mobile device of the group administrator and the process is started over again.

In this proposed architecture, three steps were covered. The first step was forming a group profile and doing a collaborative search using that profile to obtain a prioritised collaborative recommendation list for the group. The second step was determining the reliability of the recommendation by leveraging the trust relationships between the individual group members and the rating group members (Avesani, et al., 2005; Golbeck & Hendler, 2006). Finally, a final reliability rating was determined by using the calculated rating in the second step and the trust relationships amongst the group members themselves (Avesani, et al., 2005; Golbeck & Hendler, 2006). This resulted in a final rating for each group member, publicly displayed for each one to see (Jameson, 2004). The final rating considered the reliability of the raters as well as the reliability of the group itself. This rating could then be compared with the recommendation rating to determine how relevant that rating would be for the group.

5. Conclusion

The purpose of this paper was to present a socially-aware mobile tourist guide system which

generated reliable group recommendations. This was done by following covering a number of sections. Firstly, the concept of a RS was introduced with the three generic types of RSs being discussed. The conclusion of this section was that a mixture of a collaborative and trust-based RS was a good choice for a mobile tourist guide system. In the following section, related mobile tourist guide systems were studied with the purpose of eliciting key requirements. These requirements were used as the basis for the proposed architecture in section 4. In this section, the architecture was presented and discussed in depth. It was shown that by using trust relationships, reliable group recommendations can be generated. The process as to how this is done from an implementation perspective was outlaid using a scenario.

In conclusion, the implementation of trust is seen to be an effective methodology of determining reliable recommendations (Avesani, et al., 2005; Golbeck & Hendler, 2006). This is based on the premise that if you trust someone, you will value their opinion more highly than someone else who you trust less (Avesani, et al., 2005; Golbeck & Hendler, 2006). This fact was harnessed in the architecture to provide an effective solution to a number of problems. Firstly, recommendations of reliable recommendations can be generated for groups. Secondly, the reliability of a recommendation can be verified by the entire group using trust. Lastly, each group member had a large say in the final system recommendation with the ability to review and debate recommendations. In covering each of these issues, it's believed that the proposed architecture provides an effective solution to the problem of reliable group recommendations.

In future work, the proposed architecture will be explored in greater depth. This will mainly be done from an implementation and a more practical perspective. Consequently, we'd like to develop algorithms for the system so that reliable group recommendations can be generated. Also, we'd like to develop a prototype of the system and determine the accurateness and user satisfaction of the recommendations generated by the system. This is where our future research will be focused.

6. References

Ardissono, A., Goy, G., Petrone, M., Segnan, M. & Torasso, P. 2003. INTRIGUE: personalized recommendation of tourist attractions for desktop and handset devices. *Applied Artificial Intelligence, Special Issue on Artificial Intelligence for Cultural Heritage and Digital Libraries*, 17(8-9): 687-714.

Avesani, P., Massa, P. & Tiella, R. 2005. A trust-enhanced recommender system application: Moleskiing. In: *Proceedings of the 2005 ACM symposium on Applied computing*, March 13 -17, 2005. New Mexico: Santa Fe: 1589--1593.

Backlund-Norberg, M., Hallberg, J., Kristiansson J., Nugent, C, & Synnes, K. 2007. Creating dynamic groups using context-awareness. In: Proceedings of the 6th International Conference on Mobile and Ubiquitous Multimedia (MUM'07)*, December 12-14, 2007. Oulu, Finland: 42-49*

Buriano, L. 2006. Exploiting Social Context Information in Context-Aware Mobile Tourism Guides. Available WWW: http://www.mobileguide06.di.unito.it/pdf (Accessed 1 February 2008).

Coyle, L., Mccarthy, K., Mcginty, L., Nixon, P., Salamo, M. & Smyth, B. 2006. Group

recommender systems: a critiquing based approach. In: *IUI '06: Proceedings of the 11th international conference on Intelligent user interfaces*, January 29 – February 01, 2006. Sydney, Australia: 267-269.

Mccarthy, K., Mcginty, L., Salamo M, & Smyth, B. 2006. Social Interaction in the CATS Group Recommender. In: *Workshop on the Social Navigation and Community based Adaptation Technologies*, June 2006. Dublin, Ireland. Available WWW: http://www.sis.pitt.edu/~paws/SNC_BAT06/proceedings.html.

Golbeck, J. & Hendler, J.. 2006. FilmTrust: Movie recommendations using trust in web-based social networks. In: *Proceedings of the IEEE Consumer Communications and Networking Conference*, 8 -10 January 2006. Las Vegas, USA: 282-286.

Hinze A. & Junmanee, S. March 2005. Advanced recommendation in a mobile tourist information system. Available WWW: http://hdl.handle.net/10289/97 (Accessed 7 January 2009).

Jameson, A. 2004. More than the sum of its members: challenges for group recommender systems. In: *Proceedings of the International Working Conference on Advanced Visual Interfaces*, May 25 – 28, 2004. Gallipoli, Italy: 48–54.

Tintarev, N. 2007. Explanations of Recommendations. In: *Proceedings of the 2007 ACM conference on Recommender systems*, October 19 – October 20, 2007. Minneapolis, USA: 153-156.