Feature Selection for Collaborative Team Formation via Social Network Analysis

Michelle Cheatham
Air Force Research Lab
Information Directorate
Wright-Patterson AFB, OH 45433
Email: michelle.cheatham@wpafb.af.mil
Phone/Fax: (937) 904-9011/255-4511

Felicia Harlow
Air Force Research Lab
Information Directorate
Wright-Patterson AFB, OH 45433
Email: felicia.harlow@wpafb.af.mil
Phone/Fax: (937) 904-9173/255-4511

Kevin Cleereman
Air Force Research Lab
Information Directorate
Wright-Patterson AFB, OH 45433
Email: kevin.cleereman@wpafb.af.mil
Phone/Fax: (937) 904-9090/255-4511

***Abstract*—The goal of expert recommendation (ER) systems is to suggest individuals whose expertise can assist in solving a given problem. As organizations become more geographically distributed and virtual teams become more commonplace, the requirements put on ER systems will advance from that of merely suggesting an individual to enhance a pre-existing team to that of suggesting a full team from scratch. We address the critical question of what factors these advanced ER systems will need to consider when making recommendations for team membership. We provide a survey of the features proposed in contemporary research and discuss how they apply to this new version of the ER problem. We also conduct an analysis of these characteristics using a real-world data set to determine which of these features are relevant.**

***Index Terms*—expert team recommendation, social network analysis, feature selection, collaborative filtering, statistical methods**

I. Introduction

The first step in collaboration is to determine with whom to collaborate. However, today's large, fluid, and geographically distributed organizations make this difficult, and this complexity is often ignored by current collaboration software. Managers therefore frequently resort to an ad-hoc approach when creating a new team, pulling together whichever employees happen to be available at the time or using word-of-mouth recommendations to find potential team members. These techniques will frequently produce suboptimal teams.

Several methods have been proposed to ensure that an organization is taking advantage of its human capital. One technique is the use of "yellow page" systems, in which employees enter information about their skills which can then be searched by others. There are several problems with this approach. For instance, there is the issue of ensuring the accuracy of employee profiles. This means both verifying that the information given by employees is reasonable (they are not too modest or boastful in reporting their skills) and timely (the profiles must be periodically updated as skill sets evolve over time) [1]. Another concern is that a smooth-running team is not merely a function of the team members knowledge; there are myriad social factors that impact team performance as well [2].

McDonald and his team have suggested other ways to measure an employee's skill which they have shown to be reasonably accurate. These techniques include actually quizzing employees on a variety of topics or asking employees to rank each others' knowledge [3]. However, both of these methods are time-consuming. Recent developments in the field of expertise-mining have the potential to provide an automated solution to the problem. Applications parse work products created by employees as part of their day-to-day activities (web pages, reports, emails, etc.) and use automated text-mining algorithms to attempt to determine the skills of the employees who authored the documents. Examples of such systems are Agentware Knowledge Server (document based) and MEMOIR (web page and keyword based) discussed in [4], which also describes key features of several other systems. The accuracy of such systems is still a matter of research. Nevertheless, even when these systems are mature enough to provide a high level of accuracy, the impact of social factors on team performance still needs to be considered.

Some expertise recommendation systems do attempt to take social factors into account. Examples of such systems are described in the next section. However, these systems (and all other ER systems we could find in a literature search) are limited in the sense that they recommend experts solely for a one-to-one collaboration – they do not consider the interplay between members of a *group*. We are currently developing a social network-based ER system that will consider these interactions. There are many more factors that can be considered when analyzing group interactions rather than one-on-one relationships. The goal of this paper is to explore the utility of both single-link and group-based features for collaborative team formation.

The remainder of this paper is organized as follows: Section 2 explains the concept of social network analysis and describes current related work with a particular focus on ER applications, Section 3 describes the features latent in a social network representation and discusses their possibile utility for team recommendation, in Section 4 we describe our test data and methodology, we present and analyze our results in Section 5 and conclude the paper with a summary and ideas for future work in Section 6.

II. Social Networks for Expert Recommendation

A social network can be represented by a graph in which the nodes represent actors (typically people) and the edges represent relations between them. These relations can be communication, disease transmission, appearing in the same movie, or any number of other possibilities. The edges may be weighted or unweighted, directed or undirected. One commonly studied social network is co-authorship of scientific publications (because the data is readily available from sources such as Citeseer). Figures 1 and 2 show an example of such a network, based on publications of the Collaborative Technologies branch within the Air Force Research Laboratory between 2003 and 2005. In Figure 1, the nodes of the graph are authors, and the lines between them indicate that the connected authors have collaborated on a paper. The thickness of the line indicates the number of papers they have coauthored. Newman suggests that affiliation networks (networks in which nodes are related by membership in a common group) are fundamentally bipartite graphs with one type of node representing individuals and the other representing the groups they belong to such that no edge exists between like types of vertices [5]. In Figure 2, the same dataset is shown as a bipartite graph where one class is the authors (represented as squares) and the other is the papers (represented as circles). This view still allows us to see the different workgroups that exist and makes it is easier to see what those groups are working on.

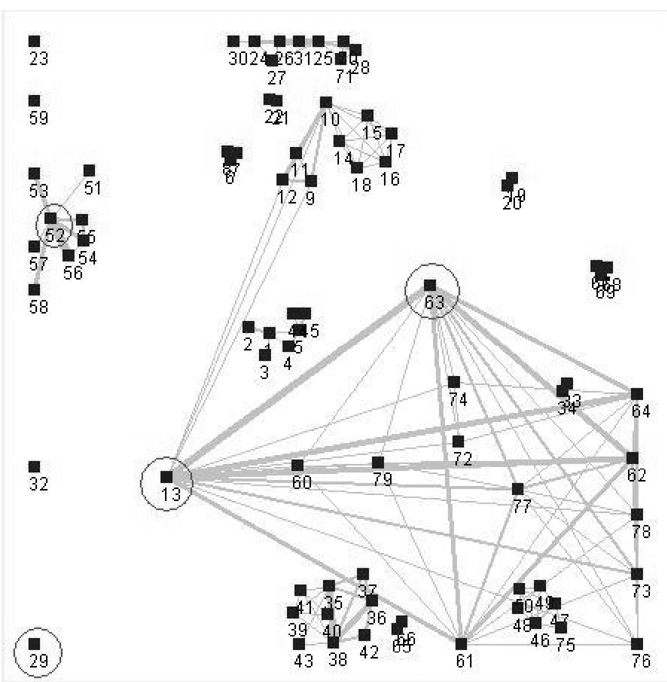

Fig. 1: Traditional View

Social network analysis has been used to study many other types of social interactions in which the underlying data is inherently relational. The most frequently studied interactions are based on email and IM traffic [1], [6], [7], coauthorship of publications [8], and the proximity of names on web pages [9].

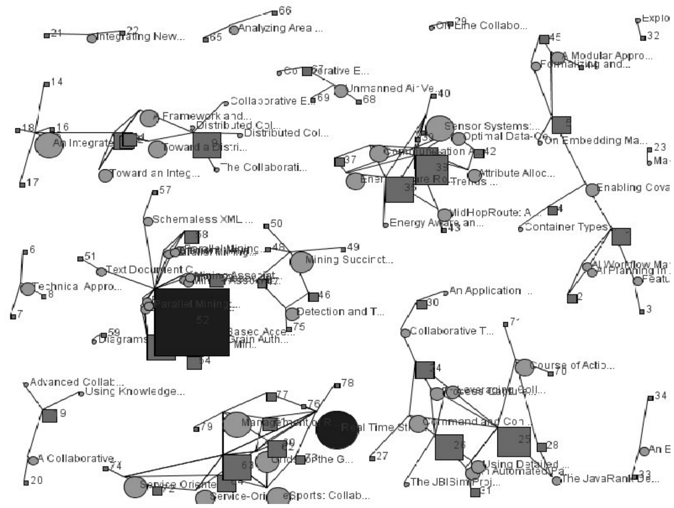

Fig. 2: Bipartite View

Most of this work examines the properties of these social networks and tries to interpret them in terms of the underlying community being studied. For example, [10] points out that the greater the distance (e.g. average path length) between nodes in a social network, the longer it will take new information to diffuse throughout the network. [11] relates an individual's centrality in a social network to his ability to coordinate a project.

This basic research has led to applications that allow users to visualize and explore their social network [12], [10]. Mori et al take this a step further and provide the ability to create dissemination rules based on their social networks [13]. An important contribution of this work is to determine the level of user agreement with the social networks created though different techniques. For instance, ContactMap mines a user's email to create a representation of his social network, but the system was deemed by users to extract many more contacts than were actually relevant in their social network [12]. McDonald's group creates their social network by asking employees to perform a successive pile sort of index cards with employee names on them into groups according to "who hangs out together" and aggregating the results. When employees were asked to evaluate their egocentric network (a type of social network that is centered around one person and containing only his or her links to other people) within the overall network, they indicated that the results contained many inaccuracies [14]. While not the subject of this paper, these results indicate the strong need for more research into the creation of accurate social networks in order to build more effective applications, including ER systems, around them.

There are several ER systems based on social networks. ReferralWeb, done by Kautz's group at AT&T Laboratories, is an interesting application in this area. It mines publicly available documents on the internet and constructs a graph of names that appear in close proximity [9]. The resulting system can then answer questions such as "What is my relationship with Person A?” and “What people in my neighborhood know about topic x?” Another example is McDonald’s Expert Recommender system, which uses a two-step approach to produce its recommendations. In the first phase, the system finds the set of individuals who have some knowledge relevant to the current query. This set is then further reduced by considering the person who placed the query with the goal of recommending someone who both knows the answer and is a close colleague of the user [14].

The above systems only recommend one expert at a time for collaboration, which is significantly easier than recommending an entire team. When there are only two people involved (the requester and the recommended individual), there is only one social interaction that needs to be considered. When the goal is to recommend a team of n experts to tackle a problem, there are $\frac{n(n-1)}{2}$ relationships involved. Both the computational complexity and the types of factors that need to be considered are different. For instance, when creating a new group it may be advisable to avoid choosing more than one person who has a high degree in the social network (and is therefore likely to be a strong leader). This is not an issue when only recommending an individual.

To our knowledge, the system we are developing is the only ER system specifically designed to recommend teams. In most cases, people choose others to work with based on both competence in the subjects needed for the job and likeability [15]. We have developed a data representation that allows us to reason about both of these aspects of collaboration. Our system will use text-mining of work products (reports, presentations, email, etc.) to build a social network, with additional concept information embedded as shown in Figure 3. The circles represent papers, projects, organizational membership, or any other group that people may be a part of. Papers are connected to their authors (squares) and their primary concepts (rounded squares). In this way the network structure depicts who is working with whom and on what topics. While a user may utilize the network directly to view their own ego-centric network, or to answer such questions as “Which of the people I have collaborated with in the past know about sensors?” and “Who in the organization is working on the same topics I am but has yet to directly collaborate with me?”, we plan to use it as the basis for a more advanced ER system. The basic use case is that a manager has a project description, and he needs to form a new team to work on this project. The manager will supply the project description to the application, which will use standard text-mining techniques to extract the key words related to the skills/knowledge that will be required. All potential team members will then be filtered with respect to these key words to arrive at the set of all employees that have experience with at least one of the required areas. This will reduce the number of people that need to be considered in the more computationally-intensive next stage, which will use information from the social network (such as how connected the employee is, how similar they are to one another, etc.) to recommend a project team.

Determining which features from the social network will be used in the second stage is the goal of this paper. The manager will be provided a list of features that can be considered and will weight them according to the needs of the current situation. However, it is not realistic to expect every manager to be well-versed in organizational psychology and team theory. Instead, we would like to be able to provide a set of default weights as a starting point. The work presented here identifies a set of likely features which will be analyzed further as part of our future work in order to find reasonable values for the default weights.

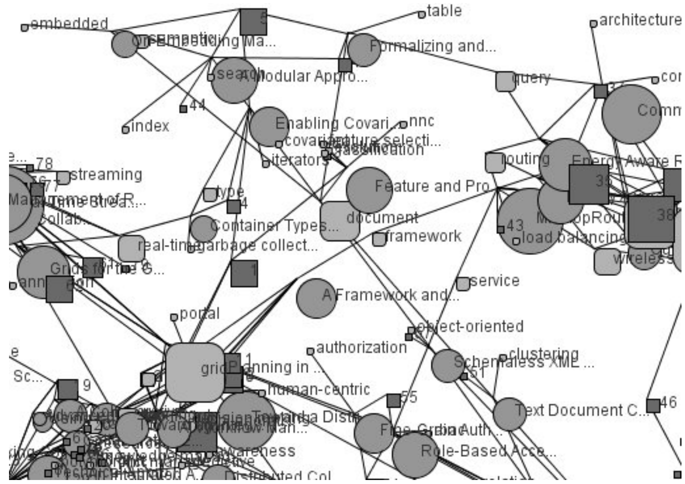

Fig. 3: Enhanced View

III. Features

In this section we outline the features available from our social network representation that may be useful for team recommendation. For each feature we provide a formula or definition, an interpretation in the team recommendation context, and a brief examination of other research related to the metric.

A. Centrality

Centrality is an intuitive measure of a person’s importance within a social network. If a person is at the center of things, he is likely to be well-informed and have the resources and connections necessary to get things done. Research has shown that central nodes are most often identified as the leader of a group [11]. There are numerous ways to measure a node’s centrality, and because our social network representation considers both a project’s actors and concepts, each of these measures can be interpreted in multiple ways.

Degree/Reach: The degree of a node is simply the number of incident edges, represented as $d(V_i)$ for a vertex V. It is the easiest centrality measure to compute because it depends solely on information local to the node in question; however, this may also limit its utility. [11] points out that a node’s degree only corresponds to local authority because it does not take into account indirect ties to other nodes. [16] uses two-degree reachability in an attempt to provide similar information to degree while considering more of the network. They calculate the percentage of the network within two links from the current node.

Given that $G = (V_R, V_P, E)$ represents a bipartite graph of a social network of researcher nodes, V_R, and project nodes, V_P, let U_R be the set of all V_R reachable within 2 hops from V_{Ri}. Then the two degree reachability of V_{Ri} is defined as

$$d_2(V_{Ri}) = \frac{|U_R|}{|V_R|}$$

Both degree and reach are meant to give an indication of the resources or information at a node's disposal. Because our representation is a bipartite graph in which researchers and concepts are only connected via projects (and not directly to each other), these measures can be calculated and interpreted in various ways. The standard degree in our representation is simply the number of projects a researcher has worked on. A node's set of researchers reachable by two or fewer links is more similar to the standard degree measure in traditional social networks. It is the set of people with whom the researcher has collaborated directly with. Similarly, the set of concepts in the 2-reachable set of the researcher are those concepts that the individual has direct experience with. Newman points out in [17] that the more people working on a project, the less likely it is for group members to get to know one another well. In order to consider this, we also calculated the normalized person degree of a researcher, V_R, using the formula:

$$d_N(V_R) = \sum_{j=1}^{m} \frac{n_j - 1}{n_j}$$

where m = the number of projects on which V_R has worked and n_j = the number of researchers working on project j.

In the context of an ER system, a person's degree measures can be looked at from several different angles. Intuitively, a person's project degree may be an important feature because a person who has previously worked on many different projects may be a more attractive choice for team membership than someone without much prior project experience. Person degree is one way to get an idea if an individual is team-oriented or prefers to work alone. Concept degree indicates whether a person has a broad skill set or is focused on a particular topic. In addition to these general degree measures, we also consider corresponding features relative to a particular project. For instance, project specific person degree is the number of times an individual has previously worked *with the other members of a project team*, and project specific concept degree is the amount of experience a person has worked with the concepts *relevant to a particular project*.

Betweenness: Betweenness is the number of shortest paths between pairs of actors that pass through a node. Theoretically, nodes with high betweenness control the flow of information and resources within the social network [17]. Given that $g_i V_j =$ the number of shortest paths between vertices i and j that pass through vertex, V, and $g_{ij} =$ the number of shortest paths between vertices i and j. The betweenness of the vertex is then

$$C_B(V) = \sum_i \sum_j \frac{g_i V_j}{g_{ij}}$$

We calculate betweenness over the enhanced social network (similar to the one shown in Figure 3). Two betweenness-based features are considered. Person betweenness is the sum of the betweenness values for all collaborators on a project. It indicates how central (and therefore more likely to be highly regarded as experts or leaders) the team members are. Concept betweenness is the sum of the betweenness values of all of the concepts that team members have experience with. This feature reflects how critical the team members' skill sets are, e.g. how much experience they have with topics that "bridge" other concepts or projects together.

B. Similarity

Similarity is a measure of how much two nodes have in common. It is well-established that people who are similar tend to associate with one another often, a concept known as *homophily* [6]. This results in similar individuals being closer together in a standard social network representation, which links people who have worked together. In our representation you can see both the cause and effect of homophily. The amount of overlap between two nodes concepts gives an idea of how similar their backgrounds and experience are, while the overlap between the nodes collaborators illustrates how often this common background has led to direct cooperation on projects.

Highly similar team members can have both a positive and negative effect on group performance. People with similar backgrounds who have worked together previously spend less time in the forming and norming stages of team development. They often have an easier time communicating because they have a common vocabulary to draw from. The downside is that groupthink is a danger in such groups. A team made up of people with diverse backgrounds and skills can bring many different perspectives to bear on a problem and therefore has the potential to develop very innovative solutions [15]. It is intuitive that similar groups perform well on routine tasks and tasks in which quick decision making is required (such as command and control), while ill-defined problems requiring creative solutions are best handled by diverse groups. Of the set of features considered in this study, similarity is the one most likely to depend on the particular problem the group is being formed to address.

There are multiple ways to measure similarity, which essentially correspond to different ways to quantify the overlap between the ego-centric networks of two nodes.

Symmetric Similarity: The symmetric similarity of two nodes A and B is simply the intersection of their direct neighbor sets [18]. In its most basic form (which we call project similarity), this implies that people who have worked directly together are good candidates to work together again because they do not need to spend time getting to know one another. If we are considering the set of neighboring individuals (person similarity), this corresponds to the idea that two people are more likely to work together if they have collaborated with at least one common person in the past. If the concept neighborhood is used in the calculation (concept similarity), the measure implies that two people are more likely to cooperate on a project if they have a background in the same subjects. The similarity of two researchers, V_{R1} and V_{R2}, who have worked on project sets P_1 and P_2, respectively, is the cardinality of their project intersections divided by the cardinality of the union of their projects sets, represented as

$$S_{sim}(V_{R1}, V_{R2}) = \frac{|P_1 \cap P_2|}{|P_1 \cup P_2|}$$

Relative Similarity: Relative similarity has the same interpretations as symmetric similarity, but it provides a more sensitive filter for querying node neighbors [18]. The underlying idea of this feature is that if researcher $R2$'s entire neighborhood (of either collaborators or concepts) is a subset of researcher $R1$'s neighborhood, then $R2$ is more similar to $R1$ than $R1$ is to $R2$, and this measure takes that asymmetry into account. We represent this formally as

$$R_{sim}(V_{R1}, V_{R2}) = \frac{|P_1 \cap P_2|}{|P_1|}$$

and similarly

$$R_{sim}(V_{R2}, V_{R1}) = \frac{|P_1 \cap P_2|}{|P_2|}$$

IV. Experimental Design

One of the challenges in the research and development of ER systems is finding an appropriate test strategy. One possibility is to compare the system's recommendations to those of a human. However, it is difficult to do large tests in this way because the number of volunteers (and the time they have available) is often limited. In addition, the human would need to have knowledge of the skills and relationships of all the employees in the system. Also, comparing two different team recommendations is difficult. Team 1 may consist of different members than Team 2, and yet both may be equally effective. Another option is to use existing data on the performance of past teams. This has its own set of challenges. Data on past projects is often not available or organizations are reluctant to release it due to privacy concerns. Even when data is available, it is difficult to come up with an appropriate metric for team performance.

We have elected to go with the second option and use project statistics collected by SourceForge for our analysis. This data is publicly available by submitting a request at SourceForge.net. A variety of information is available for each project. We are using the project title, list of project members' usernames, and topic list (keywords) to create our social network. We will use the project ranking as our measure of team performance. The ranking is the sum of metrics based on product popularity (traffic), development activity, and communication (bug fixes and forum posts). The formulae for these are provided in Table I. While project ranking is not sufficient to say that one piece of software is better than another, we believe it is a reasonable measure of team performance.

TABLE I: SourceForge Project Rankings

traffic =	$(log_{maxD}(D) + log_{maxL}(L) + log_{maxS}(S))/3$
D = downloads =	prior 7 days download total + 1
maxD = maxDownloads =	highest all-project download total + 1
L = logo =	prior 7 days logo hit total + 1
maxL = maxLogo =	highest all-project logo hit total + 1
S = site =	prior 7 days site hit total + 1
maxS = maxSite =	highest all-project site hit total + 1
development =	$(log_{maxC}(C) + ((100 - fA)/100) + ((100 - aL)/100))/3$
C = commits =	prior 7 days CVS commit total + 1
maxC = maxCommits =	highest all-project commit total + 1
fA = fileAge =	min(age of latest file release in days, 100)
aL = adminLogin =	min(days since last project administrator login, 100)
communication =	$(log_{maxT}(T) + log_{maxML}(ML) + log_{maxF}(F))/3$
T = tracker =	prior 7 days tracker submission count + 1
maxT = maxTracker =	highest all-project tracker submission count + 1
ML = MailingList =	prior 7 days ML post count + 1
maxML =	highest all-project ML post count + 1
F = forum =	prior 7 days Forum post count + 1
maxF = maxforum =	highest all-project Forum post count + 1

We initially looked at 2395 SourceForge projects pertaining to 224 topic areas. The high ratio of people to projects revealed a high percentage of members who had not worked with at least one other researcher more than once. This limited contact would yield little in the way of researcher/team success measures. For this reason, we excluded all members who had no repeated collaboration experience, retaining all who had worked at least twice with another researcher.

The scaled-down SourceForge statistics covered 230 software projects on which 426 researchers worked covering 161 concepts. The average number of projects an individual worked on was 1.141, and each project had an average of 18.805 researchers participating. Each individual worked directly with an average of 6.135 other researchers. The reduced dataset was still very sparse, which may impact several of the features under consideration. Our future work will include analyzing an independent data set, possibly co-authorship of scientific publications from citeseer, in order to determine the extent of this impact.

Our goal is to determine which factors discussed in the previous section are predictive of project ranking. We will also determine which factors are highly correlated to one another so that we may find the smallest feature set that is needed to choose an effective team. For instance, if the degree and betweenness of a person are both strongly related to team performance but are also very highly correlated with each other, they are essentially providing the same information and we need only consider one of them for use by an ER system. The ER system we are proposing would provide users with the ability to determine how much each feature is weighted when finding candidate teams. Our future work will take the features that were shown by this study to be useful and employ a neural network to find appropriate default values for these weights.

V. Analysis

Table II shows the list of 15 features that were considered in this analysis. Each feature was computed for each individual on a project and then summed to arrive at a single score for the project. Table III shows the correlation coefficients between the features and between each feature and the SourceForge ranking described in the previous section.

Looking at the correlations between features, several observations can be made. For example, all of the standard degree measures (project, person, normalized person, and concept) are all highly correlated. It is not surprising that person degree and normalized person degree are related considering they are based on the same data. That the project and concept degrees are also highly correlated with person degree implies that the most well-connected people are the ones working on the most popular projects and with the most used skills. The project specific degree measures are very different from the general degree features. They are not very correlated to any other features, including one another.

Another interesting thing to note is that person betweenness is not correlated to any type of degree, even person degree. This suggests that the people in the network that provide critical links between others (and who information is most likely to flow through) are not the most connected and active individuals within SourceForge. Conversely, concept betweenness *is* highly correlated to the standard degree metrics, implying that the people working with skills that frequently bridge gaps or connect other skills and projects are also the people working on many projects with many other people.

The similarity metrics, both symmetric and relative, are all very correlated to one another. This is not surprising since it seems obvious that people with similar skill sets and similar acquaintances are more likely to work on the same projects. In addition, the similarity metrics are all highly related to the standard degree features. Interestingly, none of them are in the top half of features most closely correlated to the project ranking, implying that homogeneous teams may not be the most successful for distributed software development projects.

When looking at the utility of these features for predicting project ranking, it is important to keep in mind that correlations of features in social science data such as this are often much lower than is typical in engineering or other technical domains. For example, when looking at how personality traits are related to the major students choose in college, correlation coefficients of .2 to .3 are considered strong [19]. With this in mind, the features with correlation coefficients in the .1 and above range can be considered related to project ranking.

TABLE II: Features

Degree	
PrD	: Project Degree
PeD	: Person Degree
NPeD	: Normalized Person Degree
CD	: Concept Degree
PrPeD	: Project Specific Person Degree
PrNPeD	: Project Specific Normalized Person Degree
PrCD	: Project Specific Concept Degree
Similarity	
SPeS	: Symmetric Person Similarity
SCS	: Symmetric Concept Similarity
SPrS	: Symmetric Project Similarity
RPeS	: Relative Person Similarity
RCS	: Relative Concept Similarity
RPrS	: Relative Project Similarity
Betweenness	
PeB	: Person Betweenness
CB	: Concept Betweenness

These include: project degree, concept degree, project specific person degree, project specific normalized person degree, person betweenness, and concept betweenness.

Project degree and concept degree are positively correlated with project ranking, indicating that researchers with the most experience (i.e. who have worked on many projects using many skills) increase a team's chances of a high ranking. Similarly, the higher a team's aggregate person and concept betweenness, the higher the ranking is likely to be. Individuals with high person betweenness are exposed to a large amount of information, while it is possible that a high concept betweenness makes a person more likely to see how different skills fit together. More organization behavior research would be required to investigate this hypothesis. Project specific person degree (and normalized project specific person degree) is negatively correlated to project ranking. This implies that a team containing many members who have worked together previously is less highly ranked. It is difficult to pinpoint exactly why this may be true, but possibilities include teams spending more time socializing or the presence of groupthink.

Of the features that are correlated to project ranking, project degree, concept degree, and concept betweenness are also highly correlated to one another. The same is true (to a lesser extent) for project specific person degree and normalized project specific person degree. If an ER system were to use these features to recommend new teams, it may only be necessary to use one feature from each of these groups in addition to the person betweenness, which was not correlated to the other features.

VI. Conclusions

We have described an enhanced social network representation that can be used as the basis for a team recommendation system. We then examined the features that can be gleaned from such a representation and described their meaning within the ER context. Finally, we computed each feature based on a real-world data set and examined their correlation between one another and with the project ranking.

TABLE III: Feature Correlation

	Degree				*Project Specific Degree*			*Betweenness*		*Symmetric Similarity*			*Relative Similarity*			—
	PrD	PeD	NPeD	CD	PrPeD	PrNPeD	PrCD	PeB	CB	SPrS	SPeS	SCS	RPrS	RPeS	RCS	Rank
PrD	1.000	0.912	0.964	0.888	0.189	-0.389	0.127	0.600	0.830	0.867	0.841	0.826	0.873	0.861	0.854	0.113
PeD	0.912	1.000	0.966	0.845	0.122	-0.393	0.122	0.429	0.810	0.971	0.960	0.959	0.971	0.969	0.971	0.076
NPeD	0.964	0.966	1.000	0.885	0.205	-0.398	0.139	0.444	0.837	0.937	0.930	0.920	0.942	0.942	0.938	0.093
CD	0.888	0.845	0.885	1.000	0.138	-0.357	0.053	0.494	0.885	0.822	0.787	0.754	0.829	0.810	0.790	0.162
PrPeD	0.189	0.122	0.205	0.138	1.000	0.684	0.043	-0.025	0.101	0.090	0.119	0.122	0.094	0.113	0.118	-0.143
PrNPeD	-0.389	-0.393	-0.398	-0.357	0.684	1.000	-0.098	-0.243	-0.363	-0.354	-0.356	-0.352	-0.359	-0.358	-0.357	-0.229
PrCD	0.127	0.122	0.139	0.053	0.043	-0.098	1.000	-0.182	0.052	0.099	0.104	0.111	0.101	0.105	0.111	0.081
PeB	0.600	0.429	0.444	0.494	-0.025	-0.243	-0.182	1.000	0.381	0.383	0.306	0.305	0.385	0.340	0.339	0.110
CB	0.830	0.810	0.837	0.885	0.101	-0.363	0.052	0.381	1.000	0.773	0.762	0.732	0.781	0.774	0.758	0.178
SPrS	0.867	0.971	0.937	0.822	0.090	-0.354	0.099	0.383	0.773	1.000	0.980	0.973	1.000	0.992	0.988	0.068
SPeS	0.841	0.960	0.930	0.787	0.119	-0.356	0.104	0.306	0.762	0.980	1.000	0.993	0.980	0.996	0.994	0.074
SCS	0.826	0.959	0.920	0.754	0.122	-0.352	0.111	0.305	0.732	0.973	0.993	1.000	0.972	0.988	0.996	0.063
RPrS	0.873	0.971	0.942	0.829	0.094	-0.359	0.101	0.385	0.781	1.000	0.980	0.972	1.000	0.993	0.988	0.071
RPeS	0.861	0.969	0.942	0.810	0.113	-0.358	0.105	0.340	0.774	0.992	0.996	0.988	0.993	1.000	0.996	0.074
RCS	0.854	0.971	0.938	0.790	0.118	-0.357	0.111	0.339	0.758	0.988	0.994	0.996	0.988	0.996	1.000	0.067
Rank	0.113	0.076	0.093	0.162	-0.143	-0.229	0.081	0.110	0.178	0.068	0.074	0.063	0.071	0.074	0.067	1.000

This work is obviously preliminary in nature. However, there has not been much comprehensive quantitative research on which social network features are useful in expert recommendation systems. This paper has begun remedying that, but its scope was necessarily limited. There are more potential features that need to be evaluated. In addition, the type of analysis done on the features considered here should be repeated on additional data sets to determine if the results apply in a more general setting. In short, there is much more work remaining to be done in this area.

REFERENCES

[1] M. Schwartz and D. Wood, “Discovering shared interests using graph analysis,” *Communications of the ACM*, vol. 36, no. 8, Aug 1993, pp. 78–89.

[2] R. F. Bales, *Social Interaction Systems: Theory and Measurement*. Transaction Publishers, 2001.

[3] D. W. McDonald, “Evaluating Expertise Recommendations,” in *Proceedings of the 2001 International Conference on Supporting Group Work (GROUP’01)*, September 2001, pp. 214–223.

[4] D. Yimam-Seid and A. Kobsa, “Expert Finding Systems for Organizations: Problem and Domain Analysis and the DEMOIR Approach,” *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 1, 2003, pp. 1–24.

[5] M. Newman, D. Watts, and S. Strogatz, “Random graph models of social networks,” in *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, February 2002, pp. 2566–2572.

[6] F. Wu, B. Huberman, L. Adamic, and J. Tyler, “Information Flow in Social Groups,” *Physica A*, vol. 337, 2004, pp. 327–335.

[7] F. Khan, T. Fisher, L. Shuler, T. Wu, and W. Pottenger, “Mining chat-room conversations for social and semantic interactions,” 2002. [Online]. Available: citeseer.ist.psu.edu/khan02mining.html

[8] M. Newman, “The structure of scientific collaboration networks,” in *Proceedings of the National Academy of Science*, vol. 98, no. 2, January 2001, pp. 404–409.

[9] H. Kautz, B. Selman, and M. Shah, “ReferralWeb: Combining Social Networks and Collaborative Filtering,” *Communications of the ACM*, vol. 40, no. 3, 1997, pp. 63–65.

[10] F. ren Lin and C. hung Chen, “Developing and Evaluating the Social Network Analysis System for Virtual Teams in Cyber Communities,” in *Proceedings of the 37th Hawaii International Conference on System Sciences*, vol. 8, 2004.

[11] L. Hossain, A. Wu, and K. Chung, “Actor Centrality Correlates to Project Based Coordination,” in *Proceedings of the 2006 Conference on Computer Supported Collaborative Work*, November 2006, pp. 363–372.

[12] B. Nardi, S. Whittaker, E. Isaacs, M. Creech, J. Johnson, and J. Hainsworth, “Integrating Communication and Information Through ContactMap,” *Communications of the ACM*, vol. 45, no. 4, April 2002, pp. 89–95.

[13] J. Mori, T. Sugiyama, and Y. Matsuo, “Real-world Oriented Information Sharing Using Social Networks,” in *Proceedings of the 2005 International Conference on Supporting Group Work (GROUP’05)*, November 2005, pp. 81–84.

[14] D. W. McDonald, “Recommending Collaboration with Social Networks: A Comparative Evaluation,” in *Proceedings of the 2003 Conference on Human Factors in Computing Systems (CHI 2003)*, vol. 5, no. 1, April 2003, pp. 593–600.

[15] T. Casciaro and M. Lobo, “Competent Jerks, Lovable Fools, and the Formation of Social Networks,” *Harvard Business Review*, June 2005, p. 92.

[16] M. Stefanone, J. Hancock, G. Gay, and A. Ingraffea, “Emergent Networks, Locus of Control, and the Pursuit of Social Capital,” in *Proceedings of the 2004 ACM Conference on Computer Supported Collaborative Work*, 2004, pp. 592–595.

[17] M. Newman, “Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality,” *Physical Review E*, vol. 64, no. 1, 2001, pp. 64–70.

[18] P. Chirita, A. Damian, W. Nejdl, and W. Siberski, “Search Strategies for Scientific Collaboration Networks,” in *Proceedings of the 2005 ACM workshop on Information retrieval in peer-to-peer networks*, November 2005, pp. 33–40.

[19] L. Adamic, O. Buyukkokten, and E. Adar, “A Social Network Caught in the Web,” *First Monday*, vol. 8, no. 6, June 2003, http://firstmonday.org/issues/issue8_6/adamic/, accessed 2 March 2007.