A Context-based

Personalized Ratings Management System

by

Cheewee Ang

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degrees of

Bachelor of Science in Electrical Engineering and Computer Science

and

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 23, 2001

[June 2001]

Author____
Department of Electrical Engineering and Computer Science
May 23, 2001

Certified by____
Professor Peter Szolovits
Thesis Supervisor

Accepted by____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

A Context-based Personalized Ratings Management System

by
Cheewee Ang

Submitted to the
Department of Electrical Engineering and Computer Science

May 23, 2001

In Partial Fulfillment of the Requirements for the Degrees of
Bachelor of Science in Electrical Engineering and Computer Science
and
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

In an online community of any size, every user garners reputations for different contexts through interactions with other users. When any user in the online community is interested in a certain context, the user would value the opinions of those users of high repute with respect to that context. However, if the online community is of any significant size, a user might not know every other user in the community. Therefore, a reputation brokering mechanism needs to be incorporated so that an interested user would be able to trace paths through other users to the reputable users. From social network theories, there exist centrality measures that can be used to determine the reputation of users in a network based on the number of indegrees and outdegrees. However, different users in the network can have varying tastes and opinions with respect to a given context. Since centrality measures determine the reputation of users based on aggregate opinion, a user who has different tastes from the majority of the other users might not agree with the reputable users selected by the centrality measures. This justifies the need for developing personalized rating systems that are able to personalize for any user a selection of other users that he would regard as highly reputable. In this thesis, two such rating systems are developed and compared against the existing centrality measures. When tested over various dimensions such as network size and network connectivity, there is evidence that the personalized rating systems perform better than the traditional measures of reputation in the selection of reputable individuals.

Thesis Supervisor: Peter Szolovits
Title: Professor, MIT Department of Electrical Engineering and Computer Science

Acknowledgements

First of all, I would like to thank my thesis supervisor, Professor Peter Szolovits. He was always helpful in providing me with insightful ideas during my research. Having discussions with him often seeded my mind with new ideas and directions for my work.

I would also like to thank Lik Mui, a very talented Ph.D. student in the Clinical Decision-Making Group at LCS. It was through initial discussions with him that brought me to this area of research for my M.Eng. thesis. Throughout the year, he proved to be a most enthusiastic colleague and enviable friend. He helped me design the structure of my thesis and the scope of my research. I was also introduced to many new subject areas, cutting-edge software and technology through him. His energy and drive was a constant source of inspiration for me.

Last but not least, I would like to thank everyone in the Clinical Decision-Making Group whom I had discussions with at one time or another, especially Mojdeh Mohtashemi, who took time to review my thesis and provided me with much useful feedback.

Table of Contents

List of Figures

1. Introduction

In a networked environment where selectively available resources are distributed widely, much effort has been expended in discovering effective ways of locating relevant resources. However, information about the quality of the retrieved resources is often unavailable. For instance, many search engines of today use keyword search in a document to determine its relevance. Anyone with no specialized knowledge about a subject area could create a totally superfluous website about it, and these search engines would have trouble rejecting the article in response to a query about the subject area. What seems to be missing from the picture is a mechanism to determine the reliability of the source, which would be the author of the website in the above example. Such a source-sanctioning process relies on an intuitive notion of source reputation as permeated within a community of users.

The proliferation of online communities has resulted in a marked increase in the number of interactions carried out over the web. An online community is defined as a set of interacting parties that participates in a specific class of transactions online. For example, the napster community is the set of users that engage in the uploading and downloading of files using the napster software. Online communities exist because their members can provide services for one another, such as problem solving skills in HelloBrain.com [20]. Besides services, members can also provide recommendations, such as on bulletin boards where users post the latest review of movies.

1.1 Trust and Reputation

The extent to which an online community is useful to its members is determined by the amount of trust that exists between them. The reputation of an individual is an attribute that determines how much trust he or she inspires in another individual. However, members of an online community are very often strangers who have never met and are unrelated to one another. As such, there are no easy means by which a member can gather any information about the reputation of the party that he is dealing with online. In any bilateral transactions, there is always a temptation for the party with the upper hand to defect from the contract in ways that will result in his personal gain [1]. For example, if a buyer pays for the product first, the seller would be tempted not to send the product after receiving the payment. Therefore, in order for the transaction to transpire successfully, the seller must inspire a certain degree of trust in the buyer. The reputation of a seller, if it is reliable and easily accessible, can be instrumental in securing the trust of a buyer.

There are many possible approaches for producing trust between members of a community. The first approach is economic pooling or filtering. Such an approach works on the basis of the costs and benefits of producing and using the resource. A user does not explicitly give a rating to the resource he has just used. Instead, economic indicators are used to assess how much he values that resource. For example, instead of asking a user to rate an article he has just read, we can use the amount of time he spent reading the article as an economic indicator of how highly he regards the article. This approach can be combined with other approaches such as social filtering and social networks that will be described later.

The second approach comes from research done in sociology. A community of users can be described using a social network. A social network is a graph where the nodes represent the users and directed edges between the nodes represent the rating that one user gives to another. In these networks, centrality and prestige [2] are key concepts which describe how well connected an individual is within a social network. The various concepts of centrality are designed for graphs that are non-directional, whereas prestige can only be quantified by directional graphs. In directional graphs, measures of indegrees and outdegrees are usually different, and prestigious members are usually those with a large number of indegrees [3]. The concept of prestige as defined by sociologists can be interpreted as the reputation of a member since it is reasonable to infer that a prestigious member is one that has been given high accolades.

The third approach is to use a social filtering mechanism, as exemplified by GroupLens [4]. In this approach, resources are evaluated by users according to their tastes and preferences. In a popular form of social filtering, known as collaborative filtering, the reputation of an individual for a certain context is personalized with respect to other users. Such personalization is done using the correlation between ratings given by any pair of users on common articles in the past. In this respect, reputation as defined by GroupLens is the compatibility of interests between two individuals.

Collaborative filtering concerns the pooling together of experiences from all users for sharing. This pooling is done without weighing the reliability or reputation of the users. The weight for any given individual upon which the combined rating depends is the subject of collaborative sanctioning as described by [5].

The final approach is to employ collaborative sanctioning as a means of estimating reputation with the goal of producing trust in a community. Collaborative sanctioning is a social coordination process that lends itself readily to our common experiences. For example, we might approve or disapprove of the president's handling of the surveillance plane incident in China, or of our friend's taste in music. The level of approval may vary depending on the taste and preferences of individuals. Trust relationships are formed over time as a result of our conceived level of approval for one another. The sociological aggregate of individuals' opinions of one another can be interpreted as the basis of their reputation in the community.

Reputation serves as an inherent social coordination function by influencing the outcomes of individual behaviors. Voting in a democratic country is a manifestation of such an outcome. Collaborative sanctioning captures this coordination process by abstracting a set of quantities of interest and rules for manipulating them.

One theoretical framework in which the concept of collaborative sanctioning is applied is in the Binary Choice Bayesian Estimation mechanism as described by [5]. In this framework, users would give a resource a rating of approval or disapproval. For a user who has not rated the resource, the framework can be used to estimate the probability that he or she will approve of the resource. This estimation is done by calculating the likelihood that, on their next encounter, an individual will agree with the subject who recommended the resource.

1.2 Ratings Management Systems

Ratings management systems are built for the purpose of producing trust between members of an online community. Such systems are an implementation of collaborative sanctioning. A ratings management system calculates the reputation of any member based on the ratings that were given to him by all the users that have interacted with him in the past. For example, using this reputation measure, a buyer can determine whether the seller is a party that could be trusted to carry out a transaction successfully. If that were not the case, the buyer could use the system to look for a seller that is more trust-worthy.

A ratings management system characterizes an online community with a social network. The members of the community are represented with nodes in the social network, and the edges between the nodes represent the existence of ratings between users. The weight associated with each edge is the current rating that is given by one user to another based on their past and current interactions. The rating management system updates these weights as the ratings change over time due to an increase in the number of bilateral transactions. The system also adds nodes to the network as the size of the community increases. The reputation of a member is calculated depending on the structure of the network as well as the ratings between the members.

1.3 Background

Much research has been done in the area of information retrieval and search engines. Some noted works in these areas include those done by [6], [7] and [8]. In information retrieval, the main objective is to select relevant resources for given query specifications. Search engines are manifestations of that paradigm in the web domain. Collaborative filtering is a mechanism for using the collective experience of many users in a community to select relevant resources in response to a query [9]. A limitation of this mechanism is that large numbers of users are needed before any reasonable performance can be achieved. Another limitation is that no metric exists to determine the reliability of the system in selecting relevant resources for its users.

Ratings management systems are built to provide reputation or reliability assessment of information sources, providing an independent measurement dimension for these sources. In a community of users, such a system is the intermediary that allows ratings of resources to propagate from one user to the next. The characteristics of the system used determine the quality of the ratings. There are a number of ratings management systems that have been introduced. However, these systems are often designed in an ad-hoc fashion. eBay provides a +1/-1 feedback system from buyers about sellers and vice-versa [10]. A few recent high profile fraud cases involving users with high eBay ratings suggest that the ratings management system currently used by eBay is in urgent need of improvement.

Sporas and Histos [11] allow users in a community to rate one another after each transaction and modify their reputations based on these ratings. Collaborative filtering, of which GroupLens [12] is a good example, allows reputation to be personalized based on similarity between users. Other systems, such as Yenta [13], Weaving a Web of Trust [14], or the Platform for Internet Content Selection (PICS) [15] require that users give a rating for themselves and either have authoritative agencies or other users verify their reputation. All of these reputation systems deal with reputation in a single context. In Sporas and Histos [11], there is mention of reputation being a multidimensional value, where an individual may enjoy a very high reputation in one domain, whilst having a low reputation in another.

Ratings management systems available in commercial websites such as eBay, Slashdot, and Amazon usually assign a single reputation to each user and therefore ignore the personalized nature of reputation. An individual's reputation, as opposed to being a fixed attribute, actually varies depending on the tastes, preferences, opinions, and biases of other people interested in knowing his reputation. We term the assignment of reputation that is not personalized as "global" reputation. [1] has provided warnings against possible attack methodologies that can be used against ratings management systems that employ a "global" notion of reputation, and has discussed simple solutions.

Given these two opposing notions of reputation, we would like to find out if "global" reputation or "personalized" reputation is a better yardstick in determining the reliability of the sources (i.e. users) in retrieving relevant information. How is "global" reputation measured? Sociologists have long studied reputation from a network perspective [3]. One such measure of reputation is known as the prestige measure. It is derived by considering the adjacency matrix associated with a network of users. The entries in the adjacency matrix are the values of association (in our case, ratings) from one user to another. If the matrix were normalized to have column sum to 1 (so as to be stochastic), then the right eigenvector associated with an eigenvalue of 1 is the stationary solution of the stochastic matrix. The prestige measure of each user in the network is represented by an entry in the eigenvector. The largest entry corresponds to the most reputable user.

In recent engineering literature, there have been many attempts to formulate a coherent set of rules for designing systems that manage reputation. Among these, [16] have proposed a "brownie-points" system to represent how conscientious an agent is in a community. [17] and [18] have each suggested a set of arbitrary formulae for calculating reputation among agents. [11] has suggested several criteria that are useful guidelines for designing ratings management systems. However, its formulation for reputation is equally arbitrary. All these schemes are based on the "global" notion of reputation where the reputation of an individual is uniform for all users of the system. The justification for using "global" reputation as opposed to "personalized" reputation has been lacking in most literature.

2. Theory

2.1 Ratings Management Systems

In an online community, members interact with one another not just to exchange services, but also to share their opinions and ideas. An important benefit that members gain from interacting with other members in the community is the ability to obtain referrals to and recommendations of services and resources which were previously unknown to them. For example, in a restaurant sanctioning system, a user can learn about various parameters of new restaurants such as their cuisine, ambience and other qualities by looking at the ratings that other users have given these new restaurants.

However, there are potentially many different types of resources and services that have been assessed by the entire online community, and any member is really only interested in a small subset of them. It would be time-consuming and expensive for a member to search through all the recommendations in order to discover the resources that are useful to him. Moreover, the rating that any member gives to a resource is subject to his own personal judgment, biases and tastes. Since members of an online community are often unrelated to one another, they may have never met and thus any member would have no prior knowledge of another's taste and preferences. In order to assess the value of another member's recommendation of a resource, it is important to know the degree to which his taste and biases reflect our own.

The members whose opinions and recommendations one would trust most are known as reputable members. In order to determine who are the reputable members in a community, we can either employ a social ratings management system or a personalized ratings management system. Both of these ratings management systems require that a social network consisting of nodes representing users and weighted edges representing user-to-user ratings be set up. The social network could be formed by asking each user to rate a few other users based on the degree to which he agrees with their recommendations. For instance, in the restaurant sanctioning system example, a user could visit the restaurants that another user has recommended and give him a rating after assessing the restaurants depending on the degree to which he agrees with the other user's recommendation.

A social ratings management system would select the members of the community that are most prestigious as determined by the number and quality of ratings that each member has received from the other members in the community. Such a rating system makes use of aggregate approval in order to select for any member the best members within the community that he could seek recommendations from. Therefore, the prestigious members as selected by the system reflect the opinions of the majority of the members in the community. If a member has opinions that differ greatly from the majority of the other users, the prominent members as selected by the system might produce recommendations that are not to his liking, since his taste is not common. He might be far better off using the recommendations of another member who is not so prominent, but who has opinions that he is more likely to agree with.

Since the social ratings management system determines the reputation of members based on measures of network centrality, it is non-personalized. In other words, it would produce the same selection of reputable members for any member in the social network, regardless of differences between these members. This sets it apart from personalized ratings management systems.

A personalized ratings management system, on the other hand, takes into account the taste and opinions of each member when selecting the other members that he is most likely to agree with. Therefore, the problem of selecting reputable members for a member with rare tastes can be tackled by deploying a personalized system. Since each member in the community has only rated a small number of other members, the personalized ratings system must predict how he would rate the other members that he has not yet rated in the social network. Therefore, different types of personalized rating systems must have different ways of estimating this rating, using techniques to propagate pair-wise ratings between members. For any particular member, the personalized ratings system can rank the other members based on the estimated ratings that this particular member would give to each of them, calculated using the ratings propagation mechanism that is specific to the rating system.

In forming the social network, each member can give another member a rating based on how much he agrees with the restaurants that the latter recommends regardless of the type of the restaurants. Using the social network formed as such, it would be possible to deploy either type of ratings management systems to select reputable members. However, the resulting selection might not be very reliable because the abilities of members in recommending different types of restaurants differ. A member who excels in recommending French restaurants might not be as good in recommending Middle Eastern restaurants. The usage of context-based ratings could therefore result in a more accurate selection of reputable individuals.

In order to make use of context-based ratings, the recommended restaurants could be categorized according to their cuisine, such as French, Italian or Indian. For each context, a separate social network could be formed by users rating other users recommending restaurants of that particular context. Therefore, a user who is interested in looking for a restaurant recommendation would first have to look for the social network of the correct context before employing the ratings management systems to discover the most reputable individuals for him.

In this thesis, we propose that the personalized ratings management system using context-based ratings would produce a better selection of reputable individuals for a member of a community than the social ratings management system. In order to test this hypothesis, we have developed two different personalized ratings management systems to compare against a social ratings management system. The two systems that we have developed are the Bayesian estimate rating system and the Preference-based rating system. The social ratings management system to be used for comparison is a Centrality-based rating system that uses a prestige measure of network centrality known as the eigenvector centrality.

2.2 Centrality-based rating system

We can represent an online community using a social network. Every node represents a member and every directed edge between two nodes represents a rating given by one member to another based on their previous interactions. Figure 2-1 shows a representation of a social network involving 3 members, with ratings between them represented by directed edges.

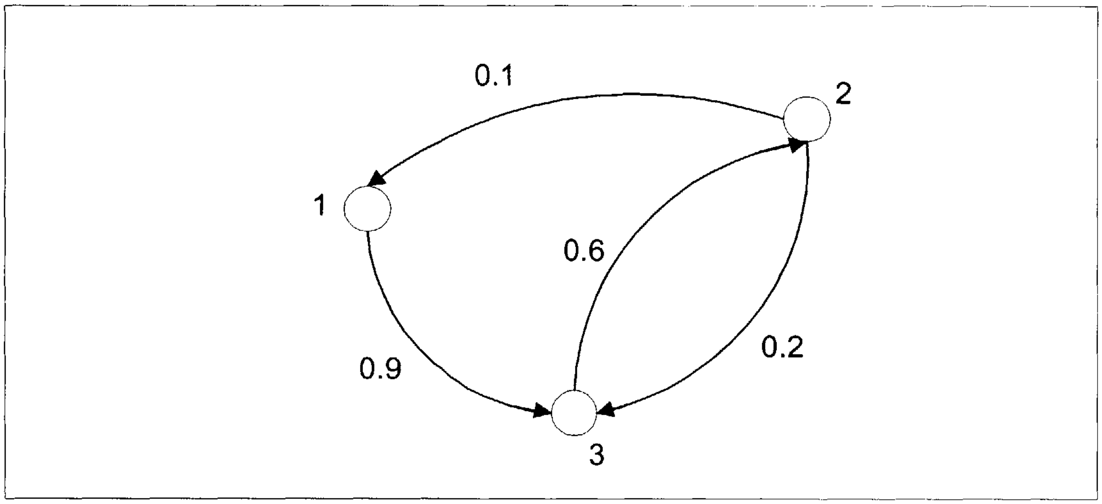

Figure 2-1: A sample social network of 3 users

The relationships between members of a social network can be represented by an adjacency matrix, A. The entry a_{ij} of matrix A represents the rating that member i has given member j. The matrix A below is the adjacency matrix for the social network shown in Figure 2-1. Note that the diagonals of the adjacency matrix are always 1 because we assume that every member would agree completely with his own judgment.

$$A = \begin{bmatrix} 1.0 & 0.0 & 0.9 \\ 0.1 & 1.0 & 0.2 \\ 0.0 & 0.6 & 1.0 \end{bmatrix} \qquad (1)$$

2.2.1 Prestige measures of centrality

In social networks, centrality is a key concept which describes how well connected an actor is within a network. In directional graphs representing social networks, measures of indegrees and outdegrees are usually different, and well-connected actors are usually those with a large number of indegrees.

As a further extension, the prestige measures of centrality are measures in which the centralities or statuses of nodes in the social network are recursively related to the centralities or statuses of the nodes to which they are connected [3]. Such a measure implies that being rated highly by a prestigious member would increase one's prestige in the social network. If highly prestigious members dominate a member's sphere of influence, his prestige should also be high. However, if only less important members are acquainted with him, then his prestige would accordingly be lower despite similar ratings given to him by both groups.

A member's prestige in the network depends not only on the quantity and quality of the ratings that he receives; it also depends on the prestige of the members giving him these ratings. Let us define x_i to be the prestige measure for member i within a social network of n members. The prestige measure, x_i, is a function of the prestige values of the members who have rated the member i. The ith column of the adjacency matrix, A, contains the ratings that the other members of the social network gave to member i. We have to multiply each rating by the prestige measure of the member who gave that particular rating before summing them up to obtain a formula for the prestige measure of member i:

$$x_i = a_{1i}x_1 + a_{2i}x_2 + \ldots + a_{ni}x_n \tag{2}$$

Therefore, it can be inferred from equation (2) that the prestige of member i, x_i, is dependent on the ratings that other members gave to it, a_{ki}, as well as the prestige of these other members, x_k. For a social network of n members, we have to vary index i in equation (2) from 1 to n to obtain n different equations, each of which depends on all the prestige indices x_i themselves. Thus, we have a system of n equations with n unknowns. To find the solutions for them, we can put the set of prestige indices into a vector $\vec{x}$ where:

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{3}$$

If we take the adjacency matrix A, and the vector of prestige indices $\vec{x}$, we can represent the system of n equations by this matrix equation:

$$\vec{x} = A'\vec{x} \tag{4}$$

If we were to rearrange the terms in equation (4), we would obtain the equation $(I - \lambda A')\vec{x} = \vec{0}$, where I is the identity matrix of dimension n, $\vec{x}$ and $\vec{0}$ are vectors of length n, and λ is equal to 1.

This equation looks identical to the characteristic equation used to determine the eigensystem of matrix A', where $\vec{x}$ is the eigenvector of matrix A' corresponding to an eigenvalue of 1. However, it is not a true characteristic equation because if we do not place any restrictions on A', or the vector of prestige indices $\vec{x}$, equation (4) has no finite solution in general [3]. The reason is that A' does not have an eigenvalue of 1 in general.

There are many possible solutions to this problem, and each of them can be categorized according to the various constraints that they put on the adjacency matrix, the vector of prestige indices or on system of equations itself. Two of these solutions would be discussed in this section and one of them would be implemented in the Centrality-based rating system.

2.2.2 Standardizing the adjacency matrix

The first solution would be to standardize the adjacency matrix to have column sums of unity [21]. This standardization would force matrix A' to have an eigenvalue of unity, and the eigenvector associated with this eigenvalue would be x', the vector of prestige indices. Such a way of standardizing matrix A does not affect the relative prestige of each member in the social network. At the same time, it allows us to find the solution to equation (4) by converting the system of equations into a familiar matrix characteristic equation that can be solved by familiar techniques.

After standardization, the largest eigenvalue of the matrix A' is guaranteed to be 1. Let the eigenvector corresponding to this eigenvalue be $\vec{p}$. Then, the elements of vector $\vec{p}$ are the prestige indices of the members in the social network represented by adjacency matrix A. The prestige of a member is the sum of the prestige values of members who have rated him, weighted by their ratings. Therefore, a member's prestige would be higher if either the ratings were higher or if the members who gave those ratings have higher prestige.

If the first solution were to be implemented in the Centrality-based rating system, the online community would first be characterized by a social network, with members represented by nodes and ratings represented by weighted, directed edges between nodes. The social network would then be converted into an adjacency matrix that contains all the member ratings in the network. The standardization of the adjacency matrix would be carried out next, and the eigenvectors of the transpose of the adjacency matrix would be calculated. The eigenvector that is associated with the eigenvalue of unity would be used to assign the prestige values of the members in the online community.

As the eigenvector contains the prestige values of each member in the community, the Centrality-based rating system can rank all the members in the community according to his prestige. The system would then select the members with the highest prestige to recommend to any other member seeking recommendations for the context in which the social network is formed. As explained in the previous section, the Centrality-based rating system is a social ratings management system that makes use of aggregate approval to select the most reputable members of a community. In selecting the members with the highest prestige from the eigenvector with eigenvalue of unity, we can see that the system indeed chooses the members whose opinions reflect that of the majority of the other members in the community. Also, the same group of members of high prestige is recommended to any member regardless of the latter's personal opinions, biases, tastes or preferences.

2.2.3 Using the general eigenvector equation

The second solution that we can use to ensure that equation (4) has a solution is to generalize equation (4) so that it becomes the general eigenvector equation, as first suggested by Bonacich [21]. The interpretation of this generalization is that each member's prestige measure is only proportional, and not exactly equal to the weighted sum of the prestige values of the members who have rated it. Equation (5) illustrates this fact about x_i, where λ is the constant of proportionality.

$$\lambda x_i = a_{1i}x_1 + a_{2i}x_2 + \ldots + a_{ni}x_n \qquad (5)$$

For a social network of n members, we have to vary index i in equation (5) from 1 to n to obtain n different equations, each of which depends on all the prestige indices x_i themselves. Thus, we have a system of n equations with n unknowns. To find the solutions for them, we can put the set of prestige indices into a vector $\vec{x}$ as shown in equation (3). This set of n equations can be represented by the matrix equation:

$$A'\vec{x} = \lambda\vec{x} \qquad (6)$$

If A is an n by n matrix that represents the ratings in a social network, equation (6) has n solutions, each corresponding to a particular value of λ. These n solutions are the n solutions to the eigensystem of A'. If we solve the characteristic equation of matrix A', as given by equation (7), we would obtain n eigenvalues, with n corresponding eigenvectors.

$$(A' - \lambda I)\vec{x} = \vec{0} \tag{7}$$

Therefore, the general solution to equation (6) can be expressed as a matrix equation, in which X and Λ are both n by n matrices. Λ is a diagonal matrix of eigenvalues, and each column in X is an eigenvector of A, corresponding to the eigenvalue in the same column of Λ. The matrix equation that represents the general solution is given by equation (8).

$$A'X = X\Lambda \tag{8}$$

In equation (8), there are n eigenvectors of A', found in the columns of matrix X. Each of these eigenvectors is associated with an eigenvalue. Since each eigenvalue is a constant of proportionality in equation (6), there are n possible sets of values for the prestige indices of the members in the social network. The elements in each eigenvector represent one set of values for the prestige indices of the members. The prestige ranking of the members of the social network would differ depending on the eigenvector that is being used. Therefore, a method must be found to choose the eigenvector that would give the correct prestige ranking.

For the prestige ranking of the individuals to be equal to that given by the first solution, the eigenvector associated with the correct eigenvalue must be chosen. The procedure for choosing the correct eigenvalue is as follows. The absolute values of all the eigenvalues, as given by the diagonal elements of Λ, are calculated. The eigenvalue with the largest absolute value is selected and its eigenvector is chosen to produce the set of prestige indices of the members. Using the elements of this eigenvector to assign the prestige indices of the members, the prestige ranking of these members would be the same as that obtained in the first solution. Therefore, in calculating the prestige values of the members of a social network, both the first and second solutions produce similar results, up to an ordinal degree.

If the second solution were to be implemented in the Centrality-based rating system, the online community would first be characterized by a social network, with members represented by nodes and ratings represented by weighted, directed edges between nodes. The social network would then be converted into an adjacency matrix that contains all the member ratings in the network. The characteristic equation of the transpose of the adjacency matrix would be solved to obtain the n eigenvalues. The eigenvectors associated with each of the n eigenvalues would be calculated. The eigenvector associated with eigenvalue that has the largest absolute value would be selected. Its elements would be the prestige measure for each of the members of the online community.

Both the first and second solutions produce the same prestige ranking; however, the first solution involves less computation. Therefore, we choose to implement the first solution in the Centrality-based rating system in calculating the prestige values of members in a social network. Choosing members based on this set of prestige values, the system selects the most prestigious members that it recommends to other members seeking recommendations for the context in which the social network is formed.

2.3 Preference-based rating system

Similarly, for a Preference-based rating system, we need to represent an online community using a social network. Every node represents a member and every directed edge between two nodes represents a rating given by one member to another based on their previous interactions with respect to a given context. The Preference-based rating system is a personalized ratings management system. It takes into account the preferences of each member when selecting the reputable members in the community that he is most likely to agree with. In this framework, the preference, $\rho_i(c_p)$, of an individual, i, for some given context, c_p, is defined as the probability that he would approve of a resource that can be categorized within that context.

The subject is defined as the member for whom the rating system is currently selecting the reputable members. These reputable members give opinions that the subject is most likely to agree with. The Preference-based rating system personalizes the selection of the reputable members of a community for any subject based on his preferences. It does so by estimating the rating that the subject would have given to every other member of the community if he had interacted with them directly.

2.3.1 Definition of ratings propagation mechanism

For the members that the subject had rated personally, his opinion of them could be determined directly without any kind of estimation. However, for other members in the community that the subject member had not interacted with and rated before, his likely rating of them would have to be estimated using a ratings propagation mechanism.
The ratings propagation mechanism is defined as such:

Let $\rho_{ij}(c_p)$ be the rating that member i gives to member j with respect to a context, c_p.
Given $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$,

$$\rho_{ik}(c_p) = f_2(\rho_{ij}(c_p), \rho_{jk}(c_p)) \tag{9}$$

where $f_2(\cdot)$ is the ratings propagation function for 2 links.

2.3.2 Incorporation of personal preferences and tastes

With the estimated rating that would be given by the subject to every other member in the social network, the Preference-based rating system can select the most reputable members in the social network as would be regarded by the subject. These reputable members are the members that are predicted to receive the highest ratings from the subject if he were to rate them directly.

A social network is formed based on pair-wise ratings given with respect to a certain context. Every member in the social network has a personal preference for that context.

His preference is reflected in the way that he rates the other members that he comes into contact with. Therefore, a subject's personal preference is inherent in the direct ratings that he gives. These direct ratings are used to estimate the ratings that he would give to all the members in the social network that he had not rated directly. In this way, his personal preference is incorporated into the selection of reputable members of the social network for him. Because it factors the personal preference of the subject member into the final selection, the Preference-based rating system is a personalized system.

2.3.3 Binary preference scenario

The evolution of a suitable framework for the ratings propagation mechanism is the crucial component in the development of the Preference-based rating system. Let us start with the simplest possible case. Assume that the personal preference of a member is extreme; his opinion is either at one end of the spectrum or at the other end, but not anywhere in between. For example, let's take the case of Elvis. One member might insist that Elvis is alive, whilst another is dead certain that Elvis is dead. In this simplest case, every member can only take one of either opinions, but nothing in between. That is, no one can be unsure about whether Elvis is dead or alive. This simplest case is known as the binary preference scenario.

Let member i be the subject that the Preference-based rating system is working for. Let the personal preference of member i with respect to context c_p be represented by $\rho_i(c_p)$. In the binary preference scenario,

$$\rho_i(c_p) = \begin{cases} 0 \\ 1 \end{cases} \tag{10}$$

2.3.3.1 Pair-wise ratings

Let member j be another member in the social network of n members, $j \in \{1, n, j \neq i\}$.

Given the binary representation of $\rho_i(c_p)$, the rating that subject, member i, would give to member j with respect to context c_p, $\rho_{ij}(c_p)$ is given by:

$$\rho_{ij}(c_p) = \begin{cases} 1, \text{ if } \rho_i(c_p) = \rho_j(c_p) \\ 0, \text{ otherwise} \end{cases} \tag{11}$$

2.3.3.2 Ratings propagation function

As we had expected, the pair-wise rating between members will also be binary. Let member k be another member in the social network of n members, $k \in \{1, n, k \neq i, j\}$. The subject, member i, has not rated member k directly. To determine the ratings propagation function for 2 links, we shall examine all the possible combination of values for $\rho_i(c_p)$, $\rho_j(c_p)$ and $\rho_k(c_p)$ in Figure 2-2.

$\rho_i(c_p)$	$\rho_j(c_p)$	$\rho_k(c_p)$
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Figure 2-2: Possible combinations of personal preferences of members i, j and k

From the table in Figure 2-2, we can infer that $\rho_{ik}(c_p)$ is equal to 1 only when $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ are both equal to 1, in rows 1 and 8; or when $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ are both equal to 0, in rows 3 and 6.

Therefore, in the binary preference scenario, the ratings propagation function for 2 links is given by:

$$\begin{aligned}\rho_{ik}(c_p) &= f_2(\rho_{ij}(c_p), \rho_{jk}(c_p)) \\ &= \rho_{ij}(c_p)\rho_{jk}(c_p) + (1-\rho_{ij}(c_p))(1-\rho_{jk}(c_p))\end{aligned} \tag{12}$$

For determining the estimated ratings of members that are more than 2 links away from the subject, member i, in the social network, we have to employ equation (12) iteratively. For example, in order to obtain $\rho_{il}(c_p)$, we can first calculate $\rho_{ik}(c_p)$ using equation (12). Then, assuming that $\rho_{kl}(c_p)$ can be obtained from the social network, we can reapply equation (12) again using $\rho_{ik}(c_p)$ and $\rho_{kl}(c_p)$ as the parameters for $f_2(\cdot)$:

$$\rho_{il}(c_p) = f_2(\rho_{ik}(c_p), \rho_{kl}(c_p)) \tag{13}$$

Therefore, in the binary preference scenario, it is possible for the Preference-based rating system to use the above rating propagation mechanism to calculate the estimated rating that would be given by subject, member i, for every other member in the network. It is important to note that the personal preference of each member, $\rho_i(c_p)$, $i \in \{1,n\}$ cannot be obtained from the social network. It can only be inferred by looking at the pair-wise rating, $\rho_{ij}(c_p)$, $i,j \in \{1,n\}$, between individuals in the network. However, to calculate the estimated ratings that would be given by the subject to every other user, we only need the values of $\rho_{ij}(c_p)$ that are found in the social network, no values of $\rho_i(c_p)$ are needed.

2.3.4 Continuous preference scenario

In the next section, let us examine the case where the personal preference of a member could be anywhere on the spectrum. For instance, consider the case of Thai cuisine. One member might find Thai cuisine delectable; another might not like Thai cuisine that much. If we assign a value of zero to extreme dislike and a value of one to extreme approval, the preference of each member for Thai cuisine can range from zero to one. A member with a preference of 0.5 is ambivalent about Thai cuisine. This more complex case is known as the continuous preference scenario.

Given that each member in a community has his own preference value with respect to the context of 'Thai cuisine', let us consider how the social network can be formed in that context. In order to get a set of pair-wise ratings between the members of the community, we can get every member to visit a set of Thai restaurants. The members who enjoy Thai cuisine would give these restaurants high grades whereas those who dislike Thai cuisine would give them low grades.

Once that is done, each member would go about rating a few other members based on what they think about the food at the Thai restaurants that they have visited. Naturally, the members who hate Thai cuisine would give other such members high ratings since they all dislike the food at the restaurants. They would give those members who like the food at the Thai restaurants low ratings since their opinions differ from the latter members. Using these ratings, a social network can be set up with respect to the context of 'Thai cuisine'. If a subject is one who likes Thai food, he would want to seek another member who shares his preference in order to seek the latter's opinion about good restaurants. It is the Preference-based rating system's task to seek that latter member out for the subject.

Let member i be the subject that the Preference-based rating system is working for. Let the personal preference of member i with respect to context c_p be represented by $\rho_i(c_p)$. In the continuous preference scenario,

$$\rho_i(c_p) \in [0, 1] \tag{14}$$

2.3.4.1 Pair-wise ratings

Let member j be another member in the social network of n members, $j \in \{1, n, j \neq i\}$.

Given the continuous representation of $\rho_i(c_p)$, the rating that the subject, member i, would give to member j with respect to context c_p, $\rho_{ij}(c_p)$, is the probability that the subject would agree with member j for that context.

Let us again consider the example of Thai cuisine and Thai restaurants described above. If member i has a preference value of $\rho_i(Thai)$ for Thai cuisine, the probability that he would approve of any Thai restaurant is given by $\rho_i(Thai)$. The probability that member j would approve of any Thai restaurant is similarly given by $\rho_j(Thai)$. Therefore, the probability that the subject, member i, would agree with member j is given by the sum of the probability that both approves of any Thai restaurant, $\rho_i(Thai)\rho_j(Thai)$, and the probability that both disapproves of the restaurant, $(1-\rho_i(Thai))(1-\rho_j(Thai))$. Thus, the rating that member i would give to member j, $\rho_{ij}(c_p)$, for context c_p is given by:

$$\rho_{ij}(c_p) = \rho_i(c_p)\rho_j(c_p) + (1-\rho_i(c_p))(1-\rho_j(c_p)) \tag{15}$$

As expected, the pair-wise rating between members will also range from zero to one.

In the formation of the social network, member i with preference value $\rho_i(c_p)$, and member j with preference value $\rho_j(c_p)$ would interact with each other so that member i can give a rating to member j. However, the rating, $\hat{\rho}_{ij}(c_p)$, that member i gives to member j will not be exactly equal to $\rho_{ij}(c_p)$ because member j does not actually reveal his preference value, $\rho_j(c_p)$ to member i. Instead, member i gives member j a rating based on how much he agrees with member j's approval of a set of resources with respect to context c_p. Therefore, the pair-wise rating in the social network, $\hat{\rho}_{ij}(c_p)$, $i, j \in \{1, n\}$ is only an estimate of the true rating derived from preference values, $\rho_{ij}(c_p)$, $i, j \in \{1, n\}$.

2.3.4.2 Ratings propagation function

Let member k be another member in the social network of n members, $k \in \{1, n, k \neq i, j\}$. If we know the value of $\rho_k(c_p)$, we can just use equation (15) to determine $\rho_{ik}(c_p)$. It would not be necessary to develop a ratings propagation mechanism to estimate $\rho_{ik}(c_p)$ from $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$.

However, the social network does not provide values of $\rho_i(c_p)\ \forall\ i \in \{1,\ \mathrm{n}\}$, it only provides estimates of the pair-wise ratings, $\hat{\rho}_{ij}(c_p)$, $i, j \in \{1, n\}$ between select pairs of members, (i, j). Therefore, we would not be able to calculate the exact value of $\rho_{ik}(c_p)$ exactly using $\rho_i(c_p)$ and $\rho_k(c_p)$. Instead, we have to develop a ratings propagation mechanism to estimate $\rho_{ik}(c_p)$ from $\hat{\rho}_{ij}(c_p)$ and $\hat{\rho}_{jk}(c_p)$, where both $\hat{\rho}_{ij}(c_p)$ and $\hat{\rho}_{jk}(c_p)$ are obtained from the social network.

Next, we shall proceed to derive the ratings propagation function for members separated by 2 links, $\rho_{ik}(c_p) = f_2(\rho_{ij}(c_p), \rho_{jk}(c_p))$ from equation (15).

From equation (15), we can substitute indices to obtain the expressions for $\rho_{jk}(c_p)$ and $\rho_{ik}(c_p)$ as a function of $\rho_i(c_p)$, $\rho_j(c_p)$ and $\rho_k(c_p)$.

$$\rho_{jk}(c_p) = \rho_j(c_p)\rho_k(c_p) + (1 - \rho_j(c_p))(1 - \rho_k(c_p)) \tag{16}$$

$$\rho_{ik}(c_p) = \rho_i(c_p)\rho_k(c_p) + (1 - \rho_i(c_p))(1 - \rho_k(c_p)) \tag{17}$$

By manipulating equations (15), (16) and (17), we can obtain an expression for $\rho_{ik}(c_p)$ as a function of $\rho_i(c_p)$, $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$. The expression is given by equation (18), with the context variable, c_p, omitted for clarity:

$$\rho_{ik} = \begin{cases} \dfrac{(2\rho_i - 1)(2\rho_i\rho_{jk} - \rho_i - \rho_{jk} + \rho_{ij}) + (1-\rho_i)(2\rho_{ij} - 1)}{2\rho_{ij} - 1}, & if\ \rho_i \neq 0.5 \\ 0.5 & , if\ \rho_i = 0.5 \end{cases} \tag{18}$$

The steps of derivation of equation (18) can be found in Appendix A.

If $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ are true ratings derived from $\rho_i(c_p)$, $\rho_j(c_p)$ and $\rho_k(c_p)$ using equation (15), then $\rho_{ik}(c_p)$ as calculated by equation (18) would have a value between zero and one. When $\rho_i(c_p)$ has a value of 0.5, $\rho_{ij}(c_p)$ will have a value of 0.5 regardless of $\rho_j(c_p)$, according to equation (15). Therefore, we can infer $\rho_{ik}(c_p)$ to be 0.5 directly from equation (17).

However, we do not have the true ratings, $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ that are derived from the preference values, $\rho_j(c_p)$ and $\rho_k(c_p)$, of members j and k, since these are also unknown. Nevertheless, the social network does provide estimates of member i's true rating of member j, $\hat{\rho}_{ij}(c_p)$, and member j's true rating of member k, $\hat{\rho}_{jk}(c_p)$.
Substituting these estimated ratings for $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ in equation (18), we derive an estimated rating that member i would give to member k, as given by equation (19).

$$\begin{aligned} \hat{\rho}_{ik} &= f_2(\hat{\rho}_{ij}, \hat{\rho}_{jk}) \\ &= \begin{cases} \dfrac{(2\rho_i - 1)(2\rho_i\hat{\rho}_{jk} - \rho_i - \hat{\rho}_{jk} + \hat{\rho}_{ij}) + (1-\rho_i)(2\hat{\rho}_{ij} - 1)}{2\hat{\rho}_{ij} - 1}, & if\ \rho_i \neq 0.5 \\ 0.5 & , if\ \rho_i = 0.5 \end{cases} \end{aligned} \tag{19}$$

The preference value of the subject, member i, since it is known to him, should be easily available to the Preference-based rating system. However, since we do not know the distribution of $\rho_i(c_p)$ across all members, it might be best to determine $\rho_i(c_p)$ using the proportion of approvals that member i gave to resources in the context c_p.

In replacing $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$ in equation (18) with their estimated values, $\hat{\rho}_{ij}(c_p)$ and $\hat{\rho}_{jk}(c_p)$ respectively, the range of $\hat{\rho}_{ik}(c_p)$ now exceeds $\{0, 1\}$. Therefore, in order for $\hat{\rho}_{ik}(c_p)$ to remain in a legitimate range, if it has a value greater than one, it should be rounded down to one. Similarly, if it has a value that is less than 0, it should be rounded up to zero.

For determining the estimated ratings of members that are more than 2 links away from the subject, member i, in the social network, we have to employ equation (19) iteratively, just like in the binary preference scenario. For example, in order to obtain $\hat{\rho}_{il}(c_p)$, we can first calculate $\hat{\rho}_{ik}(c_p)$ using equation (19). Then, assuming that $\hat{\rho}_{kl}(c_p)$ can be obtained from the social network, we can reapply equation (19) again using $\hat{\rho}_{ik}(c_p)$ and $\hat{\rho}_{kl}(c_p)$ as the parameters for $f_2(\cdot)$:

$$\hat{\rho}_{il}(c_p) = f_2(\hat{\rho}_{ik}(c_p), \hat{\rho}_{kl}(c_p)) \tag{20}$$

Therefore, in the continuous preference scenario, it is possible for the Preference-based rating system to use the rating propagation function in equation (19) to calculate the estimated rating that would be given by subject, member i, for every other member in the network.

Since the continuous preference scenario is much less restrictive than the binary preference scenario, the former is chosen to be implemented in the Preference-based rating system. Since the rating the subject would give to every other member in the social network can be estimated by the ratings propagation function in equation (19), the Preference-based rating system can select the most reputable members in the social network as would be regarded by the subject.

2.4 Bayesian estimate rating system

The Bayesian estimate rating system is a personalized ratings management system that is able to personalize the selection of reputable individuals for different subjects. We represent an online community using a social network where nodes represent members and directed edges represent pair-wise ratings. Each social network is formed with respect to a specific context. Using a ratings propagation mechanism, the Bayesian estimate rating system is able to estimate the rating that the subject would have given to every other member of the community if he had interacted with them directly. The ratings propagation mechanism for 2 links is defined by a function as shown in equation (9). The exact formulation of this function is the topic of the following discussion.

Assume that there are multiple encounters between the subject, member i, and another member in the community, member j, with respect to a certain context, c_p. During each encounter, the subject either approves or disapproves of member j. The final rating that the subject gives member j is a measure that depends on the number of past approvals and disapprovals. In the context of 'Thai cuisine' and restaurants described in the previous section, each encounter between the subject and member j can be spent discussing one restaurant that both have visited in common. The subject would give member j an approval if they both agree that the restaurant has good food, or if both think that the restaurant has inferior food. After the number of encounters between the two equals the number of Thai restaurants that both have visited in common, the subject would be ready to give a final rating for member j that would be reflected in the social network.

2.4.1 Pair-wise ratings

Let $x_{ij}(n+1)$ be the binary random variable that represents member i's approval of member j for the (n + 1)th encounter between them. Based on the set of previous encounters between the two of them, we can estimate $x_{ij}(n+1)$. The details of the derivation that follows are given by [22].

Let n be the total number of encounters between member i and member j in the past.

Let p be the number of approvals that member i gave member j in those encounters.

Let θ be the true proportion of approvals that member i would give to member j.

Let $\hat{\theta}$ be an estimator for θ based on all the past encounters between member i and member j.

If member i and member j had n encounters in the past, the proportion of approvals that member i gave member j can be modeled using a Beta prior distribution as given by [22]:

$$\begin{aligned} p(\hat{\theta}) &= Beta(c_1, c_2) \\ &= \frac{\hat{\theta}^{c_1-1}(1-\hat{\theta})^{c_2-1}}{B(c_1, c_2)} \end{aligned} \tag{21}$$

$$\text{where } B(c_1, c_2) = \frac{\Gamma(c_1)\Gamma(c_2)}{\Gamma(c_1) + \Gamma(c_2)}$$

and c_1 and c_2 are parameters that are determined by prior assumptions.

Let D be the set of past encounters between member i and member j.

$$D = \{x_{ij}(1), x_{ij}(2), \ldots, x_{ij}(n)\} \tag{22}$$

Assume that for each encounter between member i and member j, the probability for approval is independent of other encounters between them. Let P be the number of approvals in n encounters.

Then, according to [22], P follows a binomial distribution:

$$P \sim Binomial(n, \hat{\theta})$$

$$\Pr_P(p) = \binom{n}{p} \hat{\theta}^p (1-\hat{\theta})^{n-p} \tag{23}$$

Let D_p be the collection of all possible sets of n encounters that contains p approvals and n - p disapprovals. Given that the estimator of the true proportion of approvals, $\hat{\theta}$, is known, $L(D_p \mid \hat{\theta})$, the likelihood for D_p, can be modeled as:

$$L(D_p \mid \hat{\theta}) = \binom{n}{p} \hat{\theta}^p (1-\hat{\theta})^{n-p}$$

$$\Rightarrow \quad L(D_p \mid \hat{\theta}) \quad \propto \quad \hat{\theta}^p (1-\hat{\theta})^{n-p} \tag{24}$$

Combining the prior and the likelihood, the posterior estimate of $\hat{\theta}$ becomes:

$$\begin{aligned} p(\hat{\theta} \mid D_p) &= \frac{L(D_p \mid \hat{\theta}) p(\hat{\theta})}{L(D_p)} \\ &= \frac{L(D_p \mid \hat{\theta}) p(\hat{\theta})}{\int_{\hat{\theta}} L(D_p \mid \hat{\theta}) p(\hat{\theta}) d\hat{\theta}} \\ &= Beta(c_1 + p, c_2 + n - p) \end{aligned} \tag{25}$$

The steps of derivation of equation (25) can be found in Appendix B.

In the Bayesian framework for reputation in [22], the rating that member i would give to member j is member i's estimate of the probability that he will approve of member j in the next encounter. This estimate is based on the n previous encounters between them. The estimate can be calculated as follows:

$$p(x_{ij}(n+1)=1 \mid D)=\int_{\hat{\theta}} p(x_{ij}(n+1)=1 \mid \hat{\theta}_n, D)\, p(\hat{\theta}_n \mid D)\, d\hat{\theta}_n \tag{26}$$

where $\hat{\theta}_n$ is the estimated proportion of approvals based on n previous encounters.

The probability of $x_{ij}(n+1)=1$ given the estimator, $\hat{\theta}_n$, from n previous encounters, is given by $p(x_{ij}(n+1)=1 \mid \hat{\theta}_n, D)$.

Replacing $p(x_{ij}(n+1)=1 \mid \hat{\theta}_n, D)$ in equation (26) with the normalized likelihood:

$$\begin{aligned} p(x_{ij}(n+1)=1 \mid D) &= \int \frac{L(x_{ij}(n+1)=1 \mid \hat{\theta}_n)}{L(x_{ij}(n+1)=1 \mid \hat{\theta}_n)+L(x_{ij}(n+1)=0 \mid \hat{\theta}_n)}\, p(\hat{\theta}_n \mid D)\, d\hat{\theta}_n \\ &= \int \hat{\theta}_n p(\hat{\theta}_n \mid D)\, d\hat{\theta}_n \\ &= E[\hat{\theta}_n \mid D] \end{aligned} \tag{27}$$

This conditional expectation, $E[\hat{\theta}_n \mid D]$, is the operational definition of $\rho_{ij}(c_p)$, the rating that member i would give to member j with respect to the context, c_p.

If $p(\hat{\theta}_n \mid D)=Beta(c_1+p, c_1+c_2+n)$, then the formula for $E[\hat{\theta}_n \mid D]$ is given by:

$$E[\hat{\theta}_n \mid D]=\frac{c_1+p}{c_1+c_2+n} \tag{28}$$

In order to calculate $E[\hat{\theta}_n \mid D]$, we need to know c_1 and c_2, which are determined by prior assumptions. Therefore, we need to assume a probability distribution for the prior, $p(\hat{\theta})$, in order to determine c_1 and c_2.

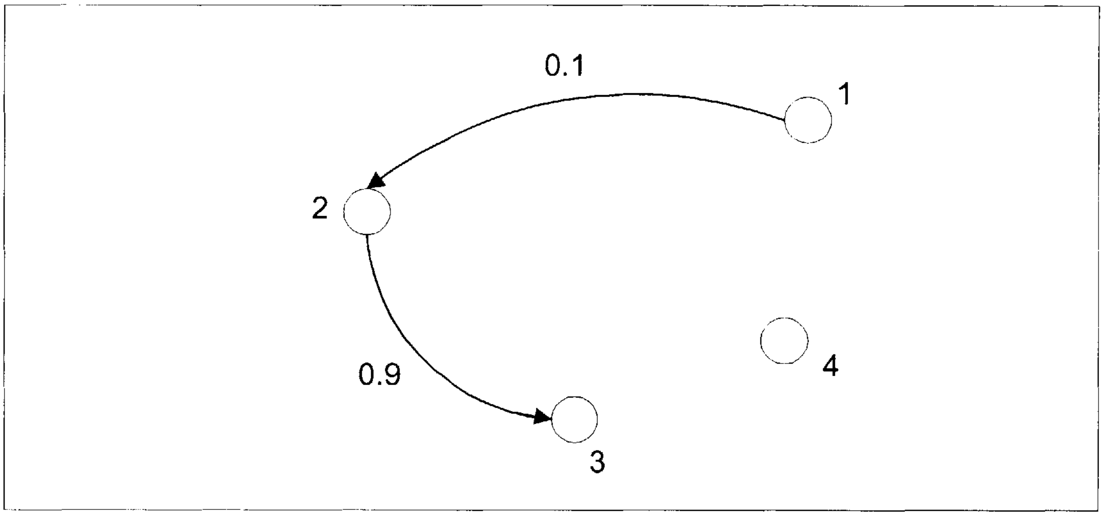

Figure 2-3: Sample ratings between members

2.4.2 Prior assumption for complete strangers

Member i and member j are defined to be complete strangers in context c_p if there are no paths between them, taking into account all the rating links between members. In Figure 2-3, member 1 and member 4 are complete strangers. In the example of Thai cuisine and Thai restaurants, all the members are not allowed to interact with one another before the formation of the social network. Therefore, after they have visited the restaurants, but before the social network is formed, the members of the community were complete strangers to one another.

If member i and member j are complete strangers, an ignorance assumption is made. When these 2 strangers first meet, their estimate for each other's reputation is uniformly distributed across the rating domain, therefore the prior is given by:

$$p(\hat{\theta}) = \begin{cases} 1, & 0 < \hat{\theta} < 1 \\ 0, & otherwise \end{cases} \quad (29)$$

Since $\hat{\theta}$ follows a Beta distribution, the parameters c_1 and c_2 must both have values of 1 in order for $p(\hat{\theta})$ to follow the distribution in equation (29). Using the prior in equation

(29) to generate the parameters needed to calculate $E[\hat{\theta}_n \mid D]$ in equation (28), we can determine $\rho_{ij}(c_p)$ for selected pairs of members (i, j) and use these pair-wise ratings to form a social network for context c_p.

Member i and member j are defined to be known strangers in context c_p if there is one or more paths between them, formed by rating links between members. In Figure 2-3, member 1 and member 3 are known strangers, since they are linked by $\rho_{12}(c_p)$ and $\rho_{23}(c_p)$. The members in a community can only be known strangers after the social network has been formed for a context, c_p.

2.4.3 Ratings propagation function

For the subject, member i, the Bayesian estimate rating system has to estimate the rating that he would have given to every other member of the community if he had interacted with them directly. The system has to do that using a ratings propagation mechanism that is described as follows.

Let member j be a member in the social network that has been rated by the subject, member i, directly. Let member k be another member in the social network of n members, $k \in \{1, n, k \neq i, j\}$. The subject, member i, has not rated member k directly. Therefore, no edge from member i to member k exists in the social network.

Let $D_{ij,n}$ represent the set of n encounters between member i and member j. Let $D_{jk,m}$ represent the set of m encounters between member j and member k. Let x_{ik} be the binary random variable for member i's approval of member k in an encounter.

In order to estimate $\rho_{ik}(c_p)$ from $\rho_{ij}(c_p)$ and $\rho_{jk}(c_p)$, we have to calculate the conditional probability $p(x_{ik} = 1 \mid D_{ij,n}, D_{jk,m})$. To calculate this conditional probability, note that the set of encounters between members i and j, and that between members j and k are independent.

$$p(x_{ik}=1 \mid D_{ij,n}, D_{jk,m}) = \frac{p(D_{ij,n}, D_{jk,m} \mid x_{ik}=1)p(x_{ik}=1)}{p(D_{ij,n}, D_{jk,m})} \tag{30}$$

To simplify equation (30) further, the conditional independence between $D_{ab,n}$ and $D_{bc,n}$ given $x_{ac}=1$, if it exists at all, needs to be verified. Consider the following random variables:

Let x_{ij} be member i's approval of member j in their next encounter.

Let x_{jk} be member j's approval of member k in their next encounter.

Therefore,

$$\begin{aligned} \rho_{ij}(c_p) &= p(x_{ij}(n+1)=1 \mid D_{ij,n}) = E[\theta_{ij,n} \mid D_{ij,n}] \\ \rho_{jk}(c_p) &= p(x_{jk}(n+1)=1 \mid D_{jk,n}) = E[\theta_{jk,n} \mid D_{jk,n}] \end{aligned} \tag{31}$$

Once we have an estimate of x_{ik}, we can use equations (30) and (31) to derive the ratings propagation function for members separated by 2 links, $\rho_{ik}(c_p) = f_2(\rho_{ij}(c_p), \rho_{jk}(c_p))$. The ratings propagation function for the Bayesian estimate rating system is given by:

$$\begin{aligned} & p(x_{ik}=1 \mid D_{ij,n}, D_{jk,m}) \\ &= p(x_{ij}=1 \mid D_{ij,n})p(x_{jk}=1 \mid D_{jk,n}) + p(x_{ij}=0 \mid D_{ij,n})p(x_{jk}=0 \mid D_{jk,m}) \\ & \Rightarrow \rho_{ik}(c_p) = \rho_{ij}(c_p)\rho_{jk}(c_p) + (1-\rho_{ij}(c_p))(1-\rho_{jk}(c_p)) \end{aligned} \tag{32}$$

For determining the estimated ratings of members that are more than 2 links away from the subject, member i, in the social network, we have to employ equation (32) iteratively, For example, in order to obtain $\rho_{il}(c_p)$, we can first calculate $\rho_{ik}(c_p)$ using equation

(32). Then, assuming that $\rho_{kl}(c_p)$ can be obtained from the social network, we can reapply equation (32) again using $\rho_{ik}(c_p)$ and $\rho_{kl}(c_p)$ as the parameters for $f_2(\cdot)$:

$$\rho_{il}(c_p) = f_2(\rho_{ik}(c_p), \rho_{kl}(c_p)) \tag{33}$$

Therefore, it is possible for the Bayesian estimate rating system to apply the rating propagation function in equation (32) one or more times to calculate the estimated rating that would be given by subject, member i, for every other member in the network. In this way, the rating system can select the most reputable members in the social network as would be regarded by the subject.

3. Simulations and Experiments

3.1 Description of the simulation system

In order to evaluate the effectiveness of the proposed rating systems, various experiments were conducted on a custom-built simulation system. The simulation system simulates an online community of users that subscribes to a web-based service where they have to rate resources as well as other users. An example of such resources are restaurants, and the web-based service could be a restaurant sanctioning service that recommends restaurants based on the ratings given by users. I would use the restaurant example to describe the rest of the system.

Using this simulation system, we can vary the number of users and their individual preferences depending on the requirements of each experiment as well as the number and attributes of restaurants. In addition to that, we can specify the number and locality of restaurants that each user would rate, as well as the number of other users that each user would rate. The system would conduct the simulation and return the result set, which consists of the restaurant and user ratings. This result set can then be analyzed by an analysis package that is specific to each experiment.

The simulation system consists of 4 major components. They are the users specification, the restaurants specification, the simulation engine and the analysis package. The users specification, the restaurants specification and the analysis package have to be tailored for each new experiment in a format that interfaces correctly with the simulation engine. Figure 3-1 shows the major components of the simulation system.

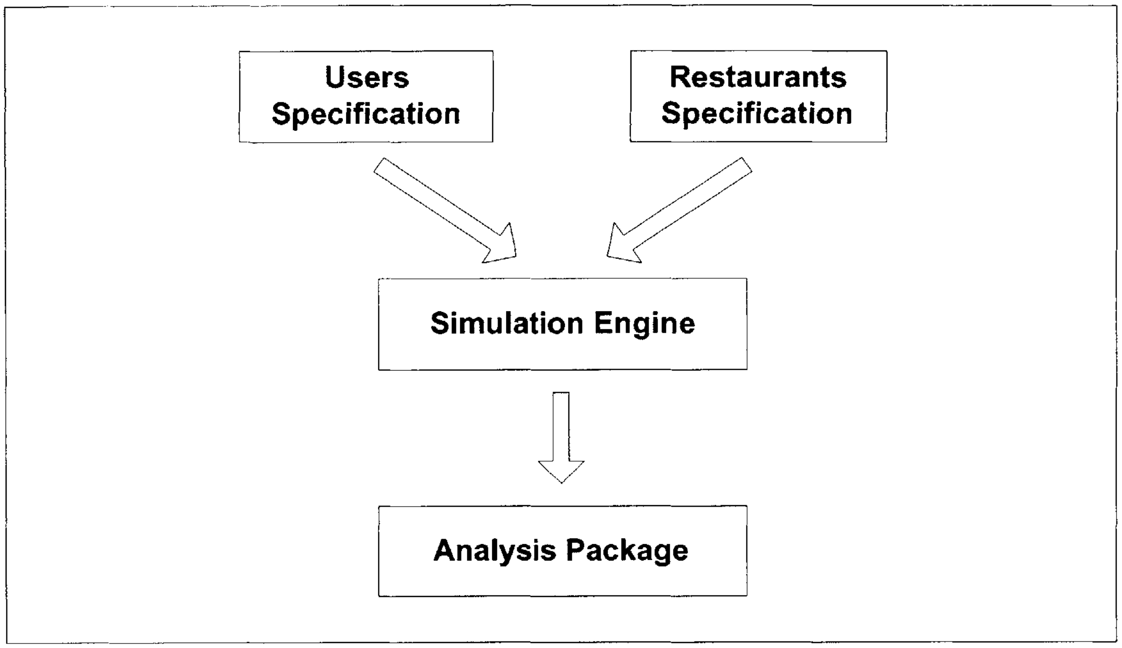

Figure 3-1: Schematic Diagram of Simulation System

3.1.1 Users Specification

The users specification contains information about the number of users to be included in the experiment, as well as the attributes of each user. These attributes include the home city of the user, the number of restaurants that he will rate and the preference scores of the user for various contexts. The user will only rate restaurants that are in his home city. The preference score of a user for any context is within the range of zero and one. The simulation system generates the rating that a user gives for a restaurant in a given context based on the user's preference score as well as the restaurant's attributes.

User Name		User 1
City of Residence		Cambridge, MA
Number of restaurants to be rated		300
Preference score	Japanese cuisine	0.9
	American cuisine	0.1
	French cuisine	0.1
	Ambience	0.5
	Other	0.5

Figure 3-2: Contents of a sample Users Specification

Figure 3-2 shows a sample users specification that describes the attributes of two users. Both users will only rate restaurants found in the city of Cambridge, they will rate up to 300 restaurants. The preference score of user 0 for the context of 'Japanese restaurants' is 0.9, whilst that of user 1 is 0.7, which means that user 0 is more likely to give a higher score for restaurants with a Japanese context than user 1 (ie. user 0 likes Japanese restaurants more than user 1). In the attribute list of each user, any number of different contexts can be specified, however, the context, 'other', must always be there. The preference score of, 'other', is used by the simulation system to generate a user rating for a restaurant in a context that is not specified in the above file, such as 'Italian'.

3.1.2 Restaurants Specification

The restaurants specification contains information about the number of restaurants to be included in the experiment, as well as the attributes of each restaurant. These attributes include the city in which the restaurant is located, the cuisine type of the restaurant, the quality score of the restaurant for cuisine (which is a context), as well as the quality score for any other contexts. The quality score of the restaurant for a context is within the range of zero to one, this score will affect how a user rates it in that context. For example, a restaurant of cuisine type Japanese with a low quality score for cuisine would tend to receive a lower rating from users for the context of 'cuisine'.

Restaurant Name		Restaurant 1
City		Cambridge, MA
Cuisine		American
Quality score	Food	0.5
	Ambience	0.5

Figure 3-3: Contents of a sample Restaurants Specification

Figure 3-3 shows a sample restaurants specification that describes the attributes of 200 restaurants. All 200 restaurants are located in the city of Cambridge. Of these, 100 are American restaurants with a quality score of 0.5 for cuisine. A score of 0.5 denotes that the restaurants are average with regards to that context. A higher quality score for a context would make it more likely for the restaurant to receive a better rating from users for that context. The other 100 restaurants above are Japanese restaurants, also with a quality score of 0.5 for cuisine. All the restaurants have a quality score of 0.5 for the context 'ambience'. In the attributes list for a restaurant, any number of different contexts can be specified, together with a quality score for each, in addition to 'ambience' that is shown in the file in Figure 3-3.

3.1.3 Simulation Engine

The simulation engine does a simulation of the various users in an experiment going about rating restaurants according to each of their personal attributes and the restaurant characteristics. The number of restaurants that each user would rate is described in his list of attributes in the users specification. For each user, the simulation engine would randomly pick restaurants in the home city of the user for him to rate. The user would then give a rating for every context of the restaurant as described in the restaurant's attributes in the restaurants specification. For example, the restaurants described in the restaurants specification in Figure 3-3 would receive two ratings from each user who chooses to rate it, one rating with respect to the context of 'cuisine' and the other with respect to the context of 'ambience'. The rating that a user gives to a restaurant with respect to any context is a function of his preference score for that context as described in the users specification, and the restaurant's quality score for that context. This rating mechanism is as follows:

Let c be the context and x_c be the rating that a user gives to a restaurant with respect to that context. Let p_c be the preference score that the user has for context c. Then,

$$X_c \sim \frac{1}{n} Binomial(n,\ p_c)\text{, where n = 20}$$

$$Mean(X_c) = p_c,\ Variance(X_c) = \frac{1}{n} p_c(1 - p_c)$$

Let q_c be the quality score for the rated restaurant with respect to context c. The above distribution for X_c only holds when $q_c = 0.5$, that is, the rated restaurant is average with respect to context c. When q_c is not equal to 0.5, then

$$X_c \sim \frac{1}{n} Binomial(n, Mean(X_c))\text{, where n = 20}$$

$$Mean(X_c) = p_c + \frac{(q_c - 0.5)(q_c - p_c)}{0.5}$$

Therefore, it can be seen that the mean value of x_c deviates from p_c if q_c is not equal to 0.5. This makes sense because the rated restaurant should receive a higher rating if it has a higher quality score for the context with respect to which it is being rated. The range of Mean(X_c) is between zero and one.

For each occasion when a user has to rate a restaurant with respect to a particular context, the simulation engine first determines the distribution of the rating, X_c, based on p_c and q_c. Then, it randomly chooses a number between zero and one, and sets the number to be CDF(x_c). The simulation engine will determine x_c to be the rating that a user would give to the rated restaurant with respect to context c. Figure 3-4 shows the distribution of X_c

when $p_c = 0.7$ and $q_c = 0.5$. From the distribution, we can see that the median is 0.7, which means that the user will most likely give a rating of 0.7 to the restaurant with respect to context c. As we move further away from $X_c = 0.7$, the likelihood that the user would choose those values of X_c as the rating decreases.

The simulation engine repeats the same process for every user until all the users have rated the restaurants that they wanted to rate. The ratings that each user gives are stored in a vector that can be passed to the analysis package, which will analyze the results of the restaurant ratings.

3.1.4 Analysis Package

The analysis package is specific to each experiment that is to be conducted using the simulation system. The simulation engine will pass the following information to the analysis package: the restaurant ratings given by all the users, the attributes of the users, and the attributes of the restaurants. Using this information, the analysis package can construct a social network and test the effectiveness of the proposed rating systems.

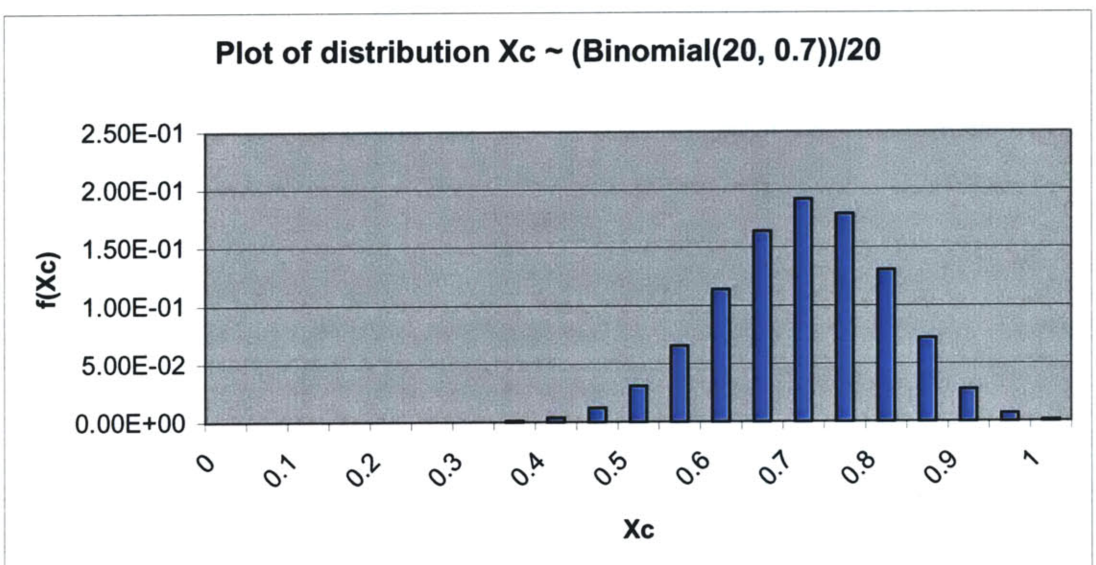

Figure 3-4: Plot of distribution of X_c for $p_c = 0.7$ and $q_c = 0.5$

3.2 Restaurant Rating Simulation

In this set of experiments, we create a group of users to rate some restaurants with respect to the context of 'ambience'. The preference score of these users for the context of 'ambience' is a random number between zero and one. Every user will randomly select some restaurants, and give each of them a rating with respect to ambience. All of the restaurants in this set have an average quality score for the context of 'ambience'.

In each simulation run, the ratings that every user gives for all of the restaurants that he has rated with respect to the context of 'ambience' are calculated. Then, based on these restaurant ratings, the ratings that every user gives to a number of other users are calculated. These user-to-user direct ratings are needed to form a social network. This social network is the framework that we use to verify the effectiveness of the various rating systems in recommending to a user other users that he is most likely to agree with when it comes to rating a restaurant's ambience.

The rating that a user gives to another user depends on the number of restaurants that they share in common as well as the their opinions of these shared restaurants. When user A decides to rate user B for the context of 'ambience', user A will first ask for the set of restaurants that user B has rated with respect to the context of 'ambience'. He will then determine the set, R_{AB} of restaurants that both of them have rated. The rating that user A will give to user B is determined by examining the set R_{AB}. User A will give user B a rating that reflects the extent to which his restaurant ratings agrees with user B's. Since the rating for each restaurant with respect to the context of 'ambience' ranges from zero to one, there is no straightforward way to determine such agreement. Therefore, we designed two algorithms that user A can use to rate user B based on R_{AB}. The first algorithm is the threshold algorithm and the second algorithm is the agreement likelihood algorithm.

Step 1: Let p be the agreement index and n be the size of R_{AB}.
Step 2: Determine $\rho_a(r_i)$ and $\rho_b(r_i)$, the ratings that user A and user B gives for restaurant r_i respectively, where $r_i \in R_{AB}$.
Step 3: If ($\rho_a(r_i) > 0.5$ and $\rho_b(r_i) > 0.5$) OR ($\rho_a(r_i) < 0.5$ and $\rho_b(r_i) < 0.5$), then {p = p + 1} else {p = p}
Step 4: Repeat steps 2 and 3 from i = 0 to i = n
Step 5: Rating that user A gives user B, $\rho_{ab} = \frac{p}{n}$

Figure 3-5: The threshold algorithm for calculating ρ_{ab}

Figure 3-5 shows the steps in the threshold algorithm. The main feature of the threshold algorithm is the use of a threshold of 0.5 to differentiate a good restaurant rating from a bad one. A restaurant rating of less than 0.5 would be considered a bad rating and a rating of more than 0.5 would be considered a good rating. If user A and user B both gave a restaurant good ratings, then their views on the restaurant are deemed to be the same and p would be increased by one. User A's final rating of user B would be the proportion of restaurants in R_{AB} for which both are in agreement.

Figure 3-6 shows the steps in the agreement likelihood algorithm. The agreement likelihood algorithm treats a restaurant rating as the probability that the user would like the restaurant with respect to the rating's context. Therefore, for each restaurant r_i from R_{AB}, the probability that users A and B will like both like r_i is given by $\rho_a(r_i)\ \rho_b(r_i)$, and the probability that they will both dislike r_i is given by $(1 - \rho_a(r_i))(1 - \rho_b(r_i))$. The likelihood that user A will agree with user B is therefore given by the sum of the two probabilities, which is used to update p. User A's final rating of user B would be the proportion of restaurants in R_{AB} for which both are in agreement.

Step 1: Let p be the agreement index and n be the size of R_{AB}.
Step 2: Determine $\rho_a(r_i)$ and $\rho_b(r_i)$, the ratings that user A and user B gives for restaurant r_i respectively, where $r_i \in R_{AB}$.
Step 3: $p = p + \rho_a(r_i)\ \rho_b(r_i) + (1 - \rho_a(r_i))(1 - \rho_b(r_i))$
Step 4: Repeat steps 2 and 3 from i = 0 to i = n
Step 5: Rating that user A gives user B, $\rho_{ab} = \frac{p}{n}$

Figure 3-6: The agreement likelihood algorithm for calculating ρ_{ab}

We can use either of the two algorithms above to provide reasonable user-to-user direct ratings between any pair of users. In each simulation run, if we were to calculate the rating that each user would give for every other user, it would be trivial to find for any user the set of other users whose opinions he is most likely to agree with. To find this set for user A for instance, we just have to rank all the other users in the social network according to the rating that A has for each of them. However, in most real-life scenarios, it is not possible for a user to rate every other user in the social network. Therefore, one important use of rating systems is to propagate ratings across one or many users. In Figure 3-7, user A has given user B a rating of ρ_{ab} and user B has given user D a rating of ρ_{bd}, but user A has not rated user D. The effectiveness of various rating systems in estimating ρ_{ad} based on ρ_{ab} and ρ_{bd} is highly important because it means that user D can be included in the list of users that could be recommended to user A. Without the rating propagating mechanism of the rating systems, it would only be possible to recommend user B and user C to user A. However, ρ_{ad} might in fact be greater than both ρ_{ab} and ρ_{ac}. In which case, user D should be the most highly recommended user for user A.

In this set of experiments, we compare the effectiveness of two personalized rating systems and one non-personalized rating system in correctly generating for every user in the social network a ranked list of neighboring users. The neighboring users of a subject user include all the users that the he has directly rated (first degree friends), as well as those that are two rating links away (second degree friends). The 'subject' user refers to a user whose ranked list of neighboring users the simulation is currently calculating. As the analysis package runs through the list of users, every user in the social network will take his turn to be the subject user. This ranked list of neighboring users is useful because the subject user can tell from the list the users whom he is most likely to agree with, and ask for their opinion.

In Figure 3-7, the neighboring users of user A consist of his first-degree friends, user B and user C, as well as his second-degree friends, user D, user E and user F. In the list, the neighboring users of a subject user are ranked according to the likelihood that he would agree with their opinions with respect to the context of 'ambience'. The rating propagation mechanism of each rating system is used to calculate the estimated rating that the subject user would give to his second-degree friends. This facilitates the ranking of the second-degree friends with the first degree ones.

The two personalized rating systems that are tested in this experiment are the Bayesian Estimation rating system and the Preference-based rating system, both of which are described in the previous sections. The non-personalized rating system that is tested is the Centrality measure. The propagation mechanism for the Bayesian Estimation rating system is given by:

$$\rho_{ik} = \rho_{ij}\rho_{jk} + (1 - \rho_{ij})(1 - \rho_{jk})$$

The propagation mechanism for the Preference-based rating system is given by:

$$\rho_{ik} = [(2\lambda_i - 1)(2\lambda_i\rho_{jk} - \lambda_i - \rho_{jk} + \rho_{ij}) + (1 - \lambda_i)(2\rho_{ij} - 1)] / (2\rho_{ij} - 1)$$

The rating, ρ_{ik}, as calculated by the Preference-based rating system will be rounded down to 1 if it exceeds 1 and rounded up to zero if it is negative. When using the same rating system, and $\rho_{ij} = 0.5$, ρ_{ik} is set to be 0.5. The variable λ_i is the average of all the restaurant ratings that user i gave to all the restaurants for the context involved. It is an estimate of user i's preference score for that context.

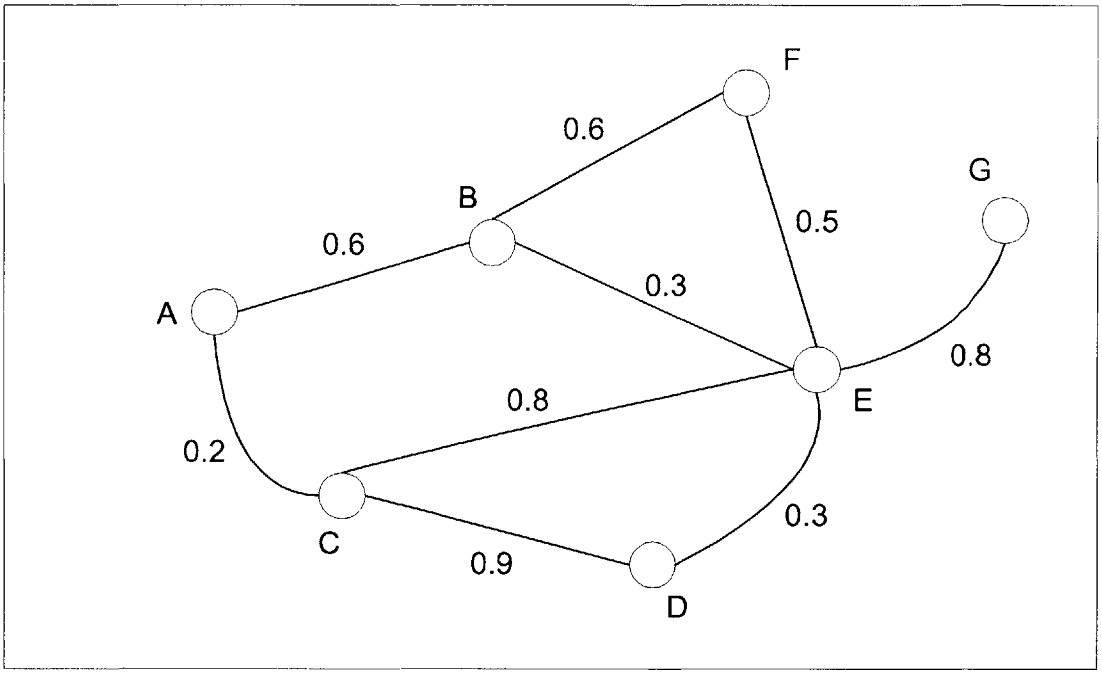

Figure 3-7: A sample social network of 7 users

When there are two or more possible paths between two users separated by two degrees, each path will produce a different ρ_{ik}. For example, in Figure 3-7, there are two paths linking user A and user E, one that passes through user B and the other that passes through user C. In order to reconcile these different values of ρ_{ik}, there are 3 different strategies that we employ. The first strategy is to find the average of these values of ρ_{ik}, the second is to weigh each ρ_{ik} by their respective ρ_{ij}, and the third strategy is to use the ρ_{ik} for the path with the largest ρ_{ij}. These strategies will be compared in the following experiments.

The error measure for the first experiment to the third experiment will be the 'ranking error measure', described as follows. For every user in the social network, the simulation run would use the propagation mechanism of the rating system to generate the ratings for his second-degree friends. Using these ratings and the direct ratings given to his first-degree friends, a list of neighboring users is generated. All the other users in the network are each given a rating of 0.5 (assuming the subject user is risk-neutral towards strangers) and added to this list. The users in the list are then ranked according to their ratings. Next, we compare the ranked list of all users each rating system generates to the correct ranked list. The correct ranked list is generated by calculating the direct rating that the subject user would have given to all users in the network, and then ranking all these users using these direct ratings.

The ranked list generated by the rating system and the correct ranked list each contains all the users in the experiment except the subject user. The ranking error measure calculates the difference in the ordering of users in the two lists. It does that by first finding the sum of the absolute differences of each user's ranking in the two lists. This sum is then normalized by dividing it by the maximum possible error for the number of users. The ranking error measure is given by this normalized value. This normalization allows the ranking error measure derived from two pairs of ranked lists of different number of users to be comparable. The maximum possible value for the ranking error measure is when the two lists of n users each are exactly reversed in order.

Let $X = \{x_1, x_2, ..., x_n\}$ and $Y = \{y_1, y_2, ..., y_n\}$ be two ranked lists with n users each. Then, the ranking error measure for X and Y would be given by:

$$\textit{Ranking error measure}\langle X, Y\rangle = \frac{\sum_{i=1}^{n} | x_i - y_i |}{V}$$

where V, the normalization factor, is given by:

$$V = \textit{Maximum error} = \begin{cases} \dfrac{(n+1)(n-1)}{2}, n \in \textit{Odd} \\ \dfrac{n^2}{2}, n \in \textit{Even} \end{cases}$$

An example of the calculation of the ranking error measure for the two lists below:

$$Ranking\ error\ measure\left\langle \begin{pmatrix} user1:4 \\ user2:2 \\ user3:1 \\ user4:3 \end{pmatrix}, \begin{pmatrix} user1:3 \\ user2:1 \\ user3:4 \\ user4:2 \end{pmatrix} \right\rangle = \frac{|4-3|+|2-1|+|1-4|+|3-2|}{8}$$

$$= 0.75$$

3.3 Movie Rating Experiments

To test the theory and confirm the simulation results, the same set of experiments performed using the simulation system are also performed on the MovieLens data set [19]. This dataset consists of 100,000 ratings by 943 users on 1682 movies of 19 different genres. In each experiment, every user is assigned a set of direct neighbors. The objective is to rank the indirect neighbors of a subject user according to the estimated rating that he would give them. As in the simulation experiments, the ranking error measure is also used. Propagation of ratings to indirect neighbors are tested using the same techniques as the restaurant rating simulations.

4. Results and Discussion

4.1 Comparing personalized and non-personalized rating systems

The goal of this set of 3 experiments is to compare the effectiveness of personalized rating systems against non-personalized rating systems in performing the task of recommending to a user other users in the community whose opinions he is most likely to agree with. The rationale behind the development of personalized rating systems as described in the previous sections is that tastes can vary substantially from individual to individual, and such taste differences justify the need to personalize ratings for every user. It is therefore crucial that we verify the performance of personalized rating systems against non-personalized systems derived from social network theories.

4.1.1 Evaluation of rating systems with varying network density

For the first experiment, we would like to determine the capabilities of the rating systems in accurately ranking the neighboring users of a subject user as we vary the density of the social network. The density of the social network is varied by changing the number of other users that each user rates. Since we are comparing the different rating systems in this experiment, for reconciling multiple paths between 2 users in each system, the strategy with the least error will be used for comparison. Therefore, the results represent the best possible performance by each rating system in ranking the neighboring users. There are a total of 300 restaurants defined for this experiment, and each user would rate 100 restaurants, resulting in relatively accurate user-to-user direct ratings.

Figure 4-1 shows the performance of each system if we use the threshold algorithm to calculate the rating that would be given by one user to another. Let x be the number of users that every user rates. For each value of x, 10 simulation runs are executed for each rating system and an average error measure for the rating system at that value of x is calculated and plotted. The total number of users is 40, and x is varied from 1 to 39.

In Figure 4-1, the line denoted 'Control' is generated by calculating the error if the ranked list of neighboring users for each subject user were to be formed by randomly inserting his second-degree friends into a ranked list of his first-degree friends. This control functions as a baseline against which we can compare the rating systems that we are testing. As long as a rating system ranks the second-degree friends better than chance, we would expect it to perform better than the control.

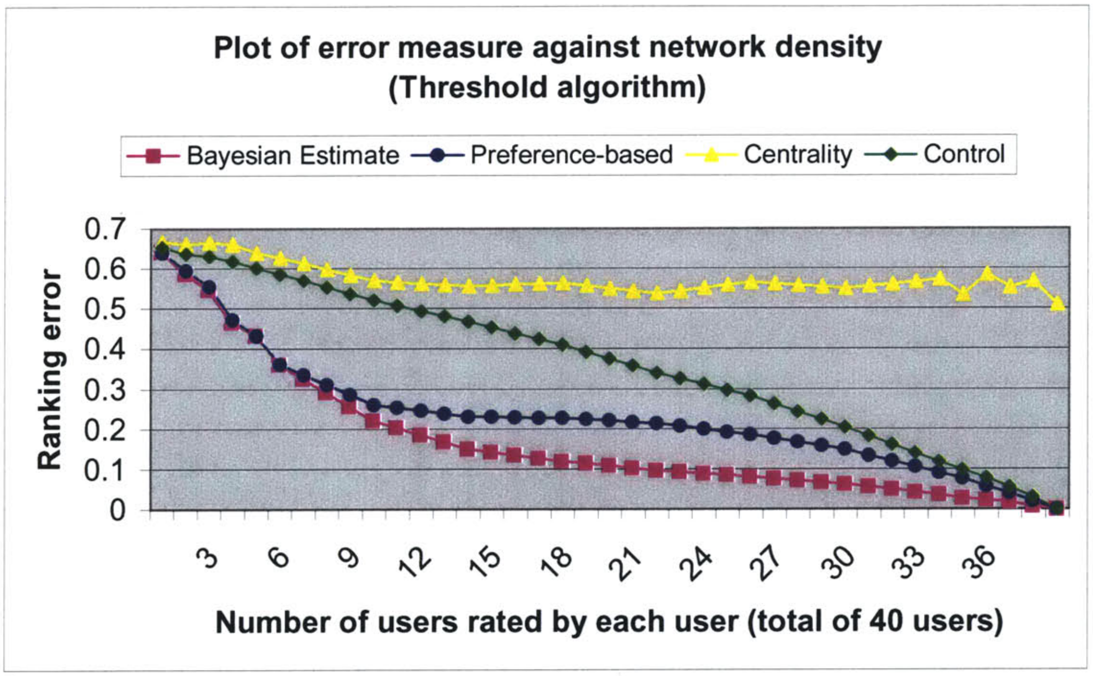

Figure 4-1: Plot of error measure for each rating system against network density, using the Threshold algorithm to calculate user-to-user direct ratings

When x = 39, Figure 4-1 shows that the ranking error for both the Bayesian estimation and the Preference-based rating systems are zero. This result is expected because when each subject user has rated every other user in the social network, he would be able to rank all of them without any error. As x decreases from 39, the effectiveness of the rating propagation mechanism of each rating system begins to affect the error measure because the ranking now includes some users that were not directly rated by the subject user. When the social network becomes denser, the Bayesian estimate and Preference-based systems perform better because there are more paths leading from the subject user to his second-degree friends, allowing them to determine the estimated rating for these second-degree friends more accurately. For a sparse network, the Preference-based system performs as well as the Bayesian estimate system. As network density increases, the error generated by the Bayesian estimate rating system decreases faster than that of the Preference-based system, giving it better performance over a large range of x.

The Centrality-based rating system, however, performs consistently worse than the control. It starts off with a high ranking error when x is small, and the error decreases at the same rate as the control until x = 12. Thereafter, the error remains constant as x increases. This experiment therefore provides some evidence that the Centrality-based rating system might not be suitable for determining the ranking of neighboring individuals for a subject user, especially for dense networks, where it performs much worse than the control. The Bayesian estimate and Preference-based rating systems both make use of the fact that the first-degree friends of a subject user have been directly rated and therefore can be accurately ranked before considering the second-degree users. However, the higher error of the Centrality-based rating system could be attributed to the fact that it ranks all users simultaneously using the eigenvector method. Therefore, other than the Centrality-based rating system, the other rating systems have improved performances as network density increases.

Figure 4-2 shows the performance of each system if we use the agreement probability algorithm to calculate the rating that would be given by one user to another. For each value along the x-axis, 10 simulation runs are executed for each rating system and the average error calculated and plotted. The number of users, x, rated by each subject user is varied from 1 to 39, out of a total of 40 users in the social network. As in Figure 4-1, the line denoted 'Control' is the control rating system, formed by randomly inserting second-degree friends into a ranked vector of first-degree friends.

The error plot for each rating system in Figure 4-2 has some characteristics that are similar to that of Figure 4-1. In Figure 4-2, the performances of both the Bayesian estimate and Preference-based systems increase with network density, and are significantly better than the control rating system. As the network density increases, multiple paths leading to the same second-degree friend allow both rating systems to calculate a more accurate estimation of the rating that the subject user should give to him.

The Preference-based rating system performs better than the Bayesian estimate system over the entire range of network densities. Its ranking error decreases drastically initially, reaching a 10% level when x is only 12. Therefore, it is the preferred rating system when the agreement likelihood algorithm is used to calculate user-to-user direct ratings.

The performance of the Centrality-based system is very poor over the range of network densities. The ranking error that it generates is very high for sparse networks, comparable with that generated by the control. As network density increases, its ranking error only decreases marginally, so much so that its performance becomes significantly worse than the control for denser networks. This experiment therefore provides evidence that the Centrality-based rating system is not suitable for ranking the immediate neighbors of a subject user when the agreement likelihood algorithm is used.

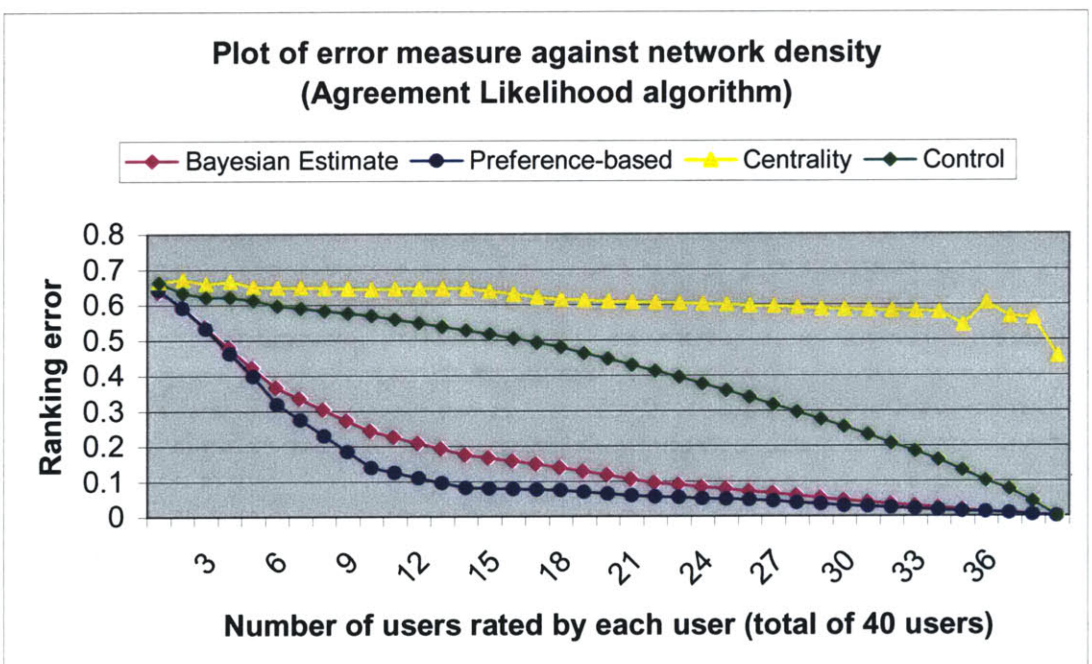

Figure 4-2: Plot of error measure against network density for each rating system, using the Agreement Likelihood algorithm to calculate ratings

Figure 4-3 shows the performance of each system if we use the MovieLens dataset. As each user gets to know the true reputation of additional direct neighbors for a specific context of rating, more information leads to a reduction in ranking error. The Centrality-based system leads to the most error while the Preference-based system leads to the least error. The Bayesian estimate rating system has a performance that closely trails that of the Preference-based system. Using the MovieLens dataset, the performance of each rating system is to the first approximation the same as those derived from the simulations as shown in figures 4-1 and 4-2.

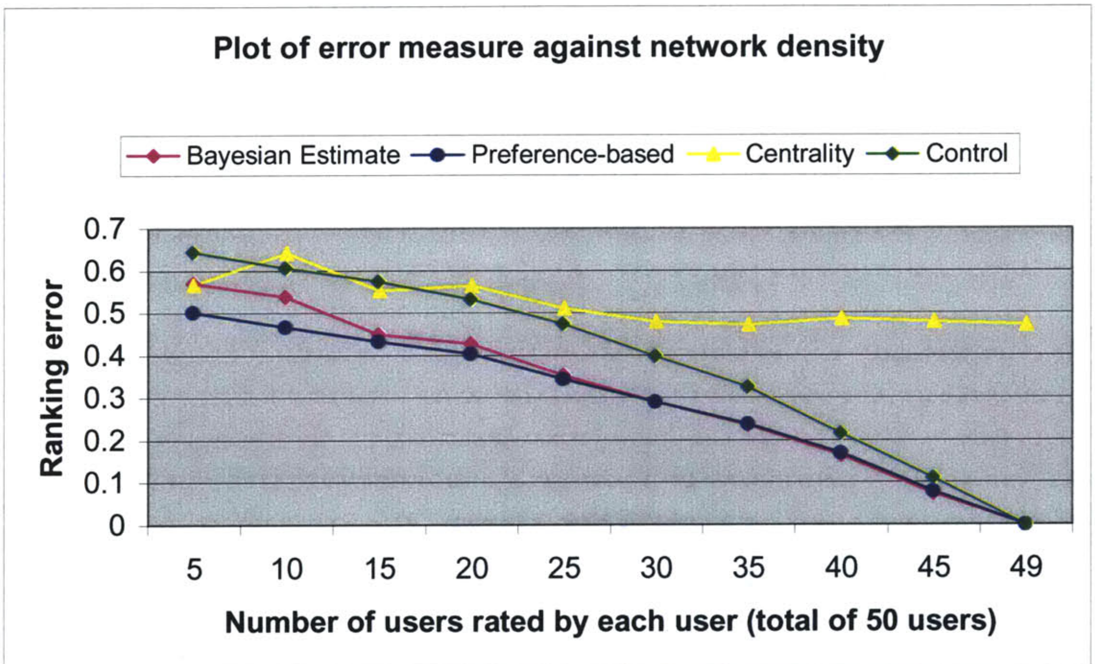

Figure 4-3: Plot of error measure against network density for each rating system, using the MovieLens dataset

4.1.2 Evaluation of rating systems with varying network

For the second experiment, we would like to determine the capabilities of the rating systems in accurately ranking the neighboring users of a subject user as we vary the size of the social network. The objective of this experiment is to test the scalability of each rating system, that is, to determine if their performance could be maintained as the number of nodes in the network increases. The percentage of the total number of users rated by each subject user is fixed at 50% as we vary the size of the network. A percentage of 50% was chosen in order to avoid tail effects that could be caused by a large or small percentage. For reconciling multiple paths between 2 users in each rating system, the strategy yielding the least error is used. Therefore, the results of the experiment represent the best possible performance by each rating system in ranking the neighboring users. There are a total of 300 restaurants defined for this experiment, and each user rates 100 restaurants, resulting in a relatively well-informed network.

Figure 4-4 shows the performance of each system if we vary the number of users in the social network in the case where the threshold algorithm is used to calculate the user-to-user direct ratings that would be given by one user to another. For every network size, 10 simulation runs are executed for each rating system and an average error measure for the rating system is calculated and plotted. The size of the network is varied from 20 users to 130 users to determine if the performance of each rating system can be maintained despite the increase in network size. The line denoted 'Control', as in the previous experiment, is the ranking error generated by the control rating system.

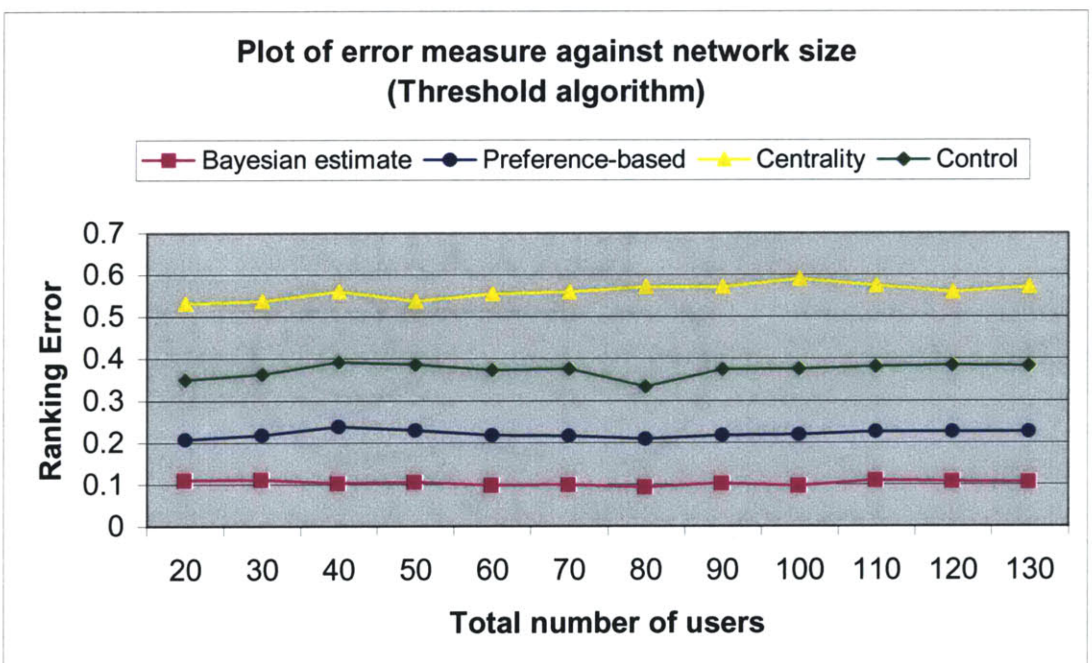

Figure 4-4: Plot of error measure against network size for each rating system, using the Threshold algorithm to calculate user-to-user direct ratings

The plot in Figure 4-4 shows that the ranking errors for all the rating systems are almost constant over the entire range of values of network size. This experiment provides evidence that all the rating systems tested here are highly scalable, and thus are able to maintain their performance with increasing network size. The ranking of the rating systems according to their performance is consistent with that in Figure 4-1 where the network size is 40 and the number of other users that each user rates is 20, that is, when each user rates 50% of the total number of users.

The Bayesian estimate rating system has the smallest ranking error of about 10%, and exhibits the smallest fluctuation among all the rating systems. The Preference-based system has the next best performance for this combination of parameters, and exhibit minimal fluctuation about an error of 22%. Both these systems perform better than the control rating system with a ranking error that fluctuates about 35%. The Centrality-based rating system, although having the worst performance at 55% error, also manages to maintain a rather steady error plot over the range of network sizes.

Figure 4-5 shows the performance of each system if we vary the number of users in the social network in the case where the agreement likelihood algorithm is used to calculate the user-to-user direct ratings that would be given by one user to another. For every network size, 10 simulation runs are executed for each rating system and an average error measure for the rating system is calculated and plotted. The size of the network is varied from 20 users to 130 users to determine if the performance of each rating system can be maintained despite the increase in network size.

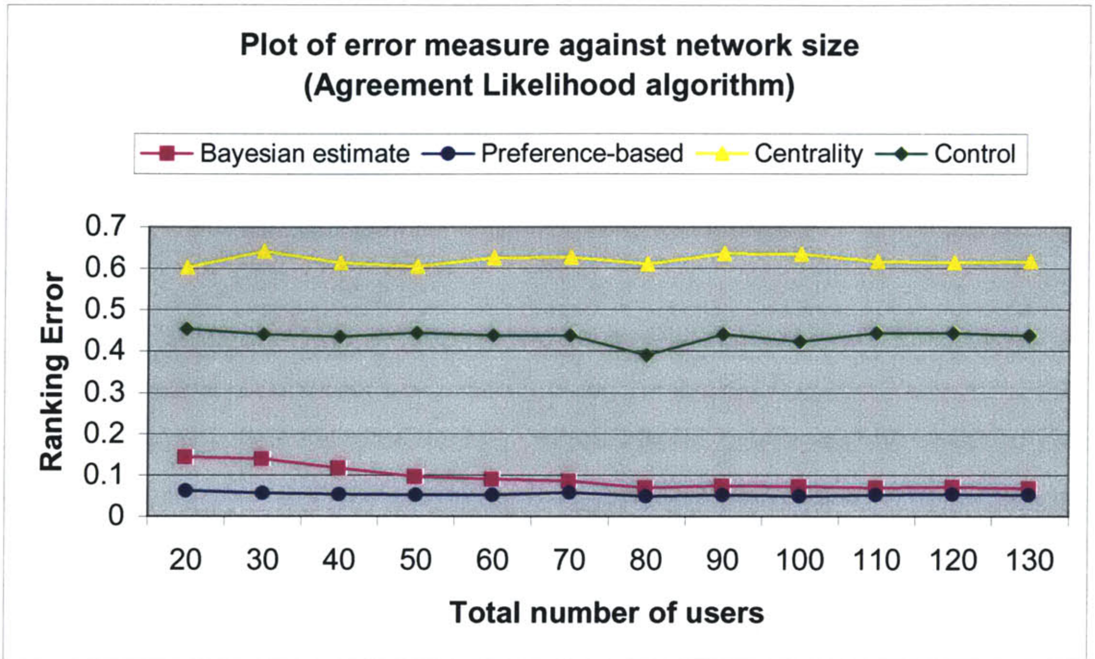

Figure 4-5: Plot of error measure against network size for each rating system, using the Agreement likelihood algorithm to calculate ratings

The plot in Figure 4-5 shows that the ranking errors of both the Centrality-based system and the control system remain almost constant over the entire range of network size. The ranking error generated by the Centrality-based system fluctuates slightly about a value of 62%, while that generated by the control system fluctuates about 43%. Although the Centrality-based rating system has a large ranking error, it can at least maintain its performance as network size increases.

The Bayesian estimate rating system and the Preference-based rating system perform 3 to 4 times better than the control system. The ranking error generated by the Bayesian estimate rating system actually decreases initially when the network size is small, and only stabilizes when the network size reaches about 50 users. Beyond 50 users, the ranking error remains almost constant at about 8%. The Preference-based rating system has the best performance among all the rating systems. The ranking error that it generates is constant throughout the entire range of network sizes, at about 5%. Therefore, this experiment provides evidence to support the fact that the Preference-based rating system and the Bayesian estimate rating system are highly scalable as they are able to maintain their good performance as the network expands.

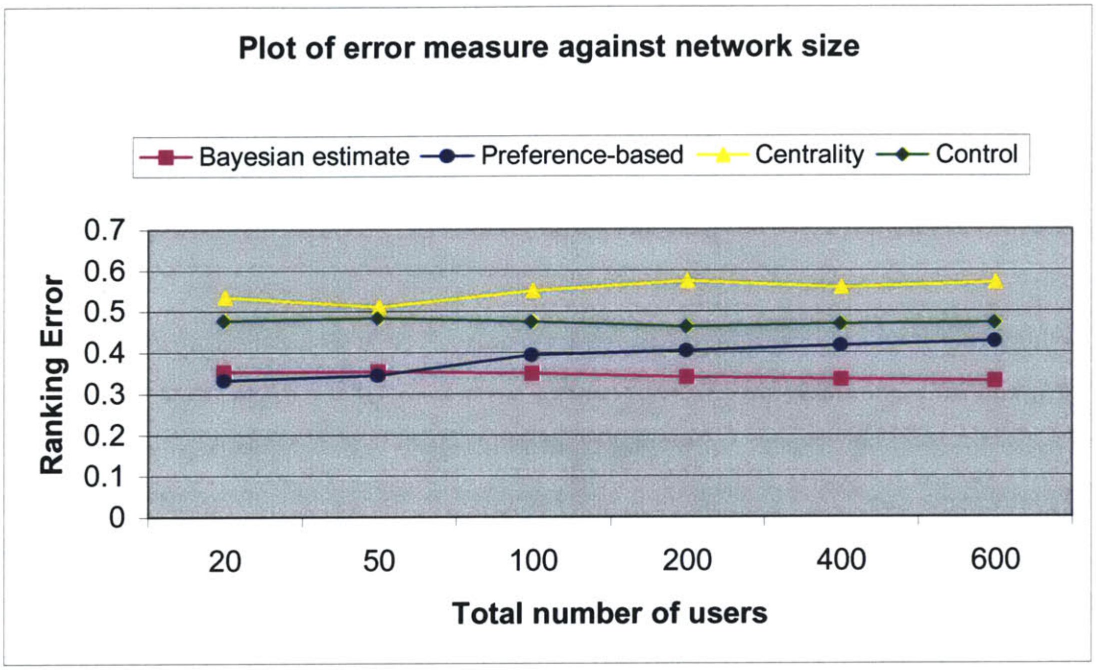

Figure 4-6: Plot of error measure against network size for each rating system, using the MovieLens dataset

Figure 4-6 shows the performance of each system if we vary the number of users in the social network in the case where the MovieLens dataset is used. Just as in the case of the simulation experiments displayed in figures 4-4 and 4-5, the network size is varied while the number of direct neighbors is kept at 50%. The performance of the Preference-based rating system seems to follow a gentle upward trend as the network size increases beyond 50 users. The results for the other rating systems do not seem to vary greatly as the size of the network is varied from 20 to 600 individuals.

4.1.3 Evaluation of rating systems with varying accuracy of ratings

For the third experiment, we would like to determine the capabilities of the rating systems in accurately ranking the neighboring users of a subject user as we vary the accuracy of the user-to-user direct ratings. The accuracy of the user-to-user direct ratings depends on the percentage of all restaurants rated by each user. As we decrease the percentage of restaurants rated by every user, we would expect the number of restaurants rated in common by any pair of users to decrease, which would in turn decrease the accuracy of the rating that a user would give another. This experiment is crucial in determining the performance of each rating system under conditions where information about each user is limited. A scenario in which that might occur is when a social network is in its infant stage, where all the users are new and have not rated enough restaurants to reveal their personal characteristics accurately.

In this experiment, 40 different users are defined, with each of them rating 20 other users, forming a moderately dense social network. The weight of the edges, representing user-to-user direct ratings, between the users in the network varies depending on the number of restaurants rated by every user, which is varied from 5% to 50% of the total of 300 restaurants defined for the experiment. As in the previous experiment, the line denoted 'Control' is generated by calculating the error if the ranked list of neighboring users for each subject user were to be formed by randomly inserting his second-degree friends into a ranked list of his first-degree friends.

Figure 4-7 shows how the performance of each rating system varies as we change the proportion of restaurants rated by every user, when the user-to-user direct ratings are calculated using the threshold algorithm. The ranking error generated by the Centrality-based rating system is almost constant at a value of 58% over the entire range of proportions. The ranking errors generated by the other rating systems start off at a high value close to 40% when the proportion of restaurants rated is just 0.05. As this proportion increases, their ranking errors begin to decrease at different rates.

The Bayesian estimate rating system has the best performance, with an error that decreases at the highest rate initially, before stabilizing at a value of 10% when the proportion of restaurants rated is 0.3. Thereafter, its performance improves only very slightly as this proportion increases. The error generated by the Preference-based rating system stabilizes at about 22% when the proportion is 0.2. These results provide evidence that the two rating systems are highly responsive to the increase in the accuracy of the user-to-user direct ratings when this accuracy is at the lower end. However, once this accuracy reaches a certain level, the rating systems become less sensitive to any further improvements in the accuracy. In other words, the marginal increase in the performance of the rating systems decreases with the increase in accuracy of the user-to-user direct ratings.

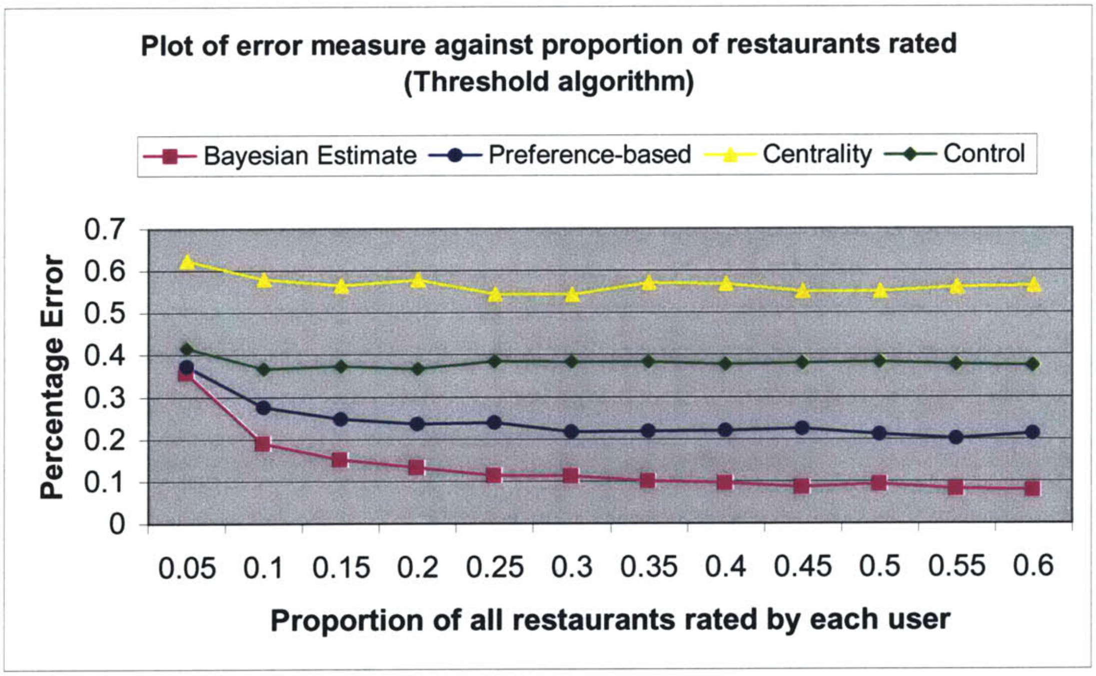

Figure 4-7: Plot of error measure against proportion of restaurants rated, using the Threshold algorithm to calculate user-to-user direct ratings

Over the entire range of proportions of restaurants rated, the Bayesian estimate rating system has the best performance, followed by the Preference-based system and then the control system. The Centrality-based system has a performance that is worse than that of the control system.

Figure 4-8 shows the variation in performance of each rating system as we change the proportion of restaurants rated by every user, when the user-to-user direct ratings are calculated using the agreement likelihood algorithm. The ranking error generated by the

Centrality-based rating system is almost constant at a value of 62% over the entire range of proportions. The control rating system has a ranking error that starts off at 50% and stabilizes at about 45% when the proportion of restaurants rated reaches 0.1.

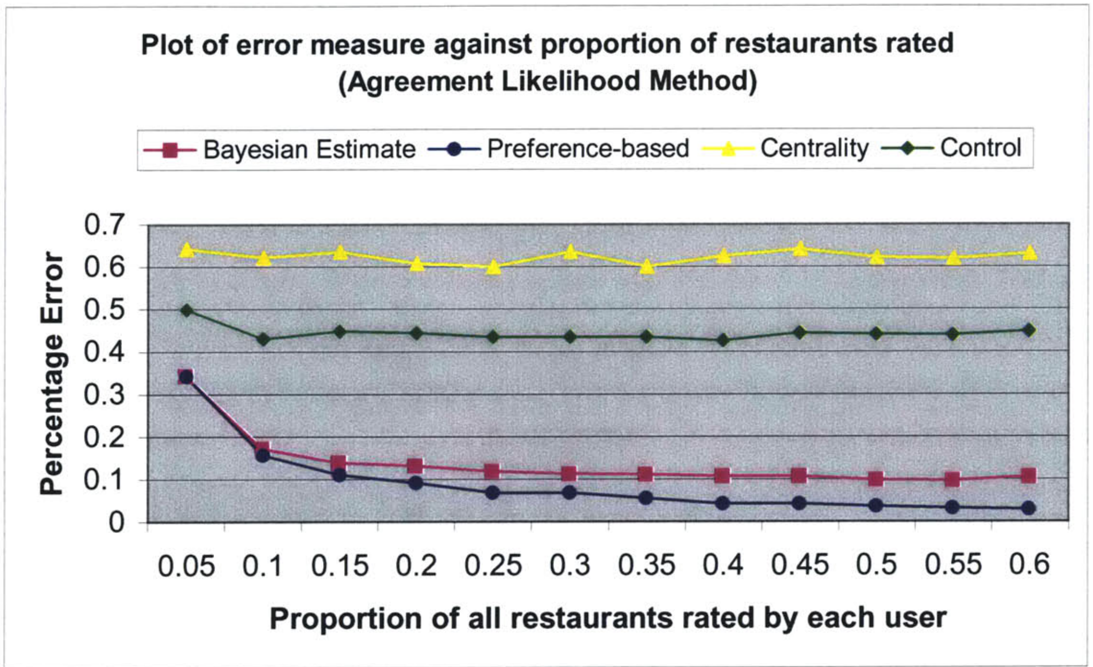

Figure 4-8: Plot of error measure against proportion of restaurants rated, using the Agreement Likelihood algorithm to calculate ratings

The ranking errors generated by the other rating systems start off at a high value close to 35% when the proportion of restaurants rated is just 0.05. As this proportion increases, their ranking errors begin to decrease at different rates. The Preference-based rating system has a ranking error that decreases rapidly until the proportion of restaurants rated reaches 0.25. Thereafter, its error continues to decrease, albeit at a slower rate. The ranking error generated by the Bayesian estimate rating system decreases rapidly until the proportion reaches 0.25, after which it stabilizes and fluctuates about a value of 10%.

These results provide evidence that, beside the Centrality-based system, all the other rating systems are highly sensitive to any increase to the accuracy of the user-to-user direct ratings initially, when this accuracy is very low. However, when this accuracy is high, any increase in it only has an effect on the Preference-based rating system, which continues to exhibit marginally better performances at the high end. Therefore, it benefits most from the improvement in accuracy and network information provided by an increase in the proportion of restaurants rated by every user over the entire range. Overall, the Preference-based rating system has the best performance, followed by the Bayesian estimate system and finally the Centrality-based system.

4.2 Comparing reconciliation strategies for parallel paths

The goal of this set of two experiments is to compare the effectiveness of different strategies in reconciling multiple paths between a pair of users when calculating user-to-user estimated ratings in personalized rating systems. In calculating the rating that one user would give to another user that is two or more links away, we have to deal with cases where more than one path connects the pair of users. Since each path will produce one rating using the rating propagation mechanism of any rating system, we must devise strategies to combine the ratings derived from these multiple paths in a meaningful manner. One example of such a pair of users is user A and user E in Figure 3-7.

The three strategies that we are comparing in this set of experiments are the averaging strategy, the weighted-averaging strategy and the maximum-weight strategy.

Let ρ_{ik}(path ijk) be the rating that user i gives to user k when path ijk is taken.

Let $\rho_{ik,\ averaging}$ be the rating given by user i to user k by the averaging strategy after reconciling the ratings derived from multiple paths between user i and user k. When user k is a second-degree friend of user i, the averaging strategy is defined as such:

$$\rho_{ik,\ averaging} = \frac{1}{n}\sum_{j=1}^{n} \rho_{ik}(\text{path ijk})$$

When user k is the second-degree friend of user i, the weighted averaging strategy is defined as such:

$$\rho_{ik,\ weighted} = [\sum_{j=1}^{n} \rho_{ij}\, \rho_{ik}(\text{path ijk})] \,/\, [\sum_{j=1}^{n} \rho_{ij}]$$

The maximum weight strategy is defined as such:

$$\rho_{\text{ik, maximum}} = \rho_{\text{ik}}(\text{path ijk}) \text{ where } \rho_{\text{ij}} = \underset{x}{Maximum}\{\rho_{\text{ix}}\}$$

In this set of experiments, like in the previous set, we create a group of users to rate some restaurants with respect to the context of 'ambience'. The preference score of these users for the context of 'ambience' is a random number between zero and one. Every user will randomly select some restaurants and give each of them a rating with respect to ambience. All of these restaurants have an average quality score for the context of 'ambience'. In each simulation run, the ratings that every user gives for all of the restaurants that he rates with respect to the context of 'ambience' are calculated. Then, based on these restaurant ratings, the ratings that every user gives to a number of other users are calculated. These user-to-user direct ratings are needed to form a social network of users. This social network is the framework that we use to verify the effectiveness of the various multiple path reconciliation strategies in recommending to a user other users that he is most likely to agree with when it comes to rating a restaurant's ambience.

For users that the subject user is to rank directly based on the number of restaurants they have both rated, the threshold algorithm and the agreement likelihood algorithm are both employed. To calculate the estimated rating that the subject user would give to a second-degree friend, the multiple path reconciliation strategies for both the Bayesian estimate rating system and the Preference-based rating system are tested.

The error measure for the first and second experiments will be the 'average difference error measure', described as follows. For every user in the social network, the simulation run would use the rating propagation mechanism of the rating system to generate the estimated ratings for all his second-degree friends. Next, the simulation would calculate the direct rating that the user would give to all his second-degree friends using either the threshold algorithm or the agreement likelihood algorithm. The 'average difference error measure 'is defined to be the average difference between each pair of direct rating and estimated rating. The range of the error measure is from zero to one. An error of one would imply that the estimated rating completely differs from the direct rating whereas an error of zero would mean that they are both exactly equal.

4.2.1 Evaluation of strategies for Bayesian estimate rating system

For the first experiment, we would like to determine the effectiveness of each reconciliation strategy when the Bayesian estimate rating system is used. The average difference error measure is calculated for each strategy as we vary the density of the network by changing the number of users that each user rates. In this experiment, 40 different users are defined, and the number of other users that each user rates is within the range of 2 to 39. There are a total of 300 restaurants defined for this experiment, and each user would rate 100 restaurants, resulting in relatively accurate user-to-user direct ratings.

For the Bayesian estimate rating system, Figure 4-9 shows the performance of each reconciliation strategy when we vary the network density, in the case where the threshold algorithm is used to calculate the user-to-user direct ratings. For every network density value, 10 simulation runs are executed for each strategy and an average error measure for the strategy is calculated and plotted. From the plot in Figure 4-9, it can be seen that the performances of the 3 different strategies generally increases as network density increases. When the network density is low, the error measure is high because there are at most one or a few paths linking any two pairs of users together. Therefore, this limited number of paths did not allow any of the reconciliation strategies to calculate accurate estimated ratings.

From Figure 4-9, we can gather that the best performing strategy is the maximum weight strategy. Its error measure is roughly equivalent to that of the averaging strategy for sparse networks. However, when the network density increases, the maximum weight strategy is able to make use of the increased number of paths between pairs of users to produce the most accurate estimated ratings. The weighted averaging strategy is the worst performing one, however, its error measure converges with that of the averaging strategy for dense networks.

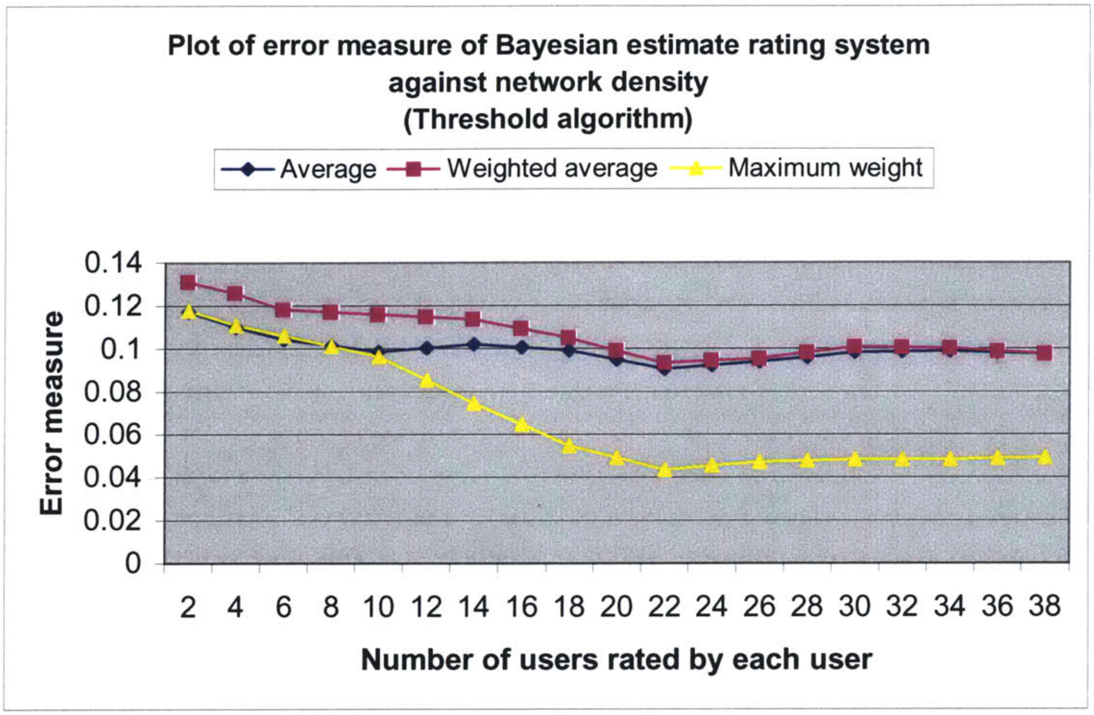

Figure 4-9: Plot of error measure of Bayesian estimate rating system against network density, using the Threshold algorithm to calculate ratings

For the Bayesian estimate rating system, Figure 4-10 shows the performance of each reconciliation strategy when we vary the network density, in the case where the agreement likelihood algorithm is used to calculate the user-to-user direct ratings. For every network density value, 10 simulation runs are executed for each strategy and an average error measure for the strategy is calculated and plotted. From the plot in Figure 4-10, it can be seen that the error measures of the 3 different strategies trends lower as the network density increases. When the network density is low, the error measure is high for all the strategies because there are very few paths linking any two pairs of users together. This limited number of paths did not allow any of the reconciliation strategies to calculate accurate estimated ratings.

The plot in Figure 4-10 shows that the maximum weight strategy is the best performing strategy across the entire range of network densities. When the network is sparse, its error measure is high, and is roughly equivalent to the error measures of the averaging and weighted averaging strategies. However, when the network density increases, the maximum weight strategy is able to make use of the increased number of paths between pairs of users to produce the most accurate estimated ratings. When the number of users rated by each user is 38, the error measure of the maximum weight strategy is only 20% that of the other two strategies.

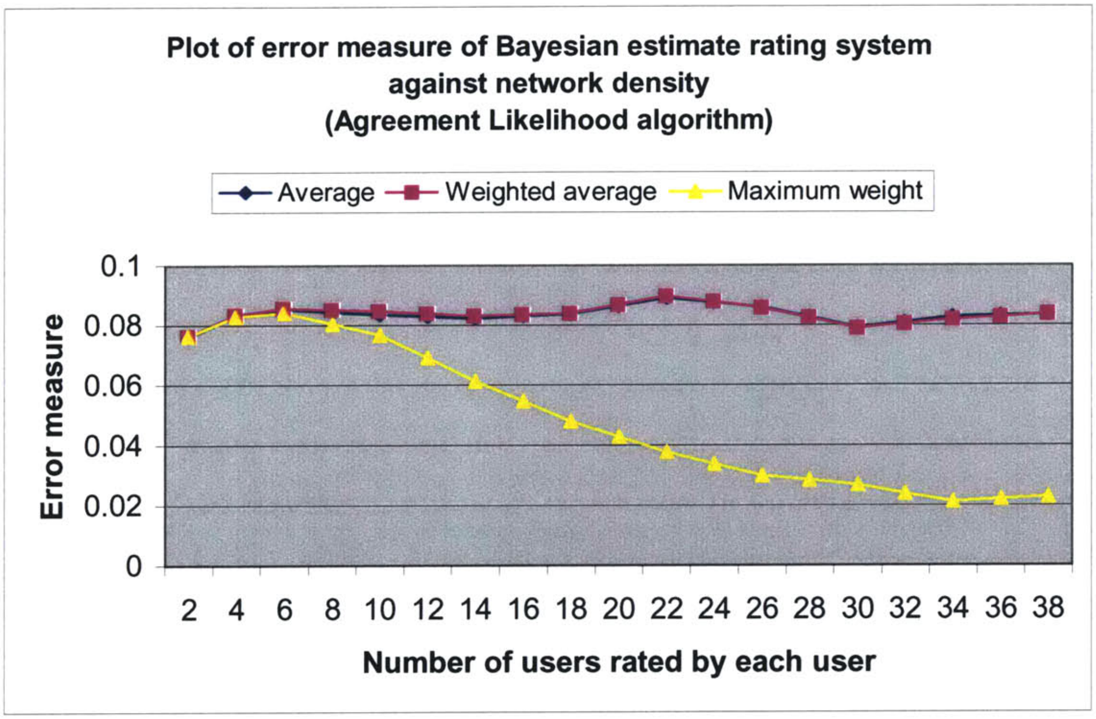

Figure 4-10: Plot of error measure of Bayesian estimate rating system against network density, using the Agreement Likelihood algorithm to calculate user-to-user direct ratings

The error measure of the averaging strategy fluctuates about a steady value of roughly 0.85 over the entire range of network densities. The weighted averaging strategy mirrors it almost exactly. These two strategies are not able to make use of the additional paths between pairs of users to produce more accurate estimated ratings.

4.2.2 Evaluation of strategies for Preference-based rating system

For the second experiment, we would like to determine the effectiveness of each reconciliation strategy when the Preference-based rating system is used. The average difference error measure is calculated for each strategy as we vary the density of the network by changing the number of users that each user rates. In this experiment, 40 different users are defined, and the number of other users that each user rates is within the range of 2 to 39. There are a total of 300 restaurants defined for this experiment, and each user would rate 100 restaurants, resulting in relatively accurate user-to-user direct ratings.

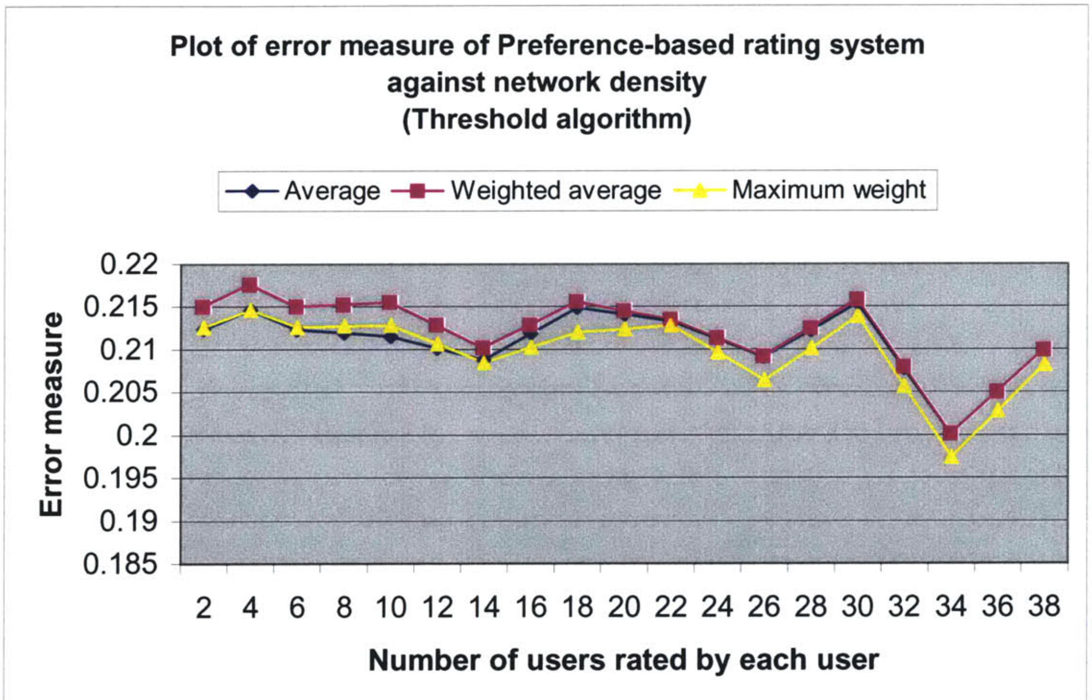

Figure 4-11: Plot of error measure of Preference-based rating system against network density, using the Threshold algorithm to calculate ratings

For the Preference-based rating system, Figure 4-11 shows the performance of each reconciliation strategy when we vary the network density, in the case where the threshold algorithm is used to calculate the user-to-user direct ratings. For every network density value, 10 simulation runs are executed for each strategy and an average error measure for the strategy is calculated and plotted. From the plot in Figure 4-11, it can be seen that the error measures of all the 3 strategies fluctuate within a very small range of values, from 0.20 to 0.22, as the network density varies. These error values are very high when compared to any strategy used for the Bayesian estimate rating system in Figure 4-9 where the threshold algorithm is also used. This provides evidence that the Preference-based rating system will not be a good rating system to used when user-to-user direct ratings are calculated using the threshold algorithm, regardless of which multiple path reconciliation strategy is the best for it.

From Figure 4-11, we can see that all the three reconciliation strategies produce almost identical errors, which are high and fluctuate about a value of about 0.21. They are therefore almost equal in performance when used with the Preference-based rating system and the threshold algorithm. However, since the error measures are so high, it would be better to use the Bayesian estimate rating system when the threshold algorithm is used to calculate user-to-user direct ratings.

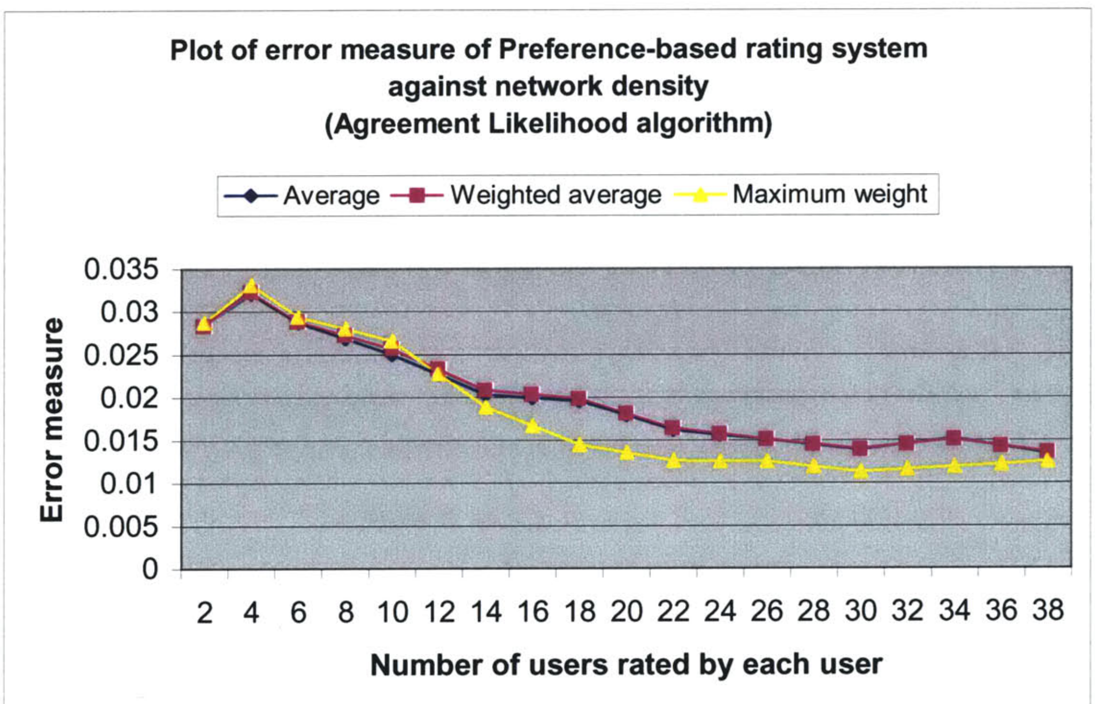

Figure 4-12: Plot of error measure of Preference-based rating system against network density, using the Agreement Likelihood algorithm to calculate user-to-user direct ratings

For the Preference-based rating system, Figure 4-12 shows the performance of each reconciliation strategy when we vary the network density, in the case where the agreement likelihood algorithm is used to calculate the user-to-user direct ratings. For every network density value, 10 simulation runs are executed for each strategy and an average error measure for the strategy is calculated and plotted. From the plot in Figure 4-12, it can be seen that the error measures of the 3 different strategies trends lower as the network density increases. When the network density is low, the error measure is high for all the strategies because there are very few paths linking any two pairs of users together. This limited number of paths did not allow any of the reconciliation strategies to calculate accurate estimated ratings.

The performances of the three different strategies are very similar throughout the entire range of varying network densities. The error measure of each of them decreases as the network density increases until the point where each user rates about half the total number of users in the network. Beyond that, the error stabilizes at a steady value. The maximum weight strategy has an error measure that stabilizes at a value of 0.012, which is slightly lower than that of the other two strategies, which stabilizes at about 0.015. Therefore, the maximum weight strategy has the best performance overall, bettering that of the other two strategies by 20% over half of the range of network densities. The averaging strategy and the weighted averaging strategy have error measures which are almost entirely identical over the range tested.

5. Conclusion and Future Work

For distributed systems in general, such as peer-to-peer networks or e-commerce systems, ratings play an increasingly important role in determining the reliability of information sources. In this thesis, the framework for the rating process has been formalized. In addition, the design of ratings management systems using the notion of personalized reputation was introduced and discussed in detail. With this concept of reputation, two different ratings management systems were designed and tested: the Bayesian estimate ratings system and the Preference-based ratings system. When compared against the Centrality-based ratings system that was designed using the global notion of reputation, these two systems have been proven to be superior when the error in ranking was used as a measure.

The relative performances of the Bayesian estimate ratings system and the Preference-based ratings system are very closely matched, and vary only slightly under different conditions. In the case where the rating of resources is binary, as represented by the threshold algorithm, the Bayesian estimate ratings system performs better than the Preference-based ratings system under varying conditions of network density and rating accuracy. This can be explained by the fact that the Bayesian estimate ratings system was designed with binary ratings as a basis. In the case where the rating of resources varies continuously between zero and one, as represented by the agreement likelihood algorithm, the Preference-based ratings system performs better than the Bayesian estimate system.

In this thesis, we have also shown that since ratings and reputations are clearly context-dependent properties, models about them should take the context into account. Any reputation value or rating should always be associated with a given context. Both the Bayesian estimate ratings system and the Preference-based ratings system make use of personalized and context-dependent ratings and reputations.

As the popularity of peer-to-peer networks increases, large communities of users would form. Each user in the network would only know a few other users. However, it is conceivable that he or she might be interested in determining the user who is the most reliable information source in the network for a given context of interest. This is where the propagation of ratings comes into play. In the case where there are multiple parallel paths between two users linked by intermediaries, there are different strategies that can be adopted to best estimate the rating. We have shown that a good strategy would be to use solely the path that contains the most trusted immediate intermediary.

Ratings management systems could be an important part of online businesses. Some of these businesses could compete in the same market by offering similar services or products. Accurate descriptions of these similar businesses, and their services and products, would allow consumers to locate them. The description, however, does not fully indicate the quality or suitability of the service or product provided by the business in relation to the needs of the consumer. Since reliability information about these businesses is hard to locate, consumers would often end up making uninformed or biased decisions in choosing between them. Provided that careful designs are undertaken to prevent or reduce the damage caused by malicious attacks, either of the two personalized ratings management systems proposed in this thesis could be used for businesses and their services and products.

There are many issues that have yet to be resolved. The framework for managing reputation and ratings discussed in this thesis deals with a single context. An inference mechanism that is capable of inferring ratings and reputations from one context to another must be developed. One way of doing that would be to employ an ontology-based classification to determine the degree to which different contexts are related. Using this approach however, would require the solution to a difficult problem: to resolve the different ontological views of the world held by different agents. The exact metric or function that should be used to transfer a rating from one context to another is yet to be worked out.

In testing the scalability of the rating systems, the percentage of the total number of users in a community rated by each subject user is chosen to be 50%. A network connectivity of 50% is reasonable for a small community of users, but unrealistic for a large network like the Internet. Therefore, further work needs to be done to determine the scalability of the rating systems for large networks with lower connectivity.

Another issue is that the current framework for reputation management requires that agents reveal their personal ratings of other agents and resources. This could create privacy concerns as personal attributes such as preferences and biases may be contained in this revealed information. To resolve this issue, circles of trust could be defined and agents could choose to reveal the information only to those other agents that meet certain well-defined criteria.

Appendix A

Derivation for the ratings propagation function, $\rho_{ik}(c_p) = f_2(\rho_{ij}(c_p), \rho_{jk}(c_p))$, of the Preference-based rating system is given below.

$$\rho_{ij}(c_p) = \rho_i(c_p)\rho_j(c_p) + (1-\rho_i(c_p))(1-\rho_j(c_p)) \tag{1}$$
$$\rho_{jk}(c_p) = \rho_j(c_p)\rho_k(c_p) + (1-\rho_j(c_p))(1-\rho_k(c_p)) \tag{2}$$
$$\rho_{ik}(c_p) = \rho_i(c_p)\rho_k(c_p) + (1-\rho_i(c_p))(1-\rho_k(c_p)) \tag{3}$$

From equation (1),

$$\begin{aligned} \rho_{ij} &= \rho_i\rho_j + 1 + \rho_i\rho_j - \rho_i - \rho_j \\ &= \rho_j(2\rho_i - 1) - \rho_i + 1 \end{aligned} \tag{4}$$
$$\Rightarrow \quad \rho_j = \frac{\rho_{ij} + \rho_i - 1}{2\rho_i - 1}$$

From equation (2),

$$\begin{aligned} \rho_{jk} &= \rho_j\rho_k + 1 + \rho_j\rho_k - \rho_j - \rho_k \\ &= \rho_j(2\rho_k - 1) - \rho_k + 1 \end{aligned} \tag{5}$$
$$\Rightarrow \quad \rho_j = \frac{\rho_{jk} + \rho_k - 1}{2\rho_k - 1}$$

Combining equation (4) and equation (5),

$$\begin{aligned} \frac{\rho_{ij} + \rho_i - 1}{2\rho_i - 1} &= \frac{\rho_{jk} + \rho_k - 1}{2\rho_k - 1} \\ (\rho_{ij} + \rho_i - 1)(2\rho_k - 1) &= (\rho_{jk} + \rho_k - 1)(2\rho_i - 1) \\ 2\rho_k(\rho_{ij} + \rho_i - 1) - (\rho_{ij} + \rho_i - 1) &= (\rho_{jk} - 1)(2\rho_i - 1) + \rho_k(2\rho_i - 1) \\ \rho_k(2\rho_{ij} + 2\rho_i - 2 - 2\rho_i + 1) &= (\rho_{jk} - 1)(2\rho_i - 1) + (\rho_{ij} + \rho_i - 1) \\ \rho_k(2\rho_{ij} - 1) &= 2\rho_i\rho_{jk} - 2\rho_i - \rho_{jk} + 1 + \rho_{ij} + \rho_i - 1 \end{aligned} \tag{6}$$

$$\Rightarrow \quad \rho_k = \frac{2\rho_i\rho_{jk} - \rho_i - \rho_{jk} + \rho_{ij}}{2\rho_{ij} - 1}$$

Substituting equation (6) into equation (3),

$$\begin{aligned}
\rho_{ik} &= 2\rho_i\rho_k - \rho_i - \rho_k + 1 \\
&= \rho_k(2\rho_i - 1) - \rho_i + 1 \\
&= \frac{(2\rho_i - 1)(2\rho_i\rho_{jk} - \rho_i - \rho_{jk} + \rho_{ij}) + (1 - \rho_i)(2\rho_{ij} - 1)}{2\rho_{ij} - 1}
\end{aligned}$$

Appendix B

Derivation for the posterior estimate of the proportion of approvals in n encounters between member i and member j is given below, as found in [22]:

$$p(\hat{\theta} \mid D) = \frac{L(D \mid \hat{\theta}) p(\hat{\theta})}{\int_{\hat{\theta}} L(D \mid \hat{\theta}) p(\hat{\theta}) d\hat{\theta}}$$

$$= \frac{\hat{\theta}^{p}(1-\hat{\theta})^{n-p} \dfrac{\hat{\theta}^{c_1-1}(1-\hat{\theta})^{c_2-1}}{\int \hat{\theta}'^{c_1-1}(1-\hat{\theta}')^{c_2-1} d\hat{\theta}'}}{\int \hat{\theta}''^{p}(1-\hat{\theta}'')^{n-p} \dfrac{\hat{\theta}''^{c_1-1}(1-\hat{\theta}'')^{c_2-1}}{\int \hat{\theta}'''^{c_1-1}(1-\hat{\theta}''')^{c_2-1} d\hat{\theta}'''} d\hat{\theta}''}$$

$$= \frac{\hat{\theta}^{p+c_1-1}(1-\hat{\theta})^{n+c_1-p-1}}{\int \hat{\theta}''^{p+c_1-1}(1-\hat{\theta}'')^{n+c_1-p-1} d\hat{\theta}''}$$

$$= \frac{\Gamma(c_1 + c_2 + n)}{\Gamma(c_1 + p)\Gamma(n - p + c_2)} \hat{\theta}^{p+c_1-1}(1-\hat{\theta})^{n+c_1-p-1}$$

$$= Beta(c_1 + p, c_2 + n - p)$$

References

[1] Chrysanthos Dellarocas. *"Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior."* in Proceedings of the 2nd ACM Conference on Electronic Commerce, 2000.

[2] Phillip Bonacich. *"Power and Centrality: A Family of Measures."* American Journal of Sociology 92, 1170-1182, 1987

[3] Stanley Wasserman, Katherine Faust. Social Network Analysis: Methods and Applications. Cambridge, United Kingdom: Cambridge University Press, 1995.

[4] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, J. Riedl. *"GroupLens: Applying Collaborative Filtering to Usenet News."* Communications of the ACM, 40(3), pp. 77-87, 1997.

[5] Lik Mui, Cheewee Ang, Ari Halberstadt. *"Collaborative Sanctioning: Integrating Reliability and Reputation Measures into Distributed Systems."* in Special Session on Learning, Cooperation and Coordination in Multi-Agent Systems, SCI'2001.

[6] G. Salton. The SMART Information Retrieval System: Experiments in Automatic Document Processing. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[7] Marko Balabanovic, Yoav Shoham. "*Learning Information Retrieval Agents: Experiments with Automated Web Browsing*." in AAAI Spring Symposium on Information Gathering, 1995.

[8] A. Pretschner, S. Gauch. "*Ontology based personalized search*." in Proceedings 11th IEEE International Conference on Tools with Artificial Intelligence, pp. 391-398, 1999.

[9] D. Goldberg, D. Nichols, B. M. Oki, D. Terry. *"Using collaborative filtering to weave an information tapestry."* Communications of the ACM, 35(12), pp. 61-70, 1992.

[10] eBay. http://www.ebay.com/

[11] G. Zacharia, A. Moukas, P. Maes. "*Collaborative reputation mechanisms in electronic marketplaces*." in Proceedings of the HICSS-99 Conference, Electronic Commerce Minitrack (Maui, Hawaii, Jan. 5--9). IEEE Computer Society, 1999.

[12] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl. *"GroupLens: An Open Architecture for Collaborative Filtering of Netnews."* in Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work, Chapel Hill, NC: Pages 175-186, 1994.

[13] Leonard N. Foner. *"Yenta: A multi-agent, referral-based matchmaking system."* in W. Lewis Johnson and Barbara Hayes-Roth, editors, Proceedings of the 1st International Conference on Autonomous Agents, pages 301--307, New York, February5-- 8 1997. ACM Press.

[14] Rohit Khare and Adam Rifkin. *"Weaving a Web of Trust."* in World Wide Web Journal special issue on Security, Volume 2, Number 3, pages 77-112, 1997.

[15] Recreational Software Advisory Council. http://www.rsac.org/

[16] A. Glass, B. Grosz. *"Socially Conscious Decision-Making."* Autonomous Agents'2000.

[17] B. Yu, M. P. Singh. *"A Social Mechanism of Reputation Management in Electronic Communities."* CIA'2000.

[18] J. Rouchier, M. O'Connor, F. Bousquet. *"The Creation of a Reputation in an Artificial Society Organized by a Gift System."* Journal of Artificial Societies and Social Simulations, 4(2), 2001.

[19] MovieLens. http://movielens.umn.edu/

[20] HelloBrain.com. http://www.hellobrain.com/

[21] Phillip Bonacich, Paulette Lloyd. *"Eigenvector-like Measures of Centrality for Asymmetric Relations."* Social Networks. In press.

[22] Lik Mui, Cheewee Ang, Mojdeh Mohtamshemi, Peter Szolovits. *"A Probabilistic Model for Collaborative Sanctioning."* MIT Technical Memorandum MIT-LCS-TM-617, May 2001.

4493-98