Recommendation Boosted Query Propagation in the Social Network

This is not the final version of this paper. You can find the final version on the publisher web page.

Grzegorz Kukla, Przemysław Kazienko, Piotr Bródka, and Tomasz Filipowski

Institute of Informatics, Wrocław University of Technology Wyb.Wyspiańskiego 27, 50-370 Wrocław, Poland
grzegorz.kukla@pwr.wroc.pl, kazienko@pwr.wroc.pl, piotr.brodka@pwr.wroc.pl, tomasz.filipowski@pwr.wroc.pl

Abstract. Every single company or institution wants to utilize its resources in the most efficient way and one of the most important resources is knowledge. In the paper, a new SocLaKE system is introduced. It exploits the social network existing within the organization together with information about expertise of community members to recommend the best way to get the answer over the chains of acquaintances. The explanation how the system recommends people and experiments on the sample social network are presented as well.

1 Introduction

In everyday life of every organization, employees ask hundreds of questions and face hundreds of problems. Some of them might be answered quickly owing to guides, forums or content available on intranet or internet web pages, etc. However, very often there are some questions, to which it is hard to find any answer. For that reason, employees usually communicate with the company help desk, office supervisors, etc. waiting for answer or assistance. This so-called "official way" often costs much time and energy. Moreover, it provides no solution in many cases. Hence, the crucial problem within many medium-sized and large organizations is to simplify and speed-up the whole process of finding the right answers to inquiries which occur during the daily work.

"Ask your friends for help" is one of the most common advices one can be given. Maybe your friends already know the answer and will provide you with it? If not, maybe they know someone else who faced the same problem, solved it and now, they may contact you with this person? If your friends cannot help you, then they probably will ask their friends for assistance. In this way, the question is passed along until the satisfying response is found. Why do people do this? Because they are much more likely to help their acquaintances rather than strangers even though their expertise is out of scope of the question. Hence, this approach is based on the rather obvious sociological phenomenon. Namely, if people have a problem that they cannot solve by themselves, they look for help from their friends.

Standard communication systems used within companies, like e-mail or IM, do not contain any explicit information about social relationships. On the other hand, it is possible to extract the data about communication from the messages transferred inside the company's network or the calls being made within it [1, 5]. Moreover, people get into relationships during common activities, while solving tasks, developing projects, participating in meetings, etc. Information about these is often stored in the organizational IT systems. The data about communication and common activities can be used to create a multi-layered social network of the company employees [12].

In this paper we present a new system which uses this social network to improve the communication processes. It boosts query propagation by means of recommendation of some acquaintances who are either likely to provide the right explanation or know some other persons who may know the answer.

2 Related Work

The number of different knowledge management systems continuously grows [20, 4]. Unfortunately, the main source of knowledge for KMS are documents which require advanced analysis of natural language in order to retrieve knowledge of good quality [16]. The next issue is the commonly dynamic nature of knowledge; information cannot be treated as static, always accurate and up to date [14, 4]. A peerless source of knowledge are human experts. The ability to find suitable experts or relevant artifacts created by them is crucial for any modern organization [13]. However, finding an expert is a difficult task and it depends on knowledge artifacts (written statements, documents, reports, etc.) gathered by the organization [1].

Also a specialized recommender systems may be treated as KMS. Recommender systems (RS) have being developed since the middle 90's. Traditional RSs are used in various types of e-commerce and news services [15, 17]. The main goal of the recommender system is to provide a list of objects matching user needs [11]. Specific type of RSs are systems, which recommend people - social matching systems (SMS). Using social connections between humans in SN, the recommender system suggests to user u_x some other SN members with a similar profile [12]. An example of such existing solution is presented in [8].

For the last few decades, social networks (SN) have been intensively studied by computer scientists. Recently, a large number of publications, commercial implementations and theoretical models exist [6, 18, 21]. The general definition of a social network is as follows: a finite set of individuals, who are the nodes of SN, together with relations between them, which are represented by edges of the network. An example of the usage of SN within the expert finding problem is Constellation - an application, which supports expert finding by social network visualization [9]. However, to exploit all the information from the social network structure a little more than social matching or simple visualization is required. To achieve it the social query model (SQM) has been proposed. it supports decentralized answer search by routing queries through a social network [3, 2].

Dispersed search algorithm treats SN as a set of potential experts who anyone can ask a question. In SQM, effectiveness of the routing policy is measured by the probability of retrieving the correct answer. SQM takes into consideration such factors as: an expertise level, correctness, response rate and general policy [3, 2], see Section 4.1 for some details.

3 General Concept of SocLaKE

Query propagation in the social network can be defined as spreading a query from an asker down the path of friends and colleagues. This process is often used by people to find an answer to the question when they do not know a competent expert directly.

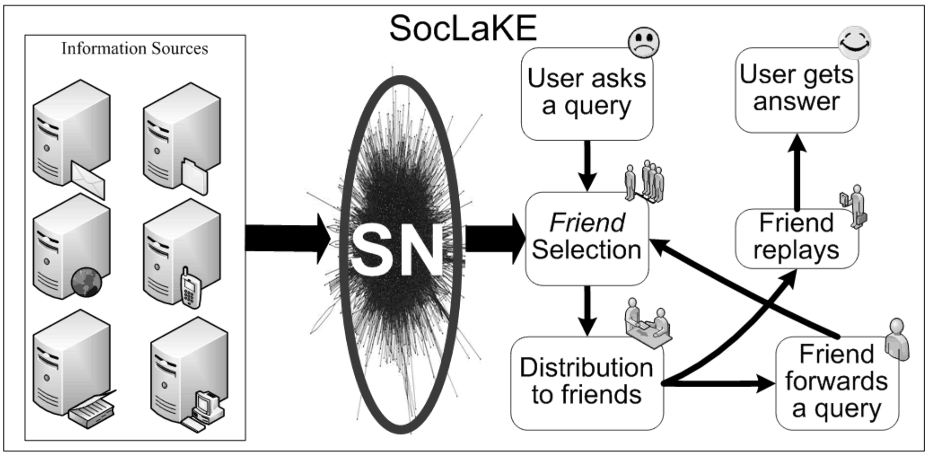
Some basic features of query propagation can be distinguished. Firstly, the query is propagated by such means of communication like phone call, email, text messages or face-to-face talk. Secondly, the way the query is spread in the social network depends only on local knowledge of the nodes. There is no central place from which a query is routed. People passing the query do not consult anyone asking who they should send their request to. Thirdly, there are two aspects by which people decide who to ask questions: substantial and social. Both of them are equally important. People often prefer to ask a question someone they have good relations with rather than the expert they do not know well or not at all [19]. This is a social phenomena of many human activities.

Based on the idea of query propagation over the social network, the system called Social Latent Knowledge Explorator (SocLaKE) was developed. The basic concept of SocLaKE was presented in Figure 1. The SocLaKE system utilizes various data about communication within the organization to extract the organizational social network. Main sources of such data are: email systems, internal phone systems, and information about organization structure i.e. who works with whom or who shares the same room. However, depending on the organization, each IT legacy system and data related to interpersonal communication can be used to extract the more accurate network of human relationships within the organization.

After creating the initial social network SocLaKE needs to gather information about areas of expertise of each member in the social network. All documents starting from official ones through notes, descriptions, forum discussions and ending with the email contents can be used for this purpose. Additionally, once again the organization structure can be utilize, i.e. the data about positions occupied and workers' responsibilities. Moreover, users themselves can provide information about their areas of expertise.

Having the social network created and areas of expertise defined, the SocLaKE system has to compute a set of coefficients which are needed to prepare recommendations. After that SocLaKE is ready to use, it means to support members of the organization to solve their problems. For more details about appropriate coefficients see Sections 4.1, 4.2.

Fig. 1. Social Latent Knowledge Explorator



The SocLaKE system may be embedded into regular organization systems used for everyday work like email agents (Outlook, Thunderbird), web browsers (IE, Firefox, Opera), instant messengers (ICQ, MSN), VoIP systems (Skype) so users do not need to turn on any special application to ask their questions. They can just type their request and send as an email or text in the instant messenger. The SocLaKE system automatically analyzes the query and recognizes its domain. Obviously, if someone is concerned about privacy of the message, then the system can be turned off. Based on the domain discovered and user relationships maintained in the social network, SocLaKE step by step calculates the best route to the expert. Next, the SocLaKE system generates a recommendation using its built-in strategy, see Section 5 for some examples. The recommendation consists of a list of friends from whom, according to the SocLaKE system, the user can get support and finally the answer to a given query. If user decides to use one or more recommendations from the list, SocLaKE sends the appropriate request to user's friends selected from the list. If some of them know the answer, they send them to asker and if he or she is satisfied with the answer, the query propagation is stooped. If none of the friends knows the answer they can forward the query using the same recommendation mechanism as the asker. The question is routed through the network until the answer is found or until the predefined number of network members are provided with the query. If the initial user agrees, the query and its answer are stored in the database and used when someone asks the same or similar question in the future. Thus, the SocLaKE system allows to find, access and gather the latent knowledge in the organization. This kind of knowledge is usually inaccessible by any other means and can be efficiently revealed only by making use of local social relationships existing between members of the large community.

4 Query Propagation Boosted with Recomendation

4.1 Query Propagation Model

The query propagation can be described using the social query model (SQM) introduced in [3, 2]. In this model, nodes in the social network are described by a set of probabilities denoting how people behave when obtain a question related to the certain domain. Using SQM one can estimate the probability of finding an answer to the query propagated over the social network. Originally, authors used their model to discuss an optimal policy of the nodes (humans) in the network, i.e. how nodes should route the query to find a proper answer effectively. SQM has been successfully applied in military knowledge sharing support systems [7].

The probability $P_i^{(q)}$ of finding the answer to a given query q asked by user u_i can be calculated using SQM concept, in the following way:

$$P_i^{(q)} = e_i^{(q)} w_i + (1 - e_i^{(q)}) \sum_{j=1}^{n} \pi_{ij} r_{ij} P_j^{(q)}, \tag{1}$$

where:

- $e_i^{(q)} \in [0, 1]$ denotes the probability that user u_i answers the query,
- $w_i \in [0, 1]$ denotes the probability that the answer of user u_i satisfies the asker,
- $\pi_{ij} \in [0, 1]$ denotes the probability that user u_i asks user u_j,
- $r_{ij} \in [0, 1]$ denotes the probability that user u_j reacts to the query from user u_i.

The probability of finding the answer by user u_i for query q is equal to the probability that u_i is an expert in the query q domain and knows a correct answer, plus a sum of probabilities that all other persons u_j asked by user u_i react and are able to find a right answer. Note that the computing based on Eq. (1) is recursive. Hence, in order to calculate a probability of obtaining the answer for user u_i, all other probabilities have to be estimated.

4.2 Influence of Recommendation on Query Propagation

In this paper, the recommendation will be defined as an ordered, finite subset of options available to the particular user. While making decisions, humans are capable of effective evaluating only a few available options. When there are too many items to choose from, our brain cannot compare them efficiently and individual options are becoming indistinguishable. Usage of recommendations prevents from information overload and enables users to make their decisions in a more efficient way. This is because recommendations are, in fact, a form of pre-evaluation of the larger set of available options. Therefore, recommendations deliver a tool to distinguish between options: there is only a small group of recommended items whereas the others remain hidden. If users trust the recommendation system they will also be able to make use of recommendation lists pretty effectively.

In terms of query propagation, recommendations can be applied at the stage of choosing the next person to pass the query to. In larger organizations, there are usually hundreds and even thousands of people who might be potentially contacted and asked, so choosing among them is almost impossible. Moreover, even if the choice is limited to the closest friends, there are still many options, which are hardly distinguishable. As mentioned before, in such case, recommendations can significantly improve the decision process. Additionally, people are not aware of all skills and competences of all their co-workers. For all these reasons, the usage of recommendations can incorporate a new knowledge into decision process.

In the SocLaKE system, a recommendation is represented by $s_{ij}^{(\Psi)}$:

$$s_{ij}^{(\Psi)} = \begin{cases} n_i^{-\lambda m_{ij}} & \text{if user } u_j \text{ is at position } m_{ij} \text{ on user's } u_i \text{ recommendation list} \\ & \text{of length } n_i \\ 0 & \text{if user } u_j \text{ is not on the list of recommended users to user } u_i. \end{cases},$$

where $\lambda = 0.1$ is the shape coefficient.

The influence of u_j's position and recommendation list length on $s_{ij}^{(\Psi)}$ is illustrated in Figure 2.

The value of $s^{(\Psi)}$ directly depend on recommendation strategy (Ψ), see Section 5 for details and examples. The SocLaKE system uses its recommendation strategy (Ψ) to generate the list of recommended people for every user based on the social network coefficients.

In our model, a recommendation changes user's policy π_{ij}. The final policy π'_{ij} for user u_i, influenced and modified by the recommendation, expresses the u_i's tendency to pass a query to another user u_j. The value of π'_{ij} is expressed as follows:

$$\pi'_{ij} = \frac{\pi_{ij} \exp(\phi_i s_{ij}^{(\Psi)})}{\sum_{k=1}^{n} \pi_{ik} \exp(\phi_i s_{ik}^{(\Psi)})}, \tag{2}$$

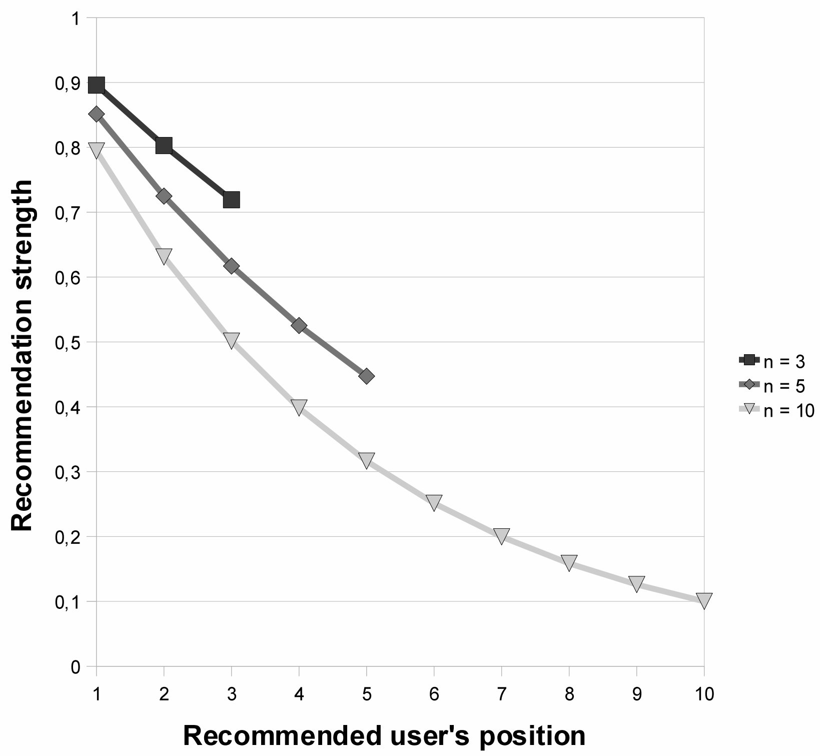
where $\phi_i \in R$ is susceptibility coefficient of user u_i.

Susceptibility ϕ_i denotes in what extent, in general, recommendations influence user u_i. The greater value of ϕ_i, the more likely user u_i follows recommendations and passes the query to the suggested people. Neither values of ϕ_i nor π'_{ij} depend on the query.

5 Strategies of Recommendation

SocLaKE is intended to operate on very large social networks. It also has to work online in order to provide recommendations to users in real time. Therefore, recommender system should be possibly optimized. Its crucial part is the recommendation strategy Ψ. This strategy is responsible for generating a list of

Fig. 2. The relation between recommendation strength and user's position on the recommendation list.



people suggested to each user based on social network data. During the experiments, a set of simple recommendation strategies Ψ were examined, in particular:

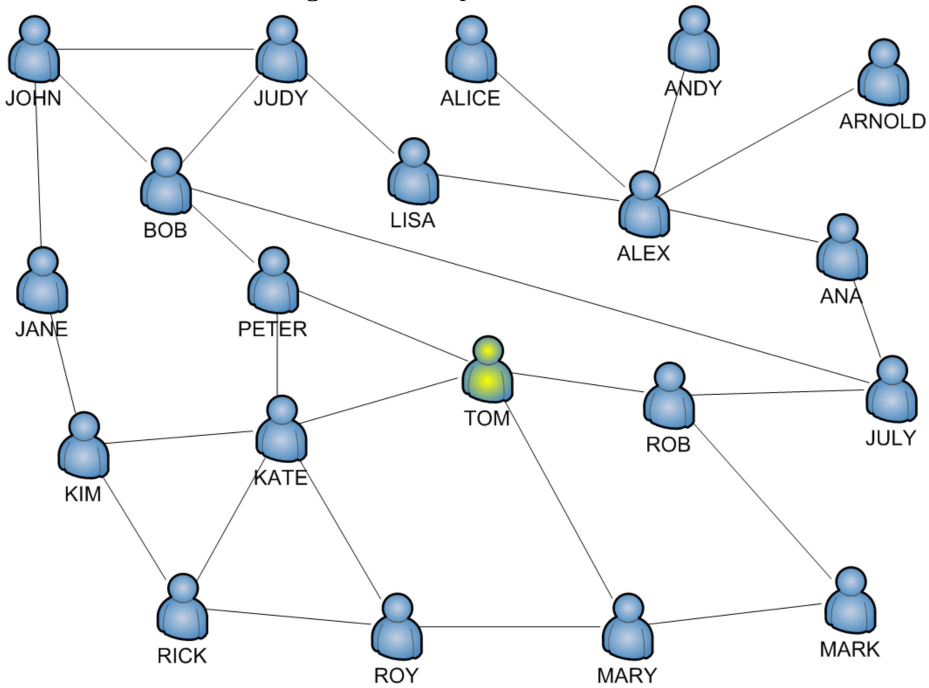
1. No recommendation (none) - system does not provide any recommendations; used to compare with other methods.
2. Random recommendation (m) - for each user, the system recommends m other users ordered randomly.
3. Expert recommendation - only a user with highest expertise is recommended; like in expert finding systems. This strategy is independent from social relationships between people.
4. Best relation (m) - for each user u_i, m other users that are most likely to be chosen according to social relationships are recommended, i.e. by descending π_{ij} value.

5. Best answering (m) - for each user, m other users most likely to respond to the query are recommended. Users u_j are ordered according to the descending value of their responsiveness r_{ij} towards the current user u_i.
6. Best answering (st) - for each user u_i, all others users having the highest response probability r_{ij} and not having exceeded the st limit of responsiveness are recommended. Users are ordered by descending responsiveness value.

6 Experiments

An example social network has been created for the case study and experiments and have been published online[1]; see Figure 3. The additional data, especially appropriate matrices were randomly assigned.

Fig. 3. An example social network



The example social network contains twenty nodes representing company employees (users). Each user u_i has been assigned three values: expertise $e_i^{(q)}$ for a given query q, the correctness w_i of user u_i's answers and susceptibility ϕ_i, see Section 4. Edges reflect relationships between users and are derived from mutual communication (email, phone calls) or common activities, e.g on internet forums

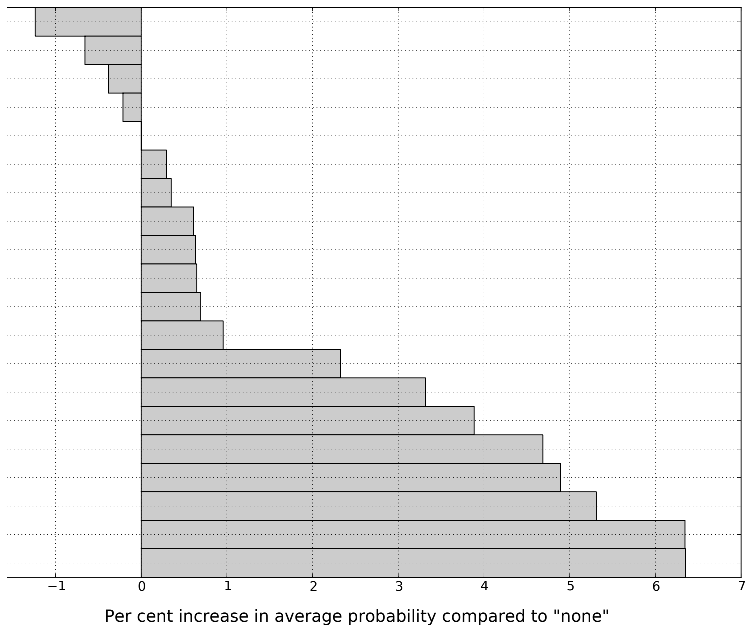
[1] http://www.zsi.pwr.wroc.pl/˜kazienko/datasets/SocLaKE/SocLaKEv1.zip

or co-authorship of documents. Every edge linking from user u_i to u_j possesses two other attributes: policy (π_{ij}) and responsiveness (r_{ij}), see Section 4.1. Overall, edges in Figure 3 represent only user contacts or collaboration extracted from the data. However, there is always a non-zero probability of the contact between nodes not connected by any edge derived from the data. These non-zero values were assigned due to simulation purpose. It means that the final social network is, in fact, a complete graph but most edges have only small values assigned to their attributes.

In experiments the various recommendation strategies were used, i.e. many different recommendation methods were studied. Each strategy estimated a recommendation for each of 20 users using the information about their relations and expertise. The generated recommendations were stored in a recommendation matrix. Then, a modified policy π' for each recommendation matrix was calculated using Eq. (2). Finally, the probability of getting an answer $P^{(q)}$ was estimated using Eq. (1).

The experiments were conducted using python programming language with the numerical calculations package (NumPy) installed.

Fig. 4. Recommendation strategies efficiency comparison



The summary of experiment results are presented in Figure 4 and Table 1. They contain average probabilities of finding an answer for the hypothetical query q using particular recommendation strategies.

Table 1. Average probabilities of finding the answer by means of different recommendation strategies ordered ascending

	Recommendation Strategy	Av.Pr.		Recommendation Strategy	Av.Pr.
1	best relation m=1	0.4181	11	best relation m=5	0.4262
2	best relation m=2	0.4205	12	random m=2	0.4273
3	random m=4	0.4217	13	best answering m=5	0.4331
4	random m=5	0.4224	14	best answering m=4	0.4373
5	none	0.4233	15	best answering m=3	0.4397
6	random m=3	0.4245	16	best answering st=4.0	0.4431
7	random m=1	0.4248	17	best answering st=3.0	0.4440
8	best relation m=3	0.4259	18	best answering m=1	0.4458
9	expert	0.4260	19	best answering m=2	0.4502
10	best relation m=4	0.4260	20	best answering st=2.0	0.4502

The recommendation strategy influences the probability of finding the answer to a query by supporting the query propagation in the social network. The best recommendation strategy applied to the sample social network improved the average probability of finding the answer by over 6%. On the other hand, an improper recommendation strategy can result in lowering this probability. In the experiment, four strategies performed worse than the base strategy (none).

"The expert strategy", which is equivalent to the standard expert recommender system did not perform well. Usage this strategy increased the average probability by less than 1%. The recommended expert - *Alice* responses well actually only one person - *Alex*. Therefore a query is more likely to be answered by *Alice* if it is passed via *Alex*.

Methods recommending people having the highest responsiveness rate (best answering - see Table 1) were classified high. It means that choosing people with the high responsiveness prolongs the life time of the query - increases probability that someone will react somehow. It is worth noticing that according to our assumptions this strategy does not depend on the query q since the responsiveness is also query independent.

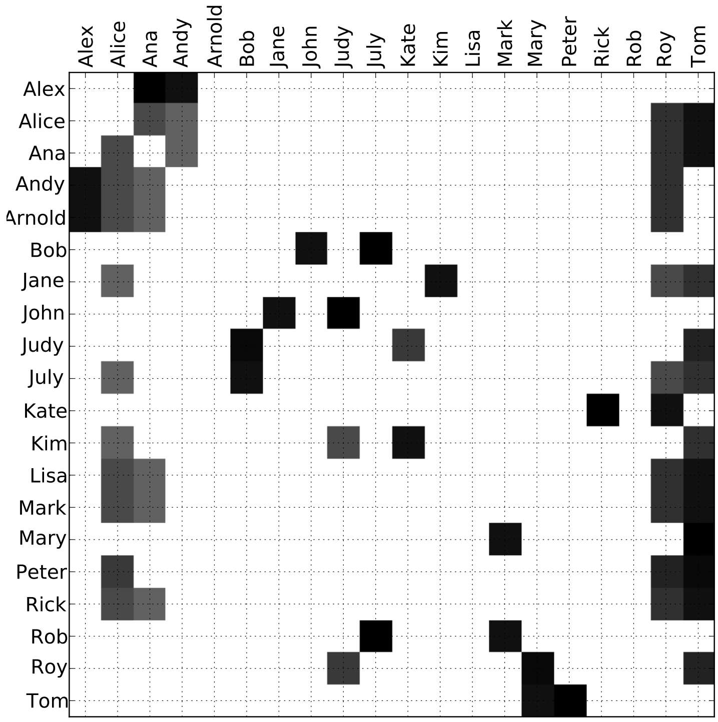
Figure 5 presents the recommendations generated using the best recommendation strategy - "best answering st=2". It has increased the average probability of finding the answer by nearly 6.5%. Every user is recommended from 2 up to 4 other users. The mostly recommended user is *Tom*. The least recommended users are *Arnold* and *Lisa*.

The experiments have shown that using recommendations can improve the efficiency of searching information in the social network. Applying even simple recommendation strategies is beneficial to the query propagation process.

7 Conclusions and Future Work

The problem of sharing tacit knowledge inside organizations has been recognized as a key issue to modern knowledge based companies. The paper addresses one

Fig. 5. Recommendation generated by "best answering st=2" strategy. The darker the field the stronger the recommendation of user in column to user in row.



aspect of this problem, namely sophisticated supporting communication inside the social network of the company. The novel SocLaKE system is proposed to carry out this task by means of recommendations. SocLaKE combines the knowledge about expertise together with social relations between members of the organization in order to provide its users with high quality information of who to ask for help when dealing with a certain topic. Besides, SocLaKE can be easily incorporated into organization's communication infrastructure and transparently support communication between its members.

The most important part of the system is the recommendation strategy. This component generates recommendations for a given user and query. A few different strategies have been examined and compared in the paper. Additionally, in order to evaluate the strategies, the social query model with recommendations has been introduced. According to the simulation performed, almost all proposed strategies improve the probability of finding the answer even though to different extent.

There are a few key issues left as future work. Before actual deployment of the SocLaKE system the privacy issues need to be addressed. The proposed solution uses sensitive data extracted from communication systems and as such has to ensure its confidence.

The other problem is the load balancing the communication inside the social network. SocLaKE should not allow its users to be overwhelmed with the amount of messages routed to them because it would result in decreased responsiveness.

The dynamic aspect of social networks is also a very important issue. The SocLaKE system should adapt easily to ever-changing relations between people as well as to organizational changes inside the company.

All of these issues will have to be addressed before the deployment of the real-world system.

8 Acknowledgement

The work was supported by The Polish Ministry of Science and Higher Education, the development project, 2009-2011.

References

1. Balog, K. and Azzopardi, L. and de Rijke, M.: Formal models for expert finding in enterprise corpora. Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval ACM (2006) 50
2. Banerjee, A. and Basu, S.: A social query model for decentralized search. Proceedings of the 2nd Workshop on Social Network Mining and Analysiss ACM New York (2000)
3. Banerjee, A. and Basu, S.: A Social Query Model for Distributed Search. University of Minnesota Minneapolis (2008)
4. Benbya, H: Knowledge Management Systems Implementation: Lessons from the Silicon Valley. Chandos Publishing Cambridge(2008)
5. Campbell, C.S. and Maglio, P.P. and Cozzi, A. and Dom, B.: Expertise identification using email communications. Information and knowledge management (2003) 528–531
6. Carrington, P.J. and Scott, J. and Wasserman, S.: Models and Methods in Social Network Analysis. Cambridge University Press New York (2005)
7. Davitz, J. and Yu, J. and Basu, S. and Gutelius, D. and Harris, A.: iLink: search and routing in social networks. Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (2007) 940
8. de Oliveira da Silva, F.E. and Motta, C.L.R. and Santoro, F.M. andde Oliveira, C.E.T.: A Social Matching Approach to Support Team Configuration. Lecture Notes in Computer Science Springer Berlin / Heidelberg **5784/2009** (2009) 49–64
9. DiMicco, J.M. and Yankelovich, N.: Constellation: Using Visualization to Find the Path to Experts. Poster Presentation at the ACM GROUP Conference New York (2007)
10. Iwata, T. and Saito, K. and Yamada, T.: Modeling user behavior in recommender systems based on maximum entropy Proceedings of the 16th international conference on World Wide Web ACM (2007) 1282
11. Kazienko, P.: Web-Based Recommender Systems and User Needs - the Comprehensive View. Frontiers in Artificial Intelligence and Applications **181** (2008) 243–258

12. Kazienko, P. and Musiał, K. and Kajdanowicz, T.: Multidimensional Social Network in the Social Recommender System. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans **accepted** (2010) in press
13. Maybury, M.T.: Expert Finding Systems. Mitre McLean (2006)
14. McInerney, C.: Knowledge Management and the Dynamic Nature of Knowledge. Journal of the American Society for Information Science and Technology **53(12)** (2002) 1009–1018
15. Montaner, M. and López, B. and de la Rosa J.L.: A Taxonomy of Recommender Agents on the Internet. Artificial Intelligence Review **19** (2003) 285–330
16. Rosner, D. and Grote, B. and Hartman, K. and Hofling, B. and Guericke, O.: From natural language documents to sharable product knowledge: a knowledge engineering approach. The Journal of Universal Computer Science **3** (1997) 955–987
17. Schafer, J.B. and Konstan, J.A. and Rield, J.: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery **5** (2001) 115–153
18. Scott, J.: Social network analysis: A handbook (2nd ed.). Sage Publications Ltd London (2000)
19. Shami, N.S. and Yuan, Y.C. and Cosley, D. and Xia, L. and Gay, G.: That's what friends are for: facilitating'who knows what'across group boundaries Proceedings of the 2007 international ACM conference on Supporting group work ACM (2007) 379–382
20. Stefanelli, M.: Knowledge management to support performance-based medicine. Methods Inf. Med. **41(1)** (2002) 36–43
21. Strogatz, S.H.: Exploring complex networks. Nature **410** (1998) 63–65