Leonard Bolc
Marek Makowski
Adam Wierzbicki (Eds.)

LNCS 6430

Social Informatics

Second International Conference, SocInfo 2010
Laxenburg, Austria, October 2010
Proceedings

Springer

Lecture Notes in Computer Science 6430

Leonard Bolc Marek Makowski
Adam Wierzbicki (Eds.)

Social Informatics

Second International Conference, SocInfo 2010
Laxenburg, Austria, October 27-29, 2010
Proceedings

Springer

Volume Editors

Leonard Bolc
Polish-Japanese Institute of Information Technology
86 Koszykowa, 02-008 Warsaw, Poland
E-mail: pjwstk@pjwstk.edu.pl

Marek Makowski
International Institute for Applied Systems Analysis (IIASA)
Schlossplatz 1. 2361 Laxenburg, Austria
E-mail: marek@iiasa.ac.at

Adam Wierzbicki
Polish-Japanese Institute of Information Technology
86 Koszykowa, 02-008 Warsaw, Poland
E-mail: adamw@pjwstk.edu.pl

Library of Congress Control Number: 2010936698

CR Subject Classification (1998): C.2, H.5, H.4, H.3, I.2.6, J.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-642-16566-4 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-16566-5 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

As information technology became ubiquitous, it did not take long for practically minded ICT specialists to realize the technology's potential for supporting and enhancing our social activities. Today, it is a truism to say that information technology has a social dimension and a social impact—it is enough to consider such applications as Facebook, LinkedIn, or Wikipedia. Proponents of the social applications of ICT will go further and claim that information technology is reshaping the way we are doing business, working, learning, playing, and making friends or enemies. They will say that, for example, Wikipedia has the potential to completely change our economy (following Don Tapscott and Anthony Williams, the authors of *Wikinomics*).

Computer science was slower than the social sciences to direct its interests toward the social uses of its products. The concept of social informatics was first invented by Rob Kling, who deemed it as an area of study of information and communication tools in cultural or institutional contexts. However, he was not the only one, as for example ethnographists quickly became interested in the ways ICT influences our culture (consider the studies of YouTube by Michael Wesch). Media science and sociology have been geared up by the seminal works of McLuhan and are ready to tackle the new media created by ICT, as in the work of Manuel Castells. Psychologists quickly became interested in the Internet and computer games. Economists recognized the quickly rising impact of e-commerce and e-business and focused much effort on their analysis.

For some years now, computer scientists have been increasingly interested in the social dimension of ICT. These interests have taken various forms and followed various paths of development. Tim Beners-Lee has proposed what he has called "Web science," as an area of research of the social impact of the Web and of how the Web can benefit from social concepts. In a seminal article in *Science* (February 2009), representatives of the social network community proposed to consider ICT as a rich source of information that should become the basis of "computational social science." This concept is based on the work of projects such as "Reality Mining" at MIT that have indeed gathered enormous quantities of data about human behavior that can be used, for example, to drive social simulation. Theoretical computer scientists have also found inspiration in the social sciences, and have created a new area of study: computational social choice theory. Social computing is also an area of interest in the community of artificial intelligence, among Web development professionals, and in other areas of ICT.

What is apparent today is that all of these diverse and fascinating areas of inquiry have one feature in common: they attempt to go beyond the task of merely using ICT as a source of data for social science. The reason for this ambition is that computer science is an inherently practical domain, where practical applications are what matters most at the bottom line. Thus, computer scientists and ICT professionals alike would like to improve the way ICT realizes social goals, supports social processes or applies social concepts. At the same time, ICT technology can become a source of information relevant for social sciences and for analysis of social phenomena (such as political or marketing analysis).

A very succinct way of describing the relationship between the Internet (in particular, the Web) and society today can be given by paraphrasing Michael Wesch: "the Web is us." This statement is similar to McLuhan's famous "the medium is the message." Computer scientists working in social informatics have been trying to add a corollary to these statements of social scientists: "The Web is us, but we have made the Web." "The medium is the message, but we have built the medium." These rephrased statements emphasize an important point: it is in our power to change the Web, the Internet, or any ICT technology. And we should aspire to change them in ways that will improve their social applications.

The two perspectives of study—of social sciences that consider the impact of ICT on social behavior, and of social informatics that considers how ICT can be improved to realize social goals—are two sides of one coin. Social informatics can and must base its research on findings from the social sciences. On the other hand, social science can find applications for its theoretical findings for improvements of ICT technology. These applications already have a real business value today in, for example, software for the support and management of virtual teamwork, available from many major ICT technology companies. Using the Internet as a source of information about social phenomena also has a real business value, as many e-marketing companies attempt to base their analyses on Web mining. The Conference on Social Informatics (SocInfo) has been launched as an attempt to bring together representatives of computer science and the social sciences, recognizing that social informatics is an interdisciplinary domain.

The interdisciplinary makeup of the conference is reflected in the Program Committee that includes both computer scientists and social scientists. All articles submitted for the conference have received reviews from specialists from both domains. Authors of accepted papers also have backgrounds in computer science and the social sciences. This emphasis on balancing the inputs of computer scientists and social scientists is motivated by the desire to make the conference a meeting place between the two disciplines.

SocInfo 2010 and SocInfo 2009 each received a large number of papers concerning social networks. Both theoretical social network analysis and its practical applications for social recommendation are represented at the conference. Another important subject concerns social aspects of virtual collaboration, ranging from social studies of computer supported collaborative work, to the study of enhancements of Wiki technology. Research on Web mining, opinion mining, and sentiment analysis has also been represented at the conference. Privacy and trust are subjects very relevant to social informatics that are an active area of research. Computational social choice is reflected by papers that consider the distributive and procedural fairness of ICT applications and systems. Last but not least, virtual teamwork is studied by several papers that attempt to recommend the best possible virtual teams or team members for a given task.

Social informatics is a maturing discipline that already has a unique set of research objectives and methods. Among these objectives, the questions posed by Tim Berners-Lee is a prominent example: can we analyze and anticipate the social properties of Web 2.0 applications before their deployment, and can we use this knowledge to improve the design of social Web applications and services? To achieve such goals, social informatics must have a deeper understanding of social goals, concepts and mechanisms. For this, social informatics must be an interdisciplinary science that draws upon the results of sociology, economics, psychology, anthropology, behavioral game theory, sociobiology and others. The SocInfo conferences are an attempt to direct research efforts toward that goal, and to enable the necessary exchange of views, knowledge and research ideas between representatives of computer and social sciences. SocInfo 2010 was an important step towards this goal.

October 2010

Adam Wierzbicki
Leonard Bolc
Marek Makowski

Organization

- Polish-Japanese Institute of Information Technology (PJIIT)
- International Institute for Applied Systems Analysis (IIASA)
- Gemius S.A. - Sponsor

Conference General Chairs

L. Bolc (Poland) - Publication Chair
M. Makowski (Austria) - Program, Co-chair
M. Makowski (Austria) - Local Chair
A. Wierzbicki (Poland) - General Chair

Conference Committee Members

K. Aberer (Switzerland)
A. Abraham (Norway)
T. Ågotnes (Norway)
G. Barnett (USA)
N. Bennani (France)
S. Bostandjiev (USA)
U. Brandes (Germany)
S. Buchegger (Germany)
C. Cornelis (Belgium)
P. Cruickshank (UK)
E. Damiani (Italy)
A. Datta (Singapore)
M. De Cock (Belgium)
Y.A. de Montjoye (Belgium)
N. Eagle (USA)
D. Feng (USA)
F. Wai-Tat (USA)
C. Goble (UK)
B. Gros (Spain)
J. Hołyst (Poland)
A. Jatowt (Japan)
L. Hung Vu (Switzerland)
M. Jelasity (Hungary)
A. Jøsang (Norway)
J. Kacprzyk (Poland)
P. Kazienko (Poland)
G. Kossinets (USA)
G. Kossinets (USA)
R. Kotowski (Poland)
K. Lamberts (UK)
J. Lang (France)
K. Lerman (USA)
M. Makowski (Austria)
D. Millard (UK)
J. Minguillón (Spain)
B. Mobasher (USA)
M. Morzy (Poland)
T. Murata (Japan)
J. O'Donovan (USA)
W. Ogryczak (Poland)
A. Papaioannou (Switzerland)
G. Petric (Slovenia)
L. Ripamonti (Italy)
A. Robbin (USA)
I. Rodrguez-Ardura (Spain)
G. Roussos (England)
S. Ruohomaa (Finland)
J. Sabater-Mir (USA)
V. Snášel (Czech Republik)
D. Taraborelli (UK)
K. Tutschku (Austria)
M. Weal (UK)
A. Wierzbicki (Poland)
W. Zhang (USA)

Reviewers

A. Carroll	Psychster LLC (USA)
W. Chen	University of Bergen (Norway)
A. Chmiel	Warsaw University of Technology (Poland)
G. Dias	University of Beria Interio (Portugal)
J. Dvorsky	VSB-Technical University Of Ostrava (Czech Republic)
T. Gjøsæter	University of Bergen (Norway)
M. Heckner	Accenture Information Management Services (Germany)
Z. Horak	VSB Technical University of Ostrava (Czech Republic)
A. Hupa	Warsaw University (Poland)
J. Kessler	Indiana University (USA)
M. Kudelka	VSB Technical University of Ostrava (Czech Republic)
H. Liu	Arizona State University (USA)
R. Nielek	Polish-Japanese Institute of Information Technology (Poland)
V. Noncheva	University of Plovdiv (Bulgaria)
O. Nov	New York University (USA)
L.E. Olsen	University of Bergen (Norway)
A. Wawer	Polish Academy of Sciences (Poland)
C. Wolff	Media Computing University of Regensburg (Germany)
R. Zafarani	Arizona State University (USA)

Table of Contents

Case-Based Team Recommendation

Yonata Andrelo Asikin, Michele Brocco, and Wolfgang Woerndl

Technische Universitaet Muenchen
{asikin,brocco,woerndl}@in.tum.de
http://www11.in.tum.de

Abstract. Team recommendation is required for composing an appropriate team for a particular task or project by selecting/choosing among the adequate/best team members. Usually project managers decide how to compose a team based on their experience in similar projects. Given this best practice we propose to algorithmically compose appropriate teams for a task by applying case-based reasoning on a previously developed meta-model for team recommendation. We evaluate our approach through comparing the ranking given by a domain expert with the result of our recommender and conclude with a discussion of these results.

Keywords: team recommendation, case-based reasoning.

1 Introduction

Humans live in a society and cooperate to solve a problem or to achieve a common objective. In many fields tasks or projects are accomplished in a team composed of members with different roles and competences. The importance of teamwork is growing and has led to a vast amount of research on team composition (see Section 2). Finding the best team for a task among a group of persons is obviously a very complex problem.

Assembling teams in large-scale communities such as open innovation networks is even more complex because of their high amount of possible team members. Open innovation [short OI] is defined by Chesbrough [6] as "the use of purposive inflows and outflows of knowledge to accelerate internal innovation, and expand the markets for external use of innovation, respectively. [This paradigm] assumes that firms can and should use external ideas as well as internal ideas, and internal and external paths to market, as they look to advance their technology." Open innovation networks can then be defined as a cooperation network of individuals and organizations participating in an or a number of innovations. An innovation shall be "manifested" in a team consisting of the members of this OI-Network. A lot of challenges arise regarding this team assembling in the open innovation network. Firstly, the large number of members in the community implies an even larger number of possible combinations of these individuals. Secondly, a team mostly consists of heterogeneous roles and has several properties. The problem becomes more complicated because a proper combination of these members' properties has to be considered as well. In an

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 1–18, 2010.

open innovation network, the members do not necessarily know each other. Competences of the members can be acquired for example from their profiles, from contributions in collaborations such as blogs, forums, or wiki entries, or their past experiences in earlier innovation projects. However, the large number of individuals and the deficiency of common social knowledge (e.g., trustworthiness) among the members make it difficult to find an appropriate team manually.

Moreover, the chosen individuals not only have to be capable of accomplishing a task in the team with their competences, but also have to fulfill possibly existing constraints, such as availability or location. Furthermore, it has to be considered, that an individual normally performs differently in distinct teams and situations. For example, a "web programmer" might perform better if she works remotely on a project from another country instead of on a project where she has to come to office everyday to report the progress. This aspect has to be considered as *the context* of team composition. We will discuss this context in more detail in the next sections. Taking the various variables into account and building an algorithmic team recommendation approach to deal with these variables is therefore challenging.

In practice, an experienced project manager could build a team and choose the best suitable individuals for the team regarding *prior* knowledge gained in the past. She does not necessarily have a particular or systematic method to find her "favorite" team, instead she just learns from her compositions in the earlier projects. Given a new project, the project manager would remember at handling several similar projects in the past. All she has to do is then choosing the best teams that performed in those similar projects and take them into consideration for the new project.

Based on this best-practice, we apply case-based reasoning (CBR) to solve the problem of team recommendation. This approach learns from the past cases in order to give recommendations of suitable teams for a given new case. It assumes that a project or a task can be solved by the teams, which have already worked on similar projects with good results in the past.

The remainder of this paper is structured as follows: the next section lists the related work in team recommendation and case-based reasoning. In Section 4 we incorporate our previous work on a meta-model for team composition in a case-based approach and then use it for building a recommender system in Section 5. An evaluation of the approach is described in Section 6 and is followed by a discussion of the results and concluding remarks.

2 Team Composition

A number of socio-psychological works have already investigated and analyzed the problem of team composition or recommendation. Most of them have focused on social aspects such as trust, heterogeneity of team member properties, organizations, conflicts, etc. [5] Additionally, there are also a considerable number of other algorithmic approaches especially using methods of artificial intelligence.

As described before, team recommendation can be encouraging for finding teams in an open-innovation network or other large-scale communities. A potential application of team recommendation is shown for instance in [17]. This article presents an approach for evaluating teams of contributors in Wikipedia based on social network analysis. The approach implicitly creates a social network of the authors based on the Wikipedia edit history, i. e. the summary of all recorded interactions among the contributors. The network consists of four variables: trust, distrust, acquaintance and knowledge and can be used to assess the quality of a team of authors. Based on the quality measurement the proposed method can recommend good teams of individual collaborators for new articles/projects in the platform. The approach shows encouraging results such as the relevancy of social aspects for team composition. In more general domains or platforms like open innovation networks, the approach could however require further aspects, for instance the categorization of projects (articles) and the properties of a user (an author).

A large group of research in algorithmic team composition uses the concept of multi-agent systems. Multi-agent systems are cooperative systems consisting of a complex network of autonomous yet interdependent agents, that interact to solve tasks. In contrast to top-down approaches where a central instance tries to compose a team, multi-agent systems enable the bottom-up composition of human teams: Each agent represents a human with her characteristics (team member variables) and communicates and computes its suitability with other agents for a task in order to build a team. Examples can be found in [11,1,10].

Another type of algorithmic approach can be for example found in [7]: an approach for the semantic composition of teams based on provided individuals skill and task descriptions. The approach is able to provide logic-based answers in case a completely satisfactory team can not be composed due to lack of desired skills. The composition criteria (i.e. the composition strategy) are static and represent only a single composition strategy. The approach is hence not flexible i.e. only skill matching and no other composition strategy can be used.

In the area of case-based recommendation, [16] presents an overview of case-based recommender systems in general. The paper consistently shows the techniques and knowledge required for building a case-based recommender and provides sufficient references, case studies, and project examples already implemented in the real world that can be taken into consideration when building a case-based recommender system.

In our previous work [4,5] we presented challenges in algorithmic recommendation of teams based on a literature review on several team composition approaches from the socio-psychological sciences and introduced a meta-model for team composition. The meta-model consists of a collection of aspect entities. These aspect entities can guide the consideration of all properties of a team required in team composition. Therefore, in the CBR approach designed in this paper we intend to cover all of the entities in the team recommendation process (Section 4).

3 Case-Based Reasoning for Recommendation

Recommendation techniques can be categorized in two basic flavors, namely *collaborative filtering* and *content-based techniques*. Case-based recommenders are a particular type of content-based recommenders. The recommenders rely on items that are represented in a structured way using a well-defined set of features and feature values. The availability of similarity knowledge for each type of feature enables the case-based recommenders to produce high-quality suggestions to the user [16].

Case-based recommender systems originate in *case-based reasoning* (short CBR) techniques. These techniques were initially used in various problem solving and classification tasks. Case-based reasoning is an approach to problem solving that emphasizes the role of prior experience. The experience is expressed in form of previously concrete problem situations which are called *cases*. A *case* is therefore defined as a past problem with its solution.

CBR solves new problems by learning and reusing the solutions of past similar problems that were already solved. The process of finding the similar problems is called *similarity assessment* (see Section 5.3). The similarity assessment is basically a comparison of properties between two *cases*. These *case* properties are called *features*. The retrieved solutions can further be *adapted* to the current problem if necessary. Therefore, case-based reasoning differs from other problem solving (*machine learning*) methods in their reliance on concrete experiences instead of problem solving knowledge in the form of codified rules and strong domain models. Figure 1 depicts the basic idea of solution retrieval in CBR.

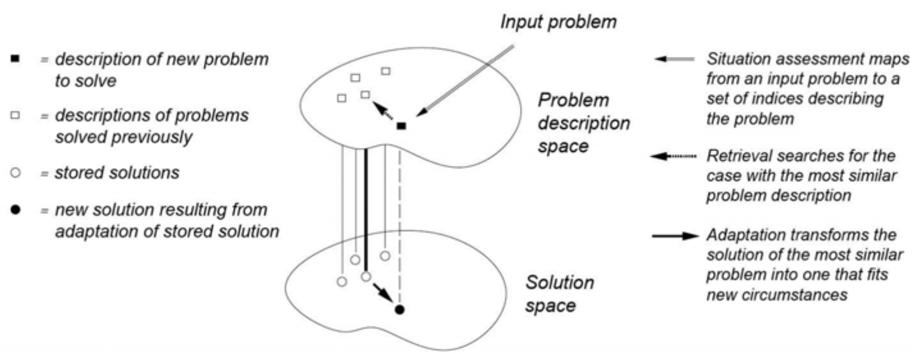


Fig. 1. CBR is the process of solving new problems based on the solutions of similar past problems [14]

Aamodt & Plaza introduced in [2] the classic model of problem solving cycle in CBR that is widely used as the general aspects of CBR. The cycle consists of the following individual tasks:

1. *retrieving* one or more most similar experienced *cases*,

2. *reusing* the solution of that *case* in one way or another to solve the problem,
3. *revising* the knowledge or solution proposed by this previous *case*,
4. *retaining* the parts of this experience likely to be useful for future problem solving by incorporating it into the existing *case base*.

These tasks will be used as guidelines for the development of our CBR approach for team recommendation in Section 5.

4 Meta-Model for Team Composition

Section 2 already mentioned a proposed meta-model of universal team recommendation. It consists of the aspect entities **team member variables (TMV)** (i.e. properties of team members), **outputs/outcomes (OUT)** (e.g. performance, idea quantity, satisfaction), **team intermediate variables (TIV)** (e.g. conflicts, group synergy), **team external variable (TEV)** (context in which the team performs), **dependencies** (relations between team member variables and outputs) and organizational **constraints** (e.g. availability of team members). Based on these aspect entities we construct our CBR approach. We refer to [4] and [5] for further details on the aspect entities.

Most of the team member variables (e.g. age, gender, etc.) and team external variables (e.g. reward system, task) can be used directly as fundamental features of *cases* in CBR. Notice that not all features are necessary to be considered in the recommendation process, instead this should be configurable. Basically, this also holds for other classes of features. Outputs/outcomes have an important role as the part of fundamental features in our CBR approach and the rank mechanism of recommended past *cases*.

In our approach, the comparison and learning of the dependencies support the automatic adaptation of (recommended) *cases*. It means that if there are no appropriate *cases* available to be recommended, the learning of the dependencies of the (limited) existing *cases* shall be sufficient to perform the composition of teams (see Section 5.4).

Finally, the constraints are especially important in the query phase of the recommendation process. The existence of constraints reduce the number of recommended teams because members or teams which do not satisfy the constraints will be eliminated directly before the *case* retrieval process.

5 Recommendation Approach

The four phases covered in the problem solving cycle of CBR (see Section 3) contribute as the basis of our recommendation approach. In this section we consider how the concepts of CBR can be applied to team recommendation and then describe how the meta-model can be incorporated into the proposed CBR approach. Furthermore, the essential CBR components are defined.

5.1 Case Representation and Definitions

We can define our *problem* and *solution* spaces: “Problem” is the description of a project (given task) consisting of (i) a specification of the project, (ii) a specification of the required members, (iii) constraints. All these specifications are configurable depending on the team composition strategy that has to be applied. For instance, if for specifying the project the project length and the presence of reward system is indispensable then corresponding features have to be defined (e.g., *project Length In Months=6,RewardSystem=true*). “Solution” is a team (i) whose specifications are similar to the specification of the problem; (ii) whose members' specifications are similar to the required members (positions); (iii) whose members fulfill the constraints. For (i) the same configuration aspects apply as for the above described “problem” space.

In comparison to case-based recommendation in other domains, e.g., a product recommendation with a number of static features, the *case* representation of a case-based team recommendation is more complex. This is attributed to the fact that (i) there are two kinds of fundamental variables, namely the variables of individual team members and the variables of the team itself that are the result of specific mappings of the individual team members (e.g. variety of gender in a team). The team member variables alone are, as mentioned earlier, not sufficient as the factor of a *good* team, because the members have not only to be *good* as an individual but also *good* in the required context (project); (ii) there are outputs to be maximized and dependencies between the team member variables and the outputs.

To solve these issues, we use a CBR approach with two levels of *cases*. The **first level** or the main *case* is denoted as TEAMCASE and consists of a specific set of team members (*a team*) and another set of *contexts (projects)*. Reusing of a team directly in a new context (project) results in the addition of the project into the set of contexts in the TEAMCASE. Revising the team of a TEAMCASE through addition/replacement of members or elimination of a team member yield a new TEAMCASE. Members are regarded as **second level** *cases* (denoted as MEMBERCASE) because a member consists of features (team member variables) and these features are required in the similarity assessment as well. The following example depicts our approach more clearly: Suppose that a best team for a “web portal” project is already recommended by the system, but the user prefers to eliminate the “web designer” for some reason and demands new recommendations for this specific position in the current team. This elimination is enabled as the part of conversational recommendation and regarded as user feedback. The CBR system recommends members suitable for this position in the required context and combines the recommended members with the returned team. In the case there are not (sufficient) *team cases* available yet, the proposed recommender system assembles the provided members by measuring their similarity to the given member specifications. At this point we introduce the following notations: A selection of team member variables TMV is defined as TMV'. E.g., $TMV' = \{age, gender\}, mc_k = (24, m)$. Similarly, TEV' denotes the selection of team external variables. E.g., $TEV' = \{projecttype, projectlength\}, ctx_k =$

$(webportal, 3)$. The selected outcomes are defined in OUT'. E.g., $OUT' = \{rating$ $of\ manager\}$, $out_k = (7.5)$. We define the described *cases* as follows:

$$\begin{aligned} TEAMCASE\ &:\ (mc, (ctx, out)) \\ mc &= mc_1, mc_2, ..., mc_N \quad \text{a set of} \quad N\ MEMBERCASEs, \\ (ctx, out) &= \{(ctx_1, out_1), (ctx_2, out_2), ..., (ctx_M, out_M)\} \\ &\quad \text{a set of tuples of M past projects (contexts) with outcomes,} \\ mc_k &\quad \text{is a vector of features defined by} \quad TMV', \\ ctx_k &\quad \text{is a vector of features defined by} \quad TEV', \\ out_k &\quad \text{is a vector of features defined by} \quad OUT'. \end{aligned}$$

5.2 Query

From the formalized *case* representation, we can derive the problem specification (QUERY) to the system. The query specifies how a user interacts with the system in order to retrieve the recommendations for a desired team.

$$\begin{aligned} QUERY\ &:\ (q, eq, con) \\ q &= \{q_1, q_2, ..., q_N\} \quad \text{denotes the specification of required members} \\ eq &= \text{vector of specifications in} \quad TEV' \\ con &= \text{vector of constraints in} \quad CON' \\ q_k &\quad \text{is a vector of member specifications in space of} \quad TMV'. \end{aligned}$$

The variable q_k corresponds to the variable t_k defined in the *case* representation. E.g., $TMV' = \{age, gender\}, q_k = ([25, 45], \{m, f\})$. The notation $[25, 45]$ is an example of an age range, that specifies that the age of the member must be between 25 and 45 years. The second notation in q_k $(\{m, f\})$ says that both, *male* and *female* are accepted as gender of the team member. We can define the specification of required external team variables in the same way. E.g., $TEV' = \{projecttype, projectlength\}$, $eq = \{\text{“news portal”}, -\}$. In this definition, the variable *eq* specifies the project type as “news portal” and does not limit the project length. The inclusion of constraints in the algorithm requires the availability of member's information besides the task-related team member variables. For example, $CON' = \{time, location\}, con = \{\text{“}June - August\text{”}, \text{“}Munich\text{”}\}$. In the definition, all members have to satisfy two constraints variables *time* and *location*. The constraints are satisfied by using hard constraints, in contrast to other entities, where through similarity assessment soft constraints are applied.

5.3 Similarity Assessment

As mentioned earlier in Section 3, similarity assessment defines the degree of similarity between past problems and the current problem in the case-based reasoning process. A solution of the problem in a more similar *case* is assumed to be a better solution than other less similar *cases*. The performance of CBR mostly depends on the suitability of an underlying similarity measurement [3].

The similarity can be computed partially, such that not all features have to be considered. Generally, similarity assessments use the concept of *nearest neighbor*. Computing the distance between a target query t and a *case* c with n features can be acquired through the L_k norm (Euclidean distance if $k = 2$): $L_k(t, c) = \left(\sum_{i=1}^{n} |t_i - c_i|^k\right)^{1/k}$.

The L_k norm distance measure can be used if we know that the features of *cases* are regarded with the same importance degree and can be computed in a uniform way. This is apparently not the case for a complex problem solving such as team recommendation with different kinds of features. More sophisticated similarity metrics have to be used by case-based recommenders, that are based on an explicit mapping of various *case* features and the availability of specialized feature level similarity knowledge. We define the standard similarity function of the two *cases* as the following weighted sum metric:

$$Similarity(t, c) = \frac{\sum_{i=1..n} w_i * sim_i(t_i, c_i)}{\sum_{i=1..n} w_i}. \tag{1}$$

In this equation, w_i denotes the weight encoding the relative importance of a feature in the similarity assessment. The assessment involves combining the similarity measurement of individual features defined by $sim_i(t_i, c_i)$. This individual feature level similarity is calculated in various ways depending on the type of the feature, e.g., *numerical*, *hierarchical* or *categorical*. The value of each similarity measurement is assumed to be normalized into the range of [0, 1]. If the similarity is assessed through a distance measure, then we have to consequently regard 0 as the highest similarity value across all of the features.

Similarity assessment of various feature types is one of the widest areas of research on CBR. [8], [3] and [13] provide an overview on several different similarity measures for different feature types. The most common feature type is the numeric feature. A numeric feature can further be categorized in *symmetric* and *asymmetric* features, which again requires distinctive similarity metrics. An example for a asymmetric feature is e.g. the required experience of a team member. If for instance a member with 5 years of experience is required, a candidate *member case* with 6 years of experience should be more preferable than another one with only 4 years of experience. Most of these individual similarity measures can also be applied to other feature types provided that the features are mapped to numerical value. For instance, the feature "grade" can have categorical values which can be mapped as "A=1", "B=2", etc. The feature type that does not require any mapping to numerical value can for instance use the string matching (comparison of categorical features) or Levenshtein distance (distance between nodes in a tree of hierarchical feature values).

Equation 1 can be applied directly for measuring the similarity between two vectors of features in the spaces of TMV' and TEV'. We can rewrite the equation to assess the similarity between a *member case* mc_k and a specification vector q_k for a member with $n = |TMV'|$:

$$Similarity_{TMV}(q_k, mc_k) = \frac{\sum_{i=1..n} w_i^{TMV'} * sim_i(q_i^k, mc_i^k)}{\sum_{i=1..n} w_i^{TMV'}}. \tag{2}$$

Likewise, the similarity between a vector of TEV' specification eq and a project ctx_k is measured with $(n = |TEV'|)$:

$$Similarity_{TEV}(eq, ctx_k) = \frac{\sum_{i=1..n} w_i^{TEV'} * sim_i(eq_i, ctx_i^k)}{\sum_{i=1..n} w_i^{TEV'}}. \tag{3}$$

At this point, we are already able to formalize our first global similarity measure between a $QUERY$ and a $TEAMCASE$. Let $query$ and tc be the instances of $QUERY$ and $TEAMCASE$, respectively and let N be the number of specified members in q:

$$\begin{aligned} Similarity(query, tc) = \frac{1}{2}(&\min_k(Similarity_{TEV}(eq, ctx_k)) \\ &+ \frac{1}{N}\sum_{l=1}^{N} Similarity_{TMV}(q_l, mc_l)). \end{aligned} \tag{4}$$

where we assume that the set mc is already ordered such that the l^{th}-element of mc is the most similar team member to the l^{th}-element of q. In the case where the members specification is not given by the user, the similarity will be measured only with the first function and the division by 2 will naturally be neglected. Note that the $min()$ function is used since the distance measure regards 0 as the maximum similarity.

5.4 Dependencies as Features

As already mentioned in Section 4, we use the *dependencies* of the meta-model as special features. The similarity of these is computed separately because they are part of domain knowledge and the values are acquired through a particular function. A typical example is the dependency of outputs on the diversity of member variables. Harrison and Klein [12] introduce types of diversity: *separation*, *variety*, and *disparity*. Separation reflects the situation where the team members stand oppositely on a task- or team-relevant issue, e.g., their preferences in development tools. Team member differences in kind, source or category of relevant knowledge or experience reflect variety, e.g., variety of team members' nationality. Finally, disparity shows the dominance of a (part of) team member(s) among the others, e.g. one member is superior to the other team members in number of research publications. The operationalizations of these diversity types are also contributed in the work and will be used as function example in our proposed approach.

The utilization of the values resulting from these functions will then depend on the concrete configuration of domain-knowledge in the recommender system. If a threshold function is known (we know for example, that a higher value is better), an asymmetric similarity function will be chosen in order to maximize/minimize the value of the dependency function to be close to the threshold value. Otherwise, a symmetric similarity function will be applied. The dependency values of *cases* with best outputs/outcomes are used as the similarity target. Thus, we can rewrite the formalization. Given n features on which the outputs are dependent, the similarity function in equation 1 between two $TEAMCASESs$ tc_a and tc_b with respect to the dependencies can be written as:

$$Similarity_{DEP}(tc_a, tc_b) = \frac{\sum_{i=1..n} w_j * sim_i(dep_i(tc_a), dep_i(tc_b))}{\sum_{i=1..n} w_i} \tag{5}$$

where dep_j defines the dependency function of feature j and w_j gives the importance weight to the dependency function. For example, the diversity of member ages in a team belongs to the type *separation* and can be for instance computed with the *Standard Deviation*. Given N members in a $TEAMCASE$ tc, the standard deviation is written as:

$$dep_{age}(tc) = \sqrt{\sum_{k=1}^{N}(mc_{age}^{k} - \overline{mc_{age}})^2/N}. \tag{6}$$

5.5 Improving the Ranking Mechanism

A user would normally expect a list of ranked recommended teams instead of a single result. Ranking by similarity is not always satisfactory in team recommendation: First, the basic idea of case-based team recommendation is to suggest solutions for a given project by selecting the best teams that have performed similar tasks in the past. The most similar teams resulting from the similarity assessment however do not necessarily represent the best teams. If the difference between similarity levels of two teams to the given query is very low, it is not immediately clear which team might perform better. As an additional heuristic in this kind of situations, we propose to take the outputs/outcomes into consideration. The outputs/outcomes are used as additional ranking criteria and to support the user's choice.

Second, another problem arises when the outputs are not available: The outputs of a team are incorporated into the *case bases* after the project is finished and analyzed. The process would normally take a long time and in most of the time the $TEAMCASE$s remain in the *case bases* without the outputs/outcomes information. This situation also occurs for newly formed teams. In these teams, members are composed to teams for the case base and have not participated in any projects previously. The proposed approach forms the team by using the similarity assessment between a given member specification and the provided members in the environment (e.g., open innovation network). For both ranking processes of retrieved $TEAMCASE$s and $MEMBERCASE$s, the novelty of ranking can be improved by increasing the diversity of *cases*. Various methods for diversity are already investigated and can for instance be found in [16]. The diversity makes it possible that more different teams or individuals are drawn from the set of *cases* such that they obtain more opportunities to take part in projects.

5.6 Algorithm

We summarize our case-based team recommendation approach as follows:

1. Assume that TEV', TMV', OUT', CON' and dependency functions are already specified. Additionally, features weight in all of the spaces and a similarity threshold (denoted as $simThreshold$) are configured by the user or a domain expert.
2. A user inputs the specification of a project, the optional members' specification, the optional constraints, and specifies $TeamsReturned$ denoting the number of teams should be returned by the system.
3. If constraints have been specified, eliminate *TEAMCASE*s and *MEMBER-CASE*s from the search space which do not satisfy these constraints.
4. Use equation 4 to assess the similarity of available $TEAMCASE$s with respect to the query. The $TEAMCASE$s whose similarity is higher than $simThreshold$ are stored into a return set *tcReturned*.
5. If the number of teams in *tcReturned* is larger or equal to $TeamsReturned$, rank the list with respect to the outputs/outcomes of the teams and return the best rank of the teams within $TeamsReturned$. Otherwise, members can be retrieved and composed by using the similarity assessment between the member specification and the existing members. The members can be saved in a combination space for further combination in new created *team cases*.

The designed steps define the algorithm of our recommendation approach. There are obviously a vast number of interactions and case possibilities that are mostly dependent on the concrete system implementation. As described in 5.4, we could for instance adapt the dependency functions as features into another similarity assessment within the algorithm steps.

5.7 Example

In order to demonstrate the usage of our recommendation approach we will go walk through the previously described configuration and computation steps with the help of an example. The first step consists in configuring the required parameters. Table 1 shows the observed variables in this example and their corresponding similarity measures. The *case* examples are created adequately for helping the comprehension:

$MEMBERCASE$:

$mc_1 = (25, m, \{\text{java, photoshop}\}, 3)$
$mc_2 = (27, f, \{\text{java, python, network}\}, 2)$
$mc_3 = (34, m, \{\text{progr., network}\}, 2)$
$mc_4 = (28, m, \{\text{java, python, network}\}, 5)$
$mc_5 = (32, f, \{\text{photoshop, flash}\}, 6)$
$mc_6 = (33, f, \{\text{web design}\}, 6)$
$mc_7 = (40, m, \{\text{machine learning}\}, 15)$
$mc_8 = (19, m, \{\text{java}\}, 1)$
$mc_9 = (29, f, \{\text{progr., web design}\}, 5)$
$mc_{10} = (37, m, \{\text{artificial intelligence, robotic}\}, 10)$

Table 1. Example of features configuration for *computer science* teams

Entity	Variable	Feature Type	Similarity Measure
TMV	age	Numeric	Absolute Distance
	gender	Nominal	Conceptual
	competences	Set	Levenshtein
	experience	Numeric (asym.)	Absolute Distance
TEV	project type	Hierarchical	Levenshtein
	project length	Numeric	Absolute Distance
	communication type	Hierarchical	Absolute Distance
OUT	manager rating	Numeric (asym.)	Absolute Distance

$$
\begin{aligned}
&TEAMCASE : (mc, (ctx, out))\\
&team_1 = (\{mc_4, mc_{10}\}, \{((\text{algorithm implementation, 3, irregular meeting}), 8.5)\})\\
&team_2 = (\{mc_2, mc_5, mc_1\}, \{((\text{website, 2, office}), 9.5), ((\text{website, 5, virtual}), 9.0)\})\\
&team_3 = (\{mc_7, mc_{10}\}, \{((\text{paper publishing, 6, virtual}), 9.0)\})\\
&team_4 = (\{mc_4, mc_6\}, \{((\text{web portal, 8, meeting}), 9.0), ((\text{website, 1, virtual}), 8.0)\})\\
&team_5 = (\{mc_8, mc_9, mc_7\}, \{((\text{web portal, 12, meeting}), 10.0),\\
&\qquad\qquad ((\text{web portal, 6, office}), 10.0)\})
\end{aligned}
$$

A query for the recommendation is for example:

$$
\begin{aligned}
&QUERY : (q, eq, con)\\
&query_1 = (\{([25, 200], \{m, f\}, \{programming\}, [1, 100]),\\
&\qquad ([25, 200]), \{m, f\}, \{webdesign\}, [2, 100])\}, (newsportal, -, -), \emptyset)
\end{aligned}
$$

We implemented our algorithm and created a set of feature value spaces for the set, hierarchical, and nominal features. Additionally, we considered all features as equally important (weight = 1). The result of similarity assessment between the query and *team cases* through equation 4 is as follows: According to the algorithm, the set of teams being recommended consist of $team_2$ (Sim=0.184), $team_4$ (Sim=0.151), and $team_5$ (Sim=0.109) for $TeamsReturned = 3$. The ratings (outputs) could be used to support a simple rank mechanism which then ranks according to the output: $team_5$ (Output=10), $team_2$ (Output=9.5), and $team_4$ (Output=9).

6 Evaluation

6.1 Motivation

Principally, two interesting research questions can be distinguished for the presented team recommendation approach. First, it can be analyzed whether an algorithmic approach (in general) facilitates a project manager so that it makes a composition among a large set of open innovation network members faster and less challenging [4]. Second, the question is, whether the quality of recommendations is comparable to the quality of recommendations of a domain expert. While the first question can be answered through the application of an arbitrary recommendation algorithm existing so far (an arbitrarily defined algorithm can recommend a team among the set of *all* possible teams faster than a project manager; the quality of the recommendation may however turn out to be low), the second research question is rather challenging to evaluate. In that sense, we want to evaluate the quality of our recommendation approach given that the application of an algorithmic approach reduces the complexity of a team composition for project managers in large-scale communities.

The evaluation of the proposed approach requires that we cope with some difficulties. First, the composed teams can only be verified to match a particular problem within a context, if they perform the task in a real-world-scenario. The performance can then be evaluated and the outcomes be retained to our *case bases* afterwards. Moreover, both the number of *cases* and new problems needed for the evaluation have to be large, in order to produce good recommendations.

6.2 Evaluation Approach

We attempt to evaluate the proposed approach in a qualitative way aided by a domain-expert. We use data collected in an experiment conducted by the Chair of Cooperative Systems at Technische Universität München involving students of Computer Science. In the experiment [9], a number of students (2-4 persons) worked on a shared workspace with their team and tried to find as much as possible qualified ideas for a given topic. Idea generation fits very well into our domain, since this is an important phase in all innovation processes (i. e. also open innovation processes). The idea behind our evaluation is that we create a task similar to the above mentioned experiment and try to recommend the teams that took part in the afore experiment for a new, very similar task. An expert in the field of computer supported creativity will then create another ranking list, in order to determine whether our algorithm works as the expert expects it to do. In this activity, the expert estimated the best teams by the means of her experiences and prior knowledge. She compared both task descriptions and considered the different output/outcomes of the teams in the previous task performed during the experiment. The expert works in the same fashion as a case-based recommender. Thus this experiment can show if our approach produces rational results.

The first step is comprised of gathering substantial variables of the teams and of the context of the experiment. This data serves as an example for a creative situation, in order to configure the composition model. The variables which have to be considered were acquired through an interview with the domain-expert. Table 2 presents the configuration of variables used in the experiment. Not all variables to which the expert allocated importance had been accumulated during the data collection for the previous experiment. We, nevertheless, present the variables to indicate their importance in a team recommendation for this particular task (creativity task). We categorize the variables into those aspect entities which represent the components of the meta-model for team recommendation. For some variables with hierarchical feature types we prepare ontology structures together with the domain-expert. The features also consist of values resulting

Table 2. Features configuration for the experiment

Entity	**Variable**	**Feature Type**
TMV	Age	Numeric
	Gender	Nominal
	Hobbies	Set
	Competencies	Not available
	Occupation	Hierarchical (Ontology)
	Experience with creativity technique	Nominal (boolean)
	Experience with CSS**	Nominal (boolean)
	Hours in front of PC (weekly)	Numeric
	Fun in the work	Nominal
	Domination	Not available
TEV	Communication media	Hierarchical (Ontology)
	Task urgency	Not available
	Topic of Task	Hierarchical (Ontology)
	Level of Creativity in Task	Nominal
	Creativity technique	Hierarchical (Ontology)
Dependencies	Team size	Numeric*
	Gender Variety	Numeric*
	Age Variety	Numeric*
	Heterogeneity of Hobbies	Numeric* (Nominal Set)
	Interference	Not available
	Heterogeneity of Occupation	Not available
OUT	Idea Quantity	Numeric
	Creativity of Ideas (Median)	Numeric
	Creativity of Ideas (Average)	Numeric
	Practicability of Ideas (Median)	Numeric
	Practicability of Ideas (Average)	Numeric

*values result from dependency functions
**Creativity Support System

from various dependency functions [12]. Similar to the example in Section 5.7, the similarity measurement is performed to produce the recommendation. Some variable names in the table may not be intuitively clear. In this configuration, creativity support system denotes a system that supports the creative problem solving process. We gain the values for *fun in the work* from the questionnaire filled in by the participants of the experiment. This variable defines whether they enjoyed participating in the experiment. The next considerable aspect is *domination* of a team member in an experiment session. A member could, for instance, contribute to more than half of the ideas generated by the team. We did not include this variable because there are no established metrics for it. However, given appropriate metrics, this variable may improve the quality of the result. *Inference* defines the level or existence of influence of team members on others during the experiment. The domain-expert is convinced that this variable can show how a team member affects the performance of other team members.

Neither have we discovered an appropriate formalization for this concept and have, hence, not included it in the calculation.

6.3 Evaluation Result

Table 3 presents the results of our recommendation in the experiment with respect to the absolute distance from the domain-expert ranks to our recommended ranks (normalized into the range $[0, 1]$). The average error values are presented at the bottom of the table. We experiment with the rank mechanism in the involvement of produced *outputs*. The column **Rank** defines the original rank proposed by the domain-expert. The ranks resulting from our case-based recommendation are presented in two columns. **No-Output** shows the result of similarity assessment without the *outputs* information. **Output** includes the *outputs* features in the similarity measures. **Error NO** and **Error O** show the error of the predicted rank without and with respect to the *outputs*, respectively.

Table 3. Evaluation result: comparison of ranks generated by domain-expert and by the case-based recommender

Rank	Team	No-Output	Output	Error NO	Error O
1	Group 18	3	20	0.087	0.826
2	Group 21	14	1	0.522	0.043
3	Group 3	16	5	0.565	0.087
4	Group 5	11	3	0.304	0.043
5	Group 8	21	15	0.696	0.435
6	Group 6	7	8	0.043	0.087
7	Group 1	23	11	0.696	0.174
8	Group 11	1	9	0.304	0.043
9	Group 27	5	2	0.174	0.304
10	Group 25	20	6	0.435	0.174
11	Group 4	22	7	0.478	0.174
12	Group 12	15	17	0.130	0.217
13	Group 13	6	19	0.304	0.261
14	Group 24	18	23	0.174	0.391
15	Group 7	9	12	0.261	0.130
16	Group 14	10	16	0.261	0.000
17	Group 10	12	10	0.217	0.304
18	Group 16	19	24	0.043	0.261
19	Group 9	13	4	0.261	0.652
20	Group 20	8	22	0.522	0.087
21	Group 17	4	14	0.739	0.304
22	Group 2	2	21	0.870	0.043
23	Group 19	24	13	0.043	0.435
24	Group 15	17	18	0.304	0.261
Average error (of 24 groups):				0.351	0.239

The configured variable OUT' consists of "creativity" and "practicability" of ideas produced by the teams which are common measures in creativity research. These values were assigned by a group of evaluators at the end of the experiment. From the table, it becomes noticeable that the *outputs* play important roles in the rank suggestion process. By considering the *outputs*, the average error values decreased. This shows that the expert might have involved them in his approach and therefore, shows the necessity of the proposed additional ranking heuristic described in Section 5.5.

The evaluation shows encouraging results since the resulting ranks are comparable to the ranks of the domain-expert in this specific kind of projects. The ranks shown in the column **Output** include for example 7 teams that are suggested by the domain-expert in the first best 10 teams. It indicates that the recommender sufficiently covers the important variables required for a team recommendation process. The *case* features consist of both properties of the projects and the individuals. Among the features, the real values resulting from dependency functions have a significant portion. It can thereby shown that random variables such as diversity of ages and team size are correlated with the output variables and motivate further focus in this part of the recommendation algorithm. These outputs of the teams in past projects moreover help the case-based recommender in characterizing good teams for future similar projects.

Few significant differences in the ranks can be caused by the following problems. Firstly, all feature types had the same weight. The variables might however differ significantly in their importance for a team composition. Another problem lies in the availability of the data. The whole data only came from one experiment (a single type of project). Better results can be obtained if the number of available projects in which the teams already participated is sufficient. In this case, the similarity assessment will gain more information from a number of *team external variables*, namely the variables representing the experiences of the teams in various past projects. Furthermore, the similarity metrics used for the similarity assessment were quite simple. However, the average error values are in our opinion still in the compromised range such that pursuing this approach seems to be reasonable despite the limitations of our evaluation (e.g. single type of problem).

7 Conclusion and Future Work

We propose a case-based team recommendation approach regarding the typical practice for composing a team where an experienced project manager composes her best teams based on prior knowledge and experiences. The case-based approach assumes that a project can be done by the teams who performed well in previous similar projects. The approach is built on the top of a meta-model which covers essential aspects that have to be considered for building a team recommendation system. In order to cope with the complex structure of a *team case*, we built a specific case representation and its similarity measure. Finally, we also considered rank mechanisms to ensure the quality of recommendation results.

A first evaluation indicated that case-based recommendation on the top of the described meta-model is a promising approach since first, the meta-model guarantees that different strategies for team compositions can be instantiated such that the main problems dimension of the team composition problem can be regarded in contrast to related work. Second, the case-based algorithm seems to resemble quite well the usual best-practice of the domain-expert when she composes teams.

There are, however, a number of shortcomings in our case-based approach. Firstly, the approach provides good results given that the past *cases* are adequate. This means that the system is affected by the cold start problem. The MEMBERCASE can be used for that, but it then requires the specification of members in the user query. Secondly, the performance or quality of a CBR approach is highly dependent on the similarity assessment applied in the *case* retrieval process. This problem is related to the necessity of configuring the feature types, the value spaces, and their weights. The configuration obviously requires a particular level of knowledge and experience. Currently, this is probably performed by managers. This corporate knowledge of configuration can easily be lost and therefore, needs to be archived for future manager. This considerable load for the configuration motivates for further solutions.

In order to improve the proposed approach and to overcome part of the identified shortcomings, the following future works come into our consideration: (i) A non-conventional *case-base structure* can be implemented for optimizing the *case* retrieval and supporting the similarity assessment. The *cases* can e.g. be structured as a search tree in the case repository according to similarity measures of specific features such that similar cases can be found with less effort. (ii) We intend to look for a suitable approach for weighting features in the domain of team recommendation. In this domain, the approaches could be limited by the long process of the outputs/outcomes acquisition. A conversational recommendation (e.g., [15]) is also difficult to implement because the user, for example in an Open Innovation Network, does not have any knowledge of the other community members. (iii) Related to the second point, we would like to investigate the possibility of estimating the outputs/outcomes of a new composed team.

References

1. Aamo, K.S., Ahn, J., Budalakoti, S., DeAngelis, D., Fullam, K.K., Jones, C.L.D., Sui, X.: Agent trust evaluation and team formation in heterogeneous organizations. In: AMAS 2007: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1–2. ACM, New York (2007)
2. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Commun. 7(1), 39–59 (1994)
3. Avramenko, Y., Kraslawski, A.: Similarity concept for case-based design in process engineering. Computers & Chemical Engineering 30(3), 548–557 (2006)
4. Brocco, M., Groh, G.: Team recommendation in open innovation networks. In: RecSys 2009: Proceedings of the third ACM conference on Recommender systems, pp. 365–368. ACM, New York (2009)

5. Brocco, M., Groh, G., Kern, C.: On the influence of social factors on team recommendations. In: Proc. Second International Workshop on Modeling, Managing and Mining of Evolving Social Networks (M3SN), 26th IEEE International Conference on Data Engineering, Long Beach, CA, USA (March 2010)
6. Chesbrough, H.: Open Innovation: The New Imperative For Creating And Profiting From Technology. Harvard Business School Press, Boston (2003)
7. Colucci, S., Di Noia, T., Sciascio, E.D., Donini, F.M., Piscitelli, G., Coppi, S.: Knowledge based approach to semantic composition of teams in an organization. In: SAC 2005: Proceedings of the 2005 ACM symposium on Applied computing, pp. 1314–1319. ACM, New York (2005)
8. Cunningham, P.: A taxonomy of similarity mechanisms for case-based reasoning. IEEE Trans. Knowl. Data Eng. 21(11), 1532–1543 (2009)
9. Forster, F., Frieß, M.R., Brocco, M., Groh, G.: On the impact of chat communication on computer-supported idea generation processes. In: Proc. of the First International Conference on Computational Creativity (ICCC X), Lisbon, Portugal (January 2010)
10. Gaston, M.E., des Jardins, M.: Agent-organized networks for dynamic team formation. In: AAMAS 2005: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, pp. 230–237. ACM, New York (2005)
11. Gonçalves, B., Lopes, A., Esteves, S., Botelho, L.: Cognitive agents based simulation for decisions regarding human team composition. In: AAMAS 2006: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems, pp. 65–72. ACM, New York (2006)
12. Harrison, D.A., Klein, K.J.: Academy of management review- forthcoming, what's the difference? diversity constructs as separation, variety, or disparity in organizations (2007)
13. Leake, D.B., Maguitman, A.G., Cañas, A.J.: Assessing conceptual similarity to support concept mapping. In: Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference, pp. 168–172. AAAI Press, Menlo Park (2002)
14. Leake, D.B., Remindings, R.F.: Cbr in context: The present and future, pp. 3–30. MIT Press, Cambridge
15. Nguyen, Q.N., Ricci, F.: Conversational case-based recommendations exploiting a structured case model. In: Althoff, K.-D., Bergmann, R., Minor, M., Hanft, A. (eds.) ECCBR 2008. LNCS (LNAI), vol. 5239, pp. 400–414. Springer, Heidelberg (2008)
16. Smyth, B.: Case-Based Recommendation. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 342–376. Springer, Heidelberg (2007)
17. Turek, P., Nielek, R., Wierzbicki, A., Hupa, A., Datta, A.: Learning about the quality of teamwork from wikiteams. In: Second IEEE International Conference on Social Computing (Socialcom 2010), August 20-22 (2010)

Toward a Behavioral Approach to Privacy for Online Social Networks

Lerone D. Banks and S. Felix Wu

The University of California at Davis,
One Shields Ave. Davis, CA 95616
ldbanks@ucdavis.edu
http://dsl.cs.ucdavis.edu

Abstract. We examine the correlation between user interactions and self reported information revelation preferences for users of the popular Online Social Network (OSN), Facebook. Our primary goal is to explore the use of indicators of tie strength to inform localized, per-user privacy preferences for users and their ties within OSNs. We examine the limitations of such an approach and discuss future plans to incorporate this approach into the development of an automated system for helping users define privacy policy. As part of future work, we discuss how to define/expand policy to the entire social network. We also present additional collected data similar to other studies such as perceived tie strength and information revelation preferences for OSN users.

1 Introduction

The current social shift of Internet–where users share information that is more representative of their offline identity such as their interests in music, movies, and travel, their locations, where they shop, their moods and feelings, etc.–is responsible for allowing users to have new, individually tailored web experiences. Sites such as Facebook (where user data is stored) become responsible for both managing how users control access to their information and how user information is shared with other sites in order to facilitate user engagement while protecting user privacy. As recent headlines suggest [5,4,24,14], this is a significant endeavor as social sites attempt to provide users with mechanisms for managing privacy policy, with mixed results [13,18].

The results of the current challenges in privacy fall into roughly three categories, user apathy, user confusion, and user revolt. Research has shown, that users do not bother modifying the default privacy settings despite their concerns about how their information is shared [12]. Consequently, users often make themselves vulnerable to having their information shared in unauthorized ways such as sharing party pictures with potential employers. Also, as the number of data objects shared by users increases as well as the number of entities (other users, applications, sites) with whom users may share their data, it becomes less feasible to manually manage access. Further compounding the preceding issues are the diverse policies for data collection and sharing of the various

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 19–34, 2010.

sites that host or request access to user information. As a result, users may have expectations for privacy that are not actually realized by the systems that they use. This is demonstrated when 56% of users incorrectly responded when asked about whether their information stored on one site was shared with third-parties [1]. The consequences of these challenges in access control are that users are often confused about how their information is shared and how to control sharing in a way that is representative of their intentions, leading to significant problems ranging from minor embarassment to catastrophies such as loss of employment [16,5].

Finally, as users become fed up with mismanagement of their data and a lack of simple controls they seek to leave social networking sites.

Our work attempts to address these challenges in managing information revelation for users by utilizing an intuitive concept for which they are already familiar – their relationships. Achieving this goal requires an understanding of the features of relationships as well as the implications of those features on user privacy preferences. Facebook has a rich set of relationship data from which we can make correlations between the attributes of relationships between network entities (users, data, and applications) and patterns of access. Information revelation based on the attributes of relationships could potentially represent the next step in access control for the Internet as the standard approach of manual, static, rule-list based mechanisms begin to reach the limits of their utility.

Facebook provides multiple avenues for users to interact (connections, chat, groups, etc.). We are exploring how social sites can balance user engagement with privacy by leveraging the information contained in these sites, representing the digital manifestation of relationships among subjects, to control access to information in a way that reduces management overhead because the policies are determined by network behavior. Perhaps this concept is best explained by a simple example. Consider the situation when one person, Anita, is asked by two people, Brandon and Cindy, for a money loan. Bob is a relative stranger, an associate that Anita only knows by name. Cindy is a childhood friend with whom Anita has a longtime positive relationship. If Anita can only afford to give money to one of the two people, she is more likely to choose Cindy based on the attributes of their relationship. This is a very simple example for exposition that illustrates our primary concept: ***Characteristics of a relationship between entities can be used to make decisions about access to information and resources.***

One of our goals is to collect data about the correlation between the features of a relationship and the information exchanged within that relationship and use what we observe to make suggestions to users about their privacy settings. In order to reach this goal, we focus on social network analysis (SNA) at the micro level; determining whether, given an individual's local view of their social network, there is a correlation between features of their relationships and information revelation. We also take a take macro level view by looking at the general behavior of a sample of Facebook users in terms of how they use Facebook features, set privacy policy, and their preferences for sharing, and attempting to extrapolate our observations. To do so, we are pursuing the following two subgoals.

Identify a relevant set of relationship attributes that impact user privacy policy.
Existing research [20,1,12,17,19] identifies some of the relationship factors that determine privacy policy. These factors include relationship context, mutual relationships, mutual interests, properties of the data being requested, quality of the relationship between the data owner and requestor, and the context of information access requests. We will begin with these while retaining the flexibility to evaluate the inclusion or removal of factors as appropriate based on the analysis of real data. It is not clear how to quantify all of these factors, for example there does not exist a unit of measurement for the closeness of a relationship between a pair of subjects. As a result, we identify a second goal.

Translate those relevant attributes identified from the previous goal into measurable values that can be collected from the relationship data of today's social networks.
Once we have determined a set of relevant attributes, we must determine how they can be quantified. The challenge of deriving meaning from social data represents an open area of research in this domain. To our knowledge there has only been one attempt [10].

In our case, we seek to analyze social data–this includes user profile information, interaction habits, object meta data (e.g. photo tags)–and map that data to factors that affect information revelation. We can validate our mapping by observing user-defined privacy settings (where privacy settings are user-defined privacy policies) and verifying whether our interpretation of social data corresponds to user policy.

Our social behavior based approach extends existing research that attempts to use social links to enhance applications such as [22,15,9]. This work relies on the existence of social relations and their inherent trust to provide additional network services. Our work seeks to complement these efforts by using the attributes of social relations as factors in making distinctions among ties in the network.

Unfortunately due to the relatively short time OSNs as we know them now have been used widely, there is little information about how users actually use them or their perceptions about their activities within the networks. For example, research suggests that for most users the concept of 'friendship' online differs significantly from friendship in reality [7]. It is still an open question to understand this difference and its implications. We expect that our work will offer some insight into this question, although we focus on understanding the association between relationship attributes and user privacy preferences.

The remainder of this paper is organized as follows. Section two describes our data collection methods including the design of our survey and sampling procedures. In section three we present an analysis of the data that we collect. Section four contains relevant background concepts and related work. In section five we summarize our conclusions and suggest future work in section six.

2 Data Collection

We have developed an online survey for Facebook users to gain insight about the relationship between user behavior, OSN manifestations of relationships, and privacy preferences. The primary purpose of the survey is to determine the correlation between interaction intensity and information revelation. Additionally, we will explore the relationship between perceived tie-strength and information revelation.

Our 'survey' consists of two parts: supplemental data collection and traditional participant questioning. The first is demographic, usage, and interaction data collection. For each participant we collect basic demographic information such as name, network membership, and education level. We also collect the participant's privacy settings of their data objects (e.g. wall posts, notes, etc.) and other usage information such as group memberships, event attendance, friend lists (user defined groupings of their ties), and their local social network. We also collect the type, date, and ids for the ties involved of interactions between the participant and their social network. The traditional aspect of our survey is in the form of four questions that the participant answers about at most 10 select 'friends.'[1] The process for how those 'friends' are selected and a summary of the data that we collect is provided below. The details of our collection polices and practices are provided to each participant before taking the survey.

Overview of six types of data that we collect:

- Supplemental data[2]–demographic, usage, and interest information about the participant and their local social network.
- Interaction data[2] –the frequency of various interaction types between a pair of Facebook users (e.g. Cherie sent 1000 messages to Lerone; Lerone 'liked' 40 of Cherie's objects).
- Privacy settings–access settings for the data objects owned by the participant defined as access control lists (ACLs)(e.g. Cherie allows only her friends to see her photos; Lerone blocks 'his mother' from seeing a video that he created).
- Privacy preferences–for a pair of Facebook users from the participant perspective, a list of information types that the participant is willing to share with or receive from a particular friend (e.g. Lerone would share his home phone number and address with Cherie).
- Type of relationship–a classification of one or more types associated with the tie between a pair of Facebook users from the participant's perspective(e.g. According to Lerone, Lerone and Cherie are friends and former classmates).

[1] Friend is in quotes because the subject of a survey question may not be a direct tie of the participant in the OSN social graph (e.g. the subject is not a friend of the participant).

[2] Visibility of objects,data, or interactions not owned by the participant may be restricted by privacy settings of the owner (either settings for the participant or settings for what the owner allows their friends to share with applications).

– Perceived relationship quality–perceived tie-strength from the participants perspective of a given tie (e.g. According to Lerone, the strength of the relationship between Lerone and Cherie is 100 on a scale of 0-100).

To select the participant's 'friends' for which to ask our survey questions, we group 'friends' based on the number of interactions between each 'friend' and the participant, relative to the total number of interactions.

We choose randomly one 'friend' from each group until a maximum number of 'friends' are selected. For each participant the 'friend' groupings will be based on either the count for a randomly selected interaction type (e.g. wall posts) or the cumulative total count of all of the participant's interactions.

For each participant we count the number of initiated conversations, received wall posts, and photo tags[3] that occur between the participant and each of their friends. We then group the participant's 'friends' by their percentage of the total number of interactions. We randomly sample 'friends' from each group and ask the participant to answer two privacy related questions. The first is about which types of data the user wants to share *with* the 'friend' and the second question is about which data they desire to receive *from* the 'friend.' For both questions, the participant fills in a checkbox for each type (out of 18 possible types) of data that she is willing to share with or receive from the given 'friend.' The 18 possible types fall into roughly three categories: Facebook specific data–data maintained and collected within Facebook such as status updates and uploaded photos, Personally Identifying Information–such as home address or work history, Other–data that does not fit into the preceding two categories but often has its own privacy implications such as GPS location, political views, or past purchases. Our survey includes an additional question about the participant's perceived relationship strength with a particular 'friend.' For this question, the participant is asked to move a horizontal slider along a scale from 'weak' to 'strong.' The final question asks participants to classify the type of relationship between the participant and a selected 'friend' by selecting one or more of seven checkboxes corresponding to different relationship types ranging from 'complete stranger' to 'spouse/partner.' The actual questions of the survey are included in the appendix.

Data collection is ongoing for future work and the survey can currently be taken by anyone with a Facebook account by visiting `http://apps.facebook.com/afne_survey/`.

2.1 Survey Discussion

Having presented the details of our survey, in this section we explain some of the motives for our design choices, attempt to address potential areas of bias, and discuss challenges to creating user surveys in this domain.

Our survey is both multi-dimensional and dynamic. It is multi-dimensional in that it collects self-reported data via survey questions and also gathers behavioral and demographic data for user Facebook accounts. Our survey is also dynamic in

[3] These interactions are defined in the next section for those unfamiliar with Facebook.

that the questions are created in real-time based on information collected from the survey participant. Because of the random selection of 'friends' that are the subject of survey questions and the constant flow of interactions, a particular participant may be presented with a different set of 'friends' if they were to take the survey more than once[4]. Behavioral and demographic data are collected from a source beyond our control (Facebook) that changes rapidly in terms of the information that is available as well as the means for collecting it. One effect of surveying in such an environment is that our programs for collecting data must be developed in such a way to quickly adapt to changes in Facebook's platform. Another effect is that our data collection is less than ideal. Due to limitations of the Facebook platform and security features we had to carefully design our survey with an understanding of the implications and limitations of the data that we are able to collect.

Interaction Data. Facebook provides numerous types of interactions that seems to increase regularly. We based our selection of which types of interactions to study on three criteria:

- available to most facebook users,
- reasonable indicator of relationship quality,
- sensible correlation with privacy.

We collect data for nine interaction types. Based on the above we focus our analysis on the following three types:

'Initiated conversations' refers to private messaging threads that are initiated by the participant. We count entire threads instead of individual messages because we believe that the number of conversations is a stronger attribute than the length of conversations. Counting individual messages would obscure this distinction. Likewise we chose to focus on conversations initiated by the user to limit the inflationary effect of message senders whereby an arbitrary sender can artificially boost their status with a particular user by frequently sending messages to that user.

Received wall posts are the messages displayed on the 'wall' of the participant's profile. While they are subject to inflation from other users, we believe that this is less likely to occur as wall posts are (by default) prominently displayed on a user's profile and therefore subject to stronger monitoring by the user or her friends. This more strict monitoring makes frequent wall posts a good indicator of relationship quality assuming that if a user allows a wall post to remain on her profile it implies that she values the post and likely the friend making the post. We recognize that there are factors that could invalidate this assumption, such as infrequent account usage that causes a user to overlook new

[4] Although we allow users to take the survey repeatedly, only a single set of results are counted for a particular user.

wall posts. Since we plan to have a relatively large sample size and we are seeking only a baseline we expect the effect of those cases to be negligible.

Photo tags are the co-occurrence of a user's name with her friends' names associated with a particular photo. Photo tags typically indicate at the least that a pair of friends were in the same place at the same time when the tags are associated with people in the photo. Friends do tag each other in photos without one or both of them appearing in the photo to signal that the tagging friend wants the tagged user to view the photo but we believe that this practice is somewhat less common. Also, similar to wall posts, photos and their tags are public by default so we assume that not removing a tag implies a positive feeling about the friends tagged in the photo. Under this assumption, a pair of friends that are frequently tagged together in photos indicates ongoing interaction between the pair.

We do not intend to imply that we have selected the best or even a complete set of interactions. We think they strike a reasonable balance between being meaningful and the cost to observe them. Based on preliminary observations, photo tagging is probably the least reliable measure because of tagging habits, for example, if the owner of a photo posts a picture, they often do not tag themselves in the picture. While it means something if a pair of users actually appear in a picture together that has been tagged with their names, tagging is optional and it is non-trivial to determine whether the tagged names actually match people in the photo. Also, privacy settings imposed by the owner of a photo may limit what is countable by our application. While counting inbox threads is sound in theory, in practice it may be weak because of the usage patterns of messaging within Facebook. Although the functionality of Facebook messaging is increasing it is quite limited and nowhere near as ubiquitous as email. One possible future alternative is to map email addresses to the Facebook accounts of a participant and her friends and observe email interactions. Currently we constrain our data collection to what is observable within Facebook.

Currently, our data collection process does not account for time between interactions. There are two reasons for this. First, we are unaware of time constraints imposed by Facebook on available data, so we collect all that we can. Second, we do collect the date and time of each interaction so that we can incorporate time as a factor in future analysis. We acknowledge that time is a significant attribute that will be a factor in our analysis and future system models and designs.

We collect some interactions that are directed toward the user such as 'like' (where one user signals their approval of a data object of another user that is visible to all viewers of the object). For now, we do not see inflation because there is little incentive but we must consider that a system that provides incentive (such as elevated status) based on interactions is vulnerable to inflation if relying on received interactions. A malicious user can artificially elevate their status by 'liking' everything that their target does within the network. We can limit this effect by choosing interactions that are resistant to inflation. Photo tags are generally vulnerable by default but can be configured to be more resistant by only counting photos that have been tagged by the user of focus instead of arbitrarily tagged photos. Wall posts are thought to be somewhat resistant because of the tendency of people to remove junk posts although there is the potential for race conditions with this type.

Sampling. The survey participants were not completely randomly selected. We rely on the viral effect prevalent in many social sites. Several students, faculty, and staff from our department (from several different countries and backgrounds) served as seeds for encouraging survey participation. Each seed spreads a message about the project throughout their social network. As users participate, information about the project propagates among their social networks. While we do not guarantee a truly random sample, we are able to collect data from a sample of users that are diverse along several dimensions. Several studies on Facebook usage have shown that the average number of friends per user is approximately 150 [8]. As it is infeasible to ask about all of a participants friends we attempt to ask questions about a representative subset so we group the friends as described in the previous section and sample randomly from each group. Facebook users that are not direct ties of the survey participant may be selected as the subject of a particular survey question because the participant allows interactions from non-directly connected users. Survey responses of this type offer insight into user feelings about non-directly connected users. Our data attempts to collect user feelings ties of varying quality and interaction frequency. The maximum number of 'friends' that are chosen for each participant (10) is somewhat arbitrary. We sought to keep the number low enough to encourage participation while still sampling a representative set of 'friends'. We attempt to offset this relatively small sample size by non-uniformly selecting 'friends.'

Survey design. Our questions about data the participant is willing to share and is willing to receive attempt to capture the read and write aspects of privacy. With the interaction data and user's privacy preferences for a representative subset of his friends, we expect to identify characteristics of interaction behavior that we can later use to suggest privacy settings for a Facebook user's 'friends.' With a question about relationship quality, we seek to understand in general, user perceptions of friendship in an OSN (Facebook in particular). The results will provide additional data about the relationships being maintained on Facebook such as whether or not users maintain many of their strong relationships within Facebook. Also, data collected from a question about tie strength allows us to observe any correlation between perceived relationship strength and interaction intensity. Likewise we can attempt to correlate interaction types and relationship strength. The question on type of relationship gives us yet another variable to correlation to tie strength,sharing preferences, and interaction behavior. We allow participants to specify multiple types for their ties, in order to capture ties that cross multiple relational dimensions (e.g. a relative for which there is a professional relationship). Since our survey is online we use web forms for data collection. The advantage of this approach is that we can validate collected data before it is submitted is part of our sample. Our survey requires that participants individually select each of the data types that they wish to share with or receive from other users. Several participants and testers complained about this aspect of the survey and requested the inclusion of an 'ALL' button that would allow them to select all types where applicable. We did not comply with this request in order to encourage explicit selection and acknowledgment of their preferences. The benefit of this design decision is that we avoid bias due to default choices. We also require that participants explicitly select 'None' to indicate their unwillingness to share or receive any information from a particular 'friend.' The form is validated programatically to prevent ambiguous responses (e.g. selecting 'none' and some other type or leaving a question blank).

The default value for our question about tie strength is 'weak' which may introduce bias toward lower tie strength. In future versions, we may consider setting the default at mid-scale. Also, in our current design we are unable to discern between selection of the default value and oversight of the question. Our survey underwent several interactions of testing and development including consultation with experienced sociologists familiar with survey design. Every effort was employed to reduce or avoid bias while presenting questions in way that would provide insight for our research goals.

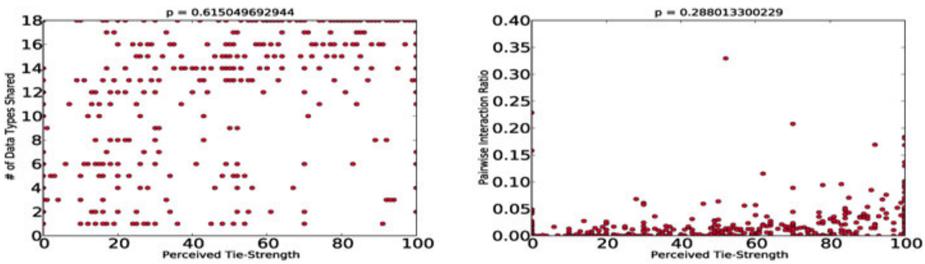
3 Data Analysis

The focus of our analysis is on the survey responses and interaction data collected from 53 survey participants. The participants were self-selected via random walks seeded at the authors and other lab members. The average number of friends that each participant was asked about was eight. There were four questions asked per friend resulting in each participant answering about 32 questions. There are more than 1700 data points of survey responses.

From our 53 participants and their friends, we collected 31,800 interactions across all nine types. There are about 25,500 interactions of the main three types as identified in earlier sections. Nearly half of these main interactions and more than one-third of all interactions are posts to a participants Facebook wall. Out of 446 individual survey responses[5] (we count each set of four questions for a participant-'friend' pair as a single response) the data type, 'status update' is shared most often as it shared in 83% of the responses. The date type 'past purchases' is shared least frequently at 33%. Overall, participants were rarely unwilling to share any information with the 'friends' of which they were asked. Sharing 'no data' was chosen for only 9% of all responses. An interpretation of a general willingness of users to share some information in our sample may be due to the fact that most participants were asked about 'friends' that are direct ties in the social graph which by default have access to at least some of their 'friend's' data in Facebook. The mean number of data types shared by our participants was 11 (out of 18 possible) and sharing all 18 types occurred in more than 25% of the responses. Finally, among our responses, the mean value for perceived tie strength was 49.285 (on a real scale from 0 to 100).

It has been suggested in existing work that tie strength could be used to determine privacy preferences[10]. As part of our study, we examine this hypothesis.

[5] For this analysis we focus on data the participant is willing to share.



(a) TS vs. Sharing (b) TS vs. PIR (main interaction types)

Fig. 1. Tie Strength Correlations

The graphs below show the correlation between self-reported tie strength and data sharing and interactions for the survey participants.

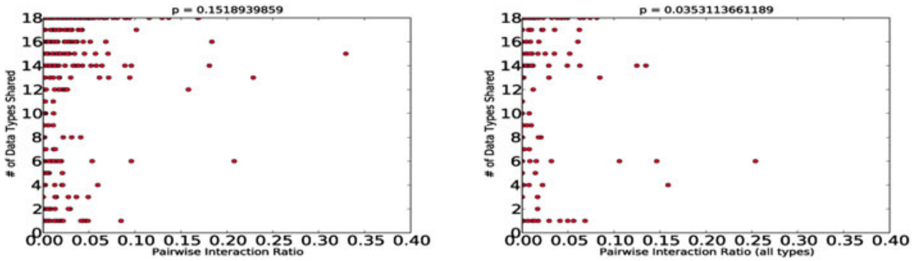
Figure 1a shows a mild linear correlation between user perceptions of tie strength and information sharing. There were several cases where a participant was willing to share all of the possible data types with 'friends' for which they reported zero tie strength. We followed up with those participants to try to understand why this would occur. The common response was that while the participant had no active relationship with the 'friend' in question, hence zero tie strength, the 'friend' was an old acquaintance so the participant did not mind sharing their data. Figure 1b shows a modest positive correlation between tie strength and interaction ratio (defined below).

To observe the correlation between privacy preferences and interaction behavior among Facebook users we generate a graph of the pairwise interaction ratio (PIR) and the number of pieces of information the participant is willing to share between a survey participant and each of the Facebook users for which that participant answered survey questions. The pairwise interaction ratio is defined for participant p, a Facebook user participating in our study and s_p, a Facebook user selected by our program to be the subject of a set of survey questions as:

$$\frac{i_{s_p}}{\sum_{j=0}^{F} i_{j_p}}$$

where i_{s_p} is the number interactions between p and s_p and the denominator is the total number of interactions for all Facebook users that have interacted with p.

We calculate PIRs for only the three standard interaction types as described in section 2 for all survey participants (N=53). We also calculate PIRs over all nine interaction types for a subset of all survey participants for which we were able to collect full interaction data (N=20). The following graphs show PIR vs. number of pieces of information shared for pairs of users.



(a) PIR (main interaction types) vs. Sharing

(b) PIR (all interaction types) vs. Sharing

Fig. 2. Interaction Correlations

As shown in the graphs and subsequent calculations of Pearson's correlation coefficient, based on our data, there exists a small, positive correlation between interaction intensity and information revelation among Facebook users. Figure 2b shows the correlation for the subset of participants for which we collected all nine interaction types.

As shown in other work, interaction volume is low overall and a given user only interacts with a small percentage of her friends [21]. These factors limit the utility of interaction intensity alone for predicting privacy preferences. Our data motivates further exploration of the meaning and value of OSN interactions as well as the inclusion of additional relationship factors such as mutual ties and interests in order to predict privacy preferences.

For this analysis we correlate the number of pieces of information shared with total number of interactions. An issue with using a standard count of pieces of information is that it assumes that all types of information are equally valuable. We accept this assumption for now because to our knowledge there is little work on quantifying the value of information types.

Perhaps this gap in understanding indicates that future privacy systems should allow users to assign values to their data that should be taken into account when making privacy decisions.

Using pairwise interaction ratios for the total number of interactions assumes that all interactions also have equal value. On the other hand focusing on a subset of interactions introduces bias towards 'friends' that utilize a particular interaction type. This bias blurs the observed outcomes in that if a particular 'friend' has high interaction intensity via an overlooked or uncounted interaction type that 'friend' will appear to be a low interaction 'friend.' This reasoning could potentially account for cases where perceived tie-strength and data sharing are high but the interaction ratio is low.

We have developed some intuition about reasoning behind the low linear correlations that we observe. It may be the case that relationships that exhibit the property of low interactions but high sharing are maintained via other channels. This is a common case among children and their parents or employers and employees. Also, the availability (or lack thereof) of complete interaction data (our current view is asymmetric) may skew our results.

At this time we present initial data from our analysis in order to determine the value of interactions in determining privacy preferences. The positive correlations that we see with our limited data and analysis indicate that more work is needed to fully understand interactions and their impact on sharing within online social networks such as Facebook.

4 Background

4.1 Social Science

At least two approaches have been discussed to determine privacy preferences among interpersonal ties: relationship context and relationship quality . Context based approaches such as [6,2,12,17,20,19] take as input the type of relationship or the context in which interactions occur as the basis for categorizing relationships for privacy (e.g. work friends vs. high school friends, army general vs. army private, etc.). This approach does not consider the characteristics of individual relations and assumes that all members of a particular category are equal. It has utility for applications such as chain of command and general communication (e.g. contacting members of one's high school class for a reunion) and offers some granularity to users for making distinctions among their social relations.

Relationship quality based approaches such as those in [11,10] offer potentially more fine grained categorization at the cost of increased complexity. Although significant research suggests factors for measuring relationship quality, the complexity and dynamic nature of humans and their preferences makes it difficult to establish an absolute and objective measure of relationship quality. The problem becomes even more challenging when attempting to measure relationship quality through observation with little or no direct input from the person being observed. After all, what does it really mean to ask, 'How intimate is A with B?' and how generally meaningful is the answer? It seems that relationship quality in general is less ambiguous as compared to context. The question of how to categorize a friend that exists in multiple contexts is challenging. Whereas with relationship quality, the quality may change over time but at any instant the quality of a relationship is understood. On the other hand, relationship quality alone may be too general of a concept for some applications. Currently, the effect and utility of relationship quality on fine-grained privacy is unclear.

Considering our primary goal of relationship-aware access control, we seek to create relationship based groups that are defined by the features of relationships between users. This is an appropriate step toward realizing a system that provides granularity of access control in an automatic way that is currently unavailable in online social networks.

4.2 Related Work

There exist numerous models for specifying privacy polices (RBAC, ABAC, CBAC, ZBAC, UCON, etc.). The issue with these approaches is that while they can be shown to be effective for defining policy they are difficult to use for the average user. Unfortunately the social shift of the web requires that each user define and manage information access policy with the deft of a trained system administrator. These models provide little insight or assistance for users in actually defining their policies thus limiting their application for casual users. Our project builds on the concepts presented in [3,23,10] that suggest the use of relationships for a variety of purposes related to future Internet designs. We narrow the focus of the use of relationships, specifically for purposes of information access control. Much of the work to date in the area of information revelation emphasizes the category of a relationship as a key factory in user privacy decisions. While we also support this observation, we endeavor to extend this idea to incorporate additional social context into the representation of a relationship in order to provide further granularity in helping users make distinctions among their social relations. Several studies exist that examine relationships within Facebook. Our studies differs from those done previously in that we attempt to examine how features of Facebook (and OSNs in general) impact privacy within environments that are designed for data sharing. To do so, we evaluate the relationship between interaction intensity and information revelation, using an approach similar to that taken by researchers at the University of Illinois at Urbana-Champaign [10].

5 Conclusion

Based on our data and early analysis, there exists a small, positive correlation between a measure of interaction intensity and data sharing within Facebook. This correlation indicates that interactions may be useful in determining user privacy preferences. On the other hand, interactions may be limited in their utility because a significant majority of users interact passively (e.g. simply viewing a profile or picture) rather than interacting in some measurable way for which we can attach meaning. The lack of a clear verdict on the utility of interactions particularly in combination with other factors that have been shown to affect data sharing suggests that more work is necessary. While we do not see a linear correlation between interactions and privacy the positive correlation given a limited view of a small data set indicates exploration of behavior as a potentially useful factor in determining privacy policy. We are actively pursuing how to get the most out of this data as well as how to incorporate it with other data that predicts privacy such as relationship type.

6 Future Work

After developing our survey and performing some analysis we see that there is significant work to be done in order to understand OSN interactions and their implications for information revelation. We are continuing to collect data as more people participate in our study. This additional data will give us a broader view of behavior within the network. In order to better understand interactions and their role in user privacy we will use advanced statistical approaches to identify relevant interaction and data types. Also, we will look to get symmetric views of interactions in these networks in order to solidify our analysis. Also, it would be useful to determine whether a particular type of interaction is associated with a particular level of interaction intensity. The answers to these and other analytical questions about user behavior within online social networks will drive the design of future privacy mechanisms by providing additional parameters that can be utilized by such systems.

Acknowledgments. Dr. Diane Felmlee, Mina Duroud, and Juan Lang their invaluable feedback.

References

1. Acquisti, A., Gross, R.: Imagined communities: Awareness, information sharing, and privacy on the facebook. In: Danezis, G., Golle, P. (eds.) PET 2006. LNCS, vol. 4258, pp. 36–58. Springer, Heidelberg (2006)
2. Anthony, D., Henderson, T., Kotz, D.: Privacy in location-aware computing environments. In: IEEE Pervasive Computing, pp. 64–72 (2007)
3. Banks, L., Wu, S.F.: All friends are not created equal: An interaction intensity based approach to privacy in online social networks. In: IEEE International Conference on Computational Science and Engineering, vol. 4, pp. 970–974 (2009)
4. Barbaro, M., Zeller Jr., T.: A face is exposed for aol searcher no.4417749. The New York Times (2006)
5. CNN.com. Forced to quit over facebook? (2009)
6. Consolvo, S., Smith, I.E., Matthews, T., LaMarca, A., Tabert, J., Powledge, P.: Location disclosure to social relations: why, when, & what people want to share. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 81–90. ACM, New York (2005)
7. Cummings, J.N., Butler, B., Kraut, R.: The quality of online social relationships. Communications of the ACM, 103–108 (2002)
8. Facebook.com. Facebook press room statistics (July 2010), http://www.facebook.com/press/info.php?statistics
9. Garriss, S., Kaminsky, M., Freedman, M.J., Karp, B., Mazières, D., Yu, H.: Re: reliable email. In: NSDI 2006: Proceedings of the 3rd conference on Networked Systems Design & Implementation, p. 22. USENIX Association, Berkeley (2006)
10. Gilbert, E., Karahalios, K.: Predicting tie strength with social media. In: CHI 2009, pp. 211–220. ACM, New York (2009)
11. Granovetter, M.S.: The strength of weak ties. The American Journal of Sociology (1973)
12. Gross, R., Acquisti, A., Heinz, J.H.: Information revelation and privacy in online social networks. In: WEPS 2005, pp. 71–80. ACM Press, New York (2005)
13. Kincaid, J.: Facebook rewrites privacy policy, foreshadows location based services (October 2009), http://techcrunch.com/2009/10/29/facebook-rewrites-privacy-policy-foreshadows-location-based-services/

14. Manor, D.: Eff proposes new bill of rights for facebook users (May 2010), http://www.allfacebook.com/2010/05/eff-proposes-new-bill-of-rights-for-facebook-users/
15. Marti, S., Ganesan, P., Garcia-Molina, H.: Dht routing using social links. In: International Workshop on Peer-to-Peer Systems 2004 (2004)
16. Matyszczyk, C.: Wife exposes chief spy's personal life on facebook (July 2009), http://news.cnet.com/8301-17852_3-10279317-71.html
17. Olson, J.S., Grudin, J., Horvitz, E.: A study of preferences for sharing and privacy. In: CHI 2005: extended abstracts on Human factors in computing systems, pp. 1985–1988. ACM, New York (2005)
18. O'Neill, N.: Infographic: The history of facebook's default privacy settings (May 2010), http://www.allfacebook.com/2010/05/infographic-the-history-of-facebooks-default-privacy-settings/
19. Spiekermann, S., Grossklags, J., Berendt, B.: E-privacy in 2nd generation e-commerce: privacy preferences versus actual behavior. In: EC 2001: Proceedings of the 3rd ACM conference on Electronic Commerce, pp. 38–47. ACM, New York (2001)
20. Tsai, J., Cranor, L., Acquisti, A., Fong, C.: What's it to you? a survey of online privacy concerns and risks. Working Papers 06-29, NET Institute (2006)
21. Wilson, C., Boe, B., Sala, A., Puttaswamy, K.P.N., Zhao, B.Y.: User interactions in social networks and their implications. In: Proceedings of the Fourth ACM European conference on Computer Systems (EuroSys), pp. 205–218. ACM, New York (2009)
22. Yu, H., Gibbons, P.B., Kaminsky, M., Xiao, F.: Sybillimit: A near-optimal social network defense against sybil attacks. In: IEEE Symposium on In Security and Privacy, pp. 3–17 (2008)
23. Zinoviev, D., Duong, V.: Toward understanding friendship in online social networks. International Journal of Technology, Knowledge, and Society (2009)
24. Zuckerberg, M.: An open letter from facebook founder mark zuckerberg (December 2009), http://blog.facebook.com/blog.php?post=190423927130&ref=share

Appendix: Survey Questions

The following are the actual questions asked in the survey.

Question A: What profile information are YOU WILLING TO SHARE WITH Friend X? (Check all that apply).

1. Friends Added
2. Groups Joined
3. Events Attended
4. Relationship Status
5. Full Birthdate
6. Received Comments
7. Work History
8. Personal Phone Number
9. Home Address
10. Posted Links
11. GPS/Exact Location
12. Religious Preference
13. Uploaded Photos
14. Past Purchases
15. Political Views
16. Education History
17. Status Updates
18. Received Wall Posts

Question B: What profile information do you want to see about Friend X? (Check all that apply).

1. Friends Added
2. Groups Joined
3. Events Attended
4. Relationship Status
5. Full Birthdate
6. Received Comments
7. Work History
8. Personal Phone Number
9. Home Address
10. Posted Links
11. GPS/Exact Location
12. Religious Preference
13. Uploaded Photos
14. Past Purchases
15. Political Views
16. Education History
17. Status Updates
18. Received Wall Posts

Question C: What is the strength of your relationship with Friend X?
On a Sliding Scale from 0-100

Question D: Which of the following types describe your relationship with Friend X?

1. Strangers/Unknown
2. Acquaintance
3. Work/Professional/Business
4. Family/Relative
5. Close Friend
6. Classmate
7. Spouse/Partner/Significant Other

A Meta Model for Team Recommendations

Michele Brocco, Georg Groh, and Florian Forster

Technische Universitaet Muenchen
{brocco,grohg,forster}@in.tum.de
http://www11.in.tum.de

Abstract. Teams are an important organizational unit and need to be composed appropriately. Whenever a high number of possible team members exists, the complexity of the composition task can not be effectively handled by humans. To support the composition in this scenario, team recommenders can be used. In this paper we discuss and formalize a flexible approach using a generic meta model for implementing various team composition strategies derived from a literature review. In order to demonstrate its use and its compatibility with a generic team recommendation approach, we then translate some of the theoretical team composition approaches found in the literature.

1 Introduction

A team is a group of people (i. e. at least two persons) working together to fulfill a certain task. In many domains, teams are an important organizational unit for solving problems.

Especially when innovative ideas are sought, the ideation process is commonly executed in a team context[12]. By sharing their knowledge and experiences within the team, the team members create a situation of mutual synergetic benefit [10]. This way, individual weaknesses of a team member can be compensated. Hence, in many cases, the results accomplished by a team synergetically exceed the results that would have been accomplished by the individuals if they would have worked on their own.

1.1 Open Innovation Networks

Traditionally, innovation processes were exclusively associated with a company's R&D department, where dedicated specialists were working on new ideas while being mostly isolated from the rest of the firm.

In contrast to this restricted concept of the innovation process, the Open Innovation (short OI) paradigm is an attempt to extend the innovation process by including internal and **external sources** for innovation, as well as finding new internal and **externals paths** to market [6]. Internal sources can be employees from other departments or from other locations, while external sources can be suppliers, customers or freelancers.

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 35–50, 2010.

To benefit from these new sources, companies inevitably have to open their innovation process, allowing information to flow accross the strict borders of their R&D departments. In turn, this openness provides a fertile ground for collaboration between all actors in the innovation process, leading to active networks of innovators [14].

1.2 Team Recommendation Systems

By opening their innovation process, companies can benefit from the creative input, experiences and the manpower of external sources and therefore have the chance to strengthen their business competitiveness. But the Open Innovation paradigm also implicates new issues such as the question of intellectual property of ideas or pieces of work that were generated outside of the company or further problems tied to the loss of control and awareness over the innovation process.

A good example for the latter type of problems is the assembly of optimal teams. In the traditional case, the potential team members for an innovation project are mostly limited to the employees of the R&D department. Project managers in charge of team composition in general have in depth knowledge of potential team members and therefore are aware of their experiences, their abilities and their personality. This is not the case for open innovation teams: The project managers are not aware of all possible team members available in the OI network. Furthermore, it quickly becomes impossible for a project manager to keep track of history and current status of the vast amount of properties of teams and team members. For these reasons, an IT support for assembling teams in OI networks as well as other large collaboration networks can be beneficial, especially for the initial phases where a core team for a project kick-off has to be composed from a large set of possible team members.

1.3 Meta Models

Teams perform several different kind of tasks or solve different kind of problems. Examples are the development of new products or the generation of new ideas for a specific problem. Even specific instances of one of these examples may substantially differ. As a consequence, a team appropriate for a general type of problem may not be adequate for a concrete instantiation of this problem. This may be due to the variation of possible environments (e. g. virtual teams vs. real teams) that the concrete problem instance occurs in and/or due to the variation of problem instances in correlation to the internal properties (e. g. skills) of the team. This implies that, depending on several aspects, teams may be composed according to different strategies. Given this statement, a comprehensive team recommendation approach must be flexible with respect to the usage of various composition strategies (also known as team composition models) such that a project manager can use a specific team composition strategy that is adequate for his needs and situation.

In order to address this issue we developed a meta model that can be used to model specific team recommendation scenarios including aspects identified in the literature in previous work (section 2.1). The meta model approach has advantages and disadvantages compared to a specifically tailored solution that we want to discuss in the following paragraphs.

As already mentioned, a meta model allows for the instantiation of several team recommendation models including their context and constraints. The advantage resides in the possibility to use meta model components to instantiate several different team composition approaches. Because meta model components are standardized, no matter what specific composition approach has been instantiated, the same recommendation algorithm can be used. In contrast to that, for a specific set of meta model components, several different algorithms can be used to generate a recommendation.

On the other hand, special solutions such as [7] or [5] have also many advantages. First of all, they are tailored for a specific domain and concept. That means, the level of abstraction of special solutions is lower for the specific concepts, while a general approach such as the meta model approach, necessarily has to employ a higher level of abstraction in order to be able to cover many team recommendation scenarios. A high abstraction level may result in imprecise recommendations compared to tailored special solutions.

Implementing and applying several tailored recommender approaches is however time consuming and complicated in domains where the variety of needed different team composition approaches is high. Additionally, training a recommender based on a meta model can be more comfortable because the underlying data structures used for all the different approaches are the same. This means that data of teams that accomplished a task in a specific scenario can be used more comfortably to train the recommender, nearly independently (i. e. given the availability of required team member variables and output/outcomes) from the actual approach used. In the tailored recommenders no common structure of aspect entities is usually defined. Therefore the conversion and adaption of input and output data may be complex.

Finally, extensibility, with respect to team composition strategies, can be achieved easier through a generic solution since new composition approaches can be “assembled” using the components specified in the meta model.

2 Previous Work

2.1 Identification of Influence Aspects

Team recommender algorithms will, in general, use some form of team composition model. Team composition models (either instances of meta models or special solutions) consist of aspect entities that together model a team’s output/outcome e. g. model how well a team with certain properties will perform in a given context. Aspect entities of our meta model, which we discuss below in section 4 in more detail, encompass team member variables, team external variables, output/outcome variables, dependencies and constraints. In [3] we broadly investigated aspect entities, on the basis of an extensive literature review, as a first step towards a meta model for team composition. In view of the development of this meta model, it was the goal to cluster these aspect entities into vertical classes. The first of these vertical classes, team member variables, encompass all variables strictly related to individual team members, such as functional knowledge variables (e. g. skills), demographic, social-network, and psychological variables. Team external variables define the environment and the context that a team acts in, e. g. variables characterizing organizational culture, team organization and processes, rewards, communication and information systems used. Constraint aspect entities exclude certain team constellations from consideration. Dependencies describe how these influence aspects relate to output/outcome of a team. Besides these vertical classes of influence aspect entities, we also considered team intermediate variables, The variables in this vertical class are dependent on variables of other classes and thus are logically dispensable. But they very often occur in team composition models from literature, because they contribute to a human understanding of the model. An example could be variables formalizing the nature, numbers or characteristics of conflicts in a team which can be traced back to more fundamental (e. g. demographic) team external variables.

2.2 Model Congruence Verification

Besides the vertical classes of aspect entities investigated in [3] and describing the different functional roles of variables, also horizontal classes of aspect entities such as socio-psychological variables, skill variables, organizational variables etc. which cluster semantic facets of influence aspect entities were taken into account in our previous work. Using the important horizontal class of socio-psychological variables or aspect entities, we tested the skeleton of our meta model by developing an instance of the meta model based solely on socio-psychological variables in [4]. After selecting relevant social variables from a literature review and our previous work in [3] we included those variables and dependencies that were empirically supported by studies. Furthermore, we formulated and empirically tested appropriate hypotheses with respect to those variables and dependencies that were not yet empirically verified. Our hypothesis that typical team intermediate variables are dispensable could be verified in this study. With this study we supported and further refined our meta model for team composition and recommendation.

3 Existing Psychological and Sociological Models

Psychological and sociological models (e. g. [11],[13],[15]) select configurations of variables and describe dependencies between these variables and a team's output/outcome (possibly mediated by intermediate variables). An extensive review of related psychology and sociology articles suggests that these models vary a lot depending on the environment in and for which they have been developed. In most cases it is not explicitly clear to which extent the suggested theories can be generalized to other environments. Such environments can be described in our nomenclature by team external variables.

When facing a concrete team composition task, a choice of an appropriate model can be a matter of experience, estimation or maybe be subject to trial and error. Often best practices of project managers, HR-personnel or other persons in charge of composing teams outperform sociological and psychological models, because these models are too special or not easily transferable to a concrete problem environment. Such best practices can also be regarded as implicit models which should be explicable in a meta model which is expressive enough to encompass as many cases as possible, which is our approach.

4 A Meta Model for Team Composition

The extensive identification, classification and systematization of aspect entities on different levels of abstraction and from various studies and sources in our previous work together with the mathematical structure we will discuss now form our meta model for team composition. Its elements are

– Input Space, consisting of team member variables
– Output Space, consisting of output/outcome variables
– Dependencies that map the Input Space to the Output Space
– Team external variables that influence a team from the outside
– Constraints that have to be satisfied before a team is composed according to a model

In other words, given these constituents we can derive four different main ingredients that make up a team recommendation system:

1. *Team composition core models*: strategies and heuristics used to compose a team through the usage of team member variables, output/outcome variables and dependencies
2. The *context* in which the team performs its actual task defined by the team external variables and,
3. the organizational *constraints* that may affect the set of possible teams available for a recommendation.
4. Algorithms for actual recommendation/optimization using the team composition model consisting of the first three elements

In our first step we focus on the first topic. The implications for this decision are that by examining and developing a system using this concept only, no algorithmic support for different external aspects (such as task meta data, reward system applied, ...etc.) or constraint solving can be applied. This will be subject of future work. In absence of the two other parts described above in the recommender system, taking into consideration different team external variables (for the usage of single model instances on different sets of team external variables) and constraint solving (for filtering the set of possible teams), is left to the user or has to be solved using other support mechanisms.

We will now focus on the elements input space, output space and dependencies. In our future work we intend to discuss in detail how team external variables and constraints can be integrated in an algorithmic solution.

Input Space:

Every team member can be described by a set of team member variables (TMV)

$$\begin{aligned} TMV &= \{age, gender, numberOfContacts, competencies, \ldots\} \\ &= \{[0,120] \in \mathbb{N}, \{m, f\}, [0,1000] \in \mathbb{N}, \\ &\qquad \{\text{``}programming\text{''}, \text{``}marketing\text{''}, \ldots\}, \ldots\} \end{aligned}$$

Hence, a selection of such team member variables is defined as $TMV' \subseteq TMV$. E. g. $TMV' = \{age, gender\}$. Defining a Cartesian product space Q over TMV' as:

$$Q = tmv'_1 \times tmv'_2 \times \ldots \times tmv'_{|TMV'|}$$

where $tmv'_i \in TMV'$, each team member is described by a tuple of properties $t_i \in Q$. A team of N members is described by a set of N such tuples

$$t = \{t_1, t_2, \ldots, t_N\}$$

The space of such team description is denoted as T. In case TMV' consists of age and gender, a team member might therefore be described through the vector t_1=(36,m). The whole team might then be described as t=$\{(36, m), (24, f), (54, f), (36, f)\}$.

Furthermore, an exact specification can be formalized in the following way: a specification $q_i \subseteq Q$ defines a single team member's desired properties. E. g. for a single team member which is desired to be in the age range from 20 to 40 where the gender does not matter could be specified as $q_i = ([20, 40], \{m, f\})$. Furthermore, more complex specifications can be formalized using simple first order logic expressions or set operations respectively (the algebras are isomorphic) $(\vee, \wedge, \neg) \leftrightarrow (\cup, \cap, \bar{\ })$. A desired exact specification for a team can then be defined as

$$q = \{q_1, q_2, \ldots, q_N\}$$

The space of such (exact) specifications is denoted as $\tilde{Q}$. E. g. if in a team, members should be at least 35 years old and two of them females we can express that (exact) specification by the following set: $q = \{([35, 90], \{f\}), ([35, 90], \{f\}), ([35, 90],\{m, f\}), \ldots\}$. As we will see below, also more general specification without specifying an exact q could be formalized by choosing an appropriate analytic mapping α described in the paragraph "Dependencies".

Output Space:

The output space consists of the possible output/outcome variables of a team and can be formalized as such:

$$\begin{aligned} OUT = \{&\textit{average satisfaction, produced money,} \\ &\textit{judgment of team leader}, \ldots\} \\ = \{&[0, 10] \in \mathbb{R}, \mathbb{N}, [0, 10] \in \mathbb{N}, \ldots\} \end{aligned}$$

A selection $OUT' \subseteq OUT$ and an according Cartesian product space

$$O = out'_1 \times out'_2 \times \ldots \times out'_{|OUT'|}$$

where $out'_i \in OUT'$, can be used to specify the predicted outcome (see below).
Dependencies:

We map dependencies in two steps:

1. *An analytic mapping*
$$\alpha : \tilde{Q} \times T \rightarrow \mathbb{R}$$
maps an optional desired team specification and an actual team description (both according to a selected set of team member variables) to a real value. This kind of mapping includes for example *heterogeneity, sum* or *exact* and *fuzzy matching.* The *exact matching* analytic mapping could for example be used in team composition to set up a team with specific properties. An example for it could be a team consisting of exactly one manager. We choose as our Q for this example $Q = orgposition \times tmv'_2 \times \ldots \times tmv'_{|TMV'|}$. Our set $orgposition = \{$*"manager", "secretary", "housekeeper",* $\ldots\}$ consists of simple tags describing the organizational role. According to that, a desired specification for a team consisting of two members, one manager and one other team member which is not a manager would be $q = \{q_1, q_2\}$ where $q_1 = (\{\text{"}manager\text{"}\}, tmv'_2, \ldots, tmv'_{|TMV'|})$ and $q_2 = (orgposition \setminus \{\text{"}manager\text{"}\}, tmv'_2, \ldots, tmv'_{|TMV'|})$. For a $t \in T$ with $t_1 = (\text{"}housekeeper\text{"}, x_2 \in tmv'_2, \ldots, x_{|TMV'|} \in tmv'_{|TMV'|})$, $t_2 = (\text{"}manager\text{"}, x_2 \in tmv'_2, \ldots, x_{|TMV'|} \in tmv'_{|TMV'|})$ the specification is satisfied, i. e. $\exists t_i \in q_1$ and $\exists t_{j \neq i} \in q_2$. In this case, this analytic mapping α would return the value 1. If this is not the case, 0 will be returned.
An example for an analytic mapping without exact specification is *heterogeneity.* Heterogeneity of nationalities for example, can be mapped (operationalized) by Blau's index [9], which in this case represents the α function. According to it and used in context of our model, $q = \{\}$ and $Q = nationality \times tmv'_2 \times \ldots \times tmv'_{|TMV'|}$. E. g. the description of team member 1 could therefore be $t_1 =$ (*"English",28,"male",...*), that of team member 2 $t_2 =$ (*"German",48,"female",...*) and that of team member 3 $t_3 =$ (*"German",32,"female",...*). For this team α for example would map the element $t = \{t_1, t_2, t_3\}$ to the value: $1 - \sum p_k^2$ where k is the number of categories (in our case 2: German and English) and p_k their proportion in the group. In that case the result of these analytic mapping would be $\frac{4}{9}$.

2. *Heuristic mappings*
$$\chi : \mathbb{R}^i \rightarrow O$$
use this value from i different analytic mappings to map it on predicted output/outcome variables, i. e. they relate the result of the analytic mapping to an output/outcome with respect to a selected set of output/outcome

variables. E. g. they map the heterogeneity (analytic mapping function) of team member nationality (team member variable) to the team performance (output/outcome).

The overall mapping $\alpha \circ \chi$ describes the dependency of the input (desired and actual team properties) and the predicted output/outcome.

4.1 Formalization Summary

To conclude from the various formalizations in this section, we now define a team recommendation system S using the described components of our team composition meta model (section 4) in the following tuple:

$$S = (C, TEV, TMV, A, H, OUT)$$

where C is the set of available constraint variables (e. g. time, organization, location, etc.), TEV, the set of available team external variables, TMV the set of available team member variables, A the set of available analytic mappings $\alpha : \tilde{Q} \times T \rightarrow \mathbb{R}$ and H the set of the $\chi : \mathbb{R}^i \rightarrow O$ mappings that relate the results of the different analytic mappings to a vector space O consisting of a selection of elements of the outout/outcome selection OUT'. However, depending on the application domain and the algorithms applied on the meta model instantiations to recommend a team, not all the tuple elements have to be specified. For instance, team recommendation algorithms that do not consider the context (i. e. team external variables TEV) may omit this aspect.

5 Example Instantiations

In this section we will sketch the usage of our meta model by implementing models for team composition found in our literature review. Since the level of detail in the following models is not always detailed enough for an algorithmic use, we will tryto operationalize their suggested concepts in an exemplified way where necessary. Due to the restrictions mentioned in section 4 we are focusing on the *team composition model* aspects. Where necessary, we will describe the specifications of the other meta model components (i. e. team external variables and/or constraints) in an informal way.

As an example we first choose to formalize Pelled's model [13] mainly dealing with demographic diversities. The model is particularly suited for our mentioned focus, because of its emphasis on team member variables, dependencies (i. e. analytic and heuristic mappings) and output/outcomes.

As a team composition model developed from the management sciences, we instantiated a study from Bouncken and Winkler [2] on multicultural innovation teams. Besides being valuable because of the more practical orientation, this article is also very appropriate for our approach due to the "innovation" domain in which the study took place which relates well to our open innovation scenario described in section 1.1.

Our third example is a game theory approach to team composition developed by Mello and Ruckes [11] which shows that our meta model can also be applied on theories beyond the classical statistical-based approaches used for instance in socio-psychological sciences.

5.1 Pelled's Model

Pelled's Model describes how teams perform depending on demographic diversity. The approach is an extension of Ancona's work on demography and team performance [1]. Pelled differentiates between two types of demographic diversities: visible and job-related. She claims that visible demographic diversity has a bad influence on people's satisfaction and team performance because it fosters affective conflicts. Furthermore, she claims that job-related demographic diversity has a positive influence on performance because it fosters substantive conflicts. Substantive conflicts can be seen as an expression of the friction between different schools of thought. The effects of diversity on conflicts are seen as moderating team longevity, i. e. the amount of time a team has been together. In our language team longevity is a typical team intermediate variable (see section 2.2).

To operationalize this model with our meta model we first have to choose team member variables and the output/outcome of the team process. As team member variables for the job-related demographic diversity we select, according to Pelled's work, functional background, educational background and organizational tenure. For the visible demographic diversity variables in turn we select (again according to her work) age, gender and race as team member variables. Pelled also identifies the variable group tenure (team longevity) as a mix of both diversites so that also this team member variable is included in our model. We can imagine it as a social network whose ties are working relationships in a team: the stronger the tie, the more often team members worked together in the past. Our set TMV' therefore for Pelled's model is

$$\begin{aligned} TMV' = \{ & \textit{functionalBG, educationalBG, orgtenure,} \\ & \textit{age, gender, race, teamsocnet}\} \end{aligned}$$

where $TMV' \subseteq TMV$. As output/outcome we select the variables "judgment of team leader" (representing the "cognitive task performance" measure stated in the article) and "turnover".

$$OUT' = \{\textit{judgment of team leader, turnover}\}$$

where $OUT' \subseteq OUT$. Our dependencies will respectively map functional background, educational background and organizational tenure to judgment of team leader, and age, gender and race to turnover and task performance. In addition, both turnover and performance, are dependent on group tenure. A general overview of the resulting model is given in figure 1.

As our last step we have to define the dependencies more specifically. As mentioned in section 4, dependencies are operationalized by two kind of mappings: analytic mappings α and heuristic mappings χ.

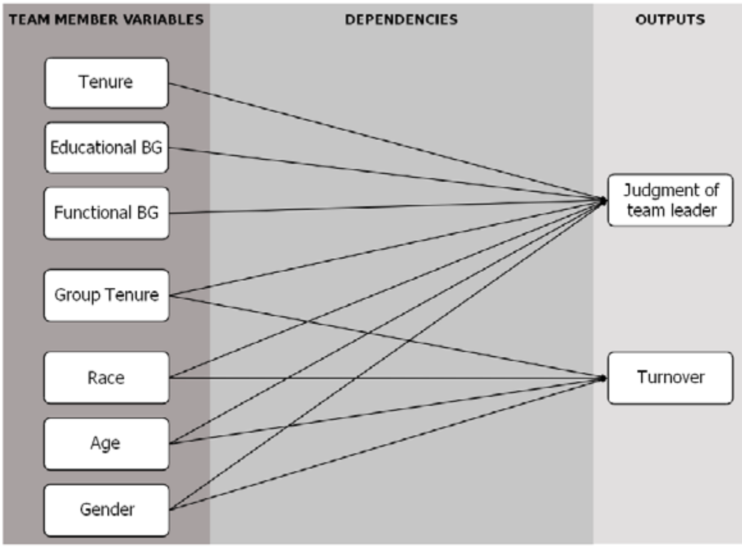


Fig. 1. Pelled's model implemented by our meta model (dependency specifications omitted for clarity)

In Pelled's case we identified two different types of analytic mappings. We will refer to them as α_1 and α_2. Both mappings are based on the concepts of diversity (or heterogeneity) mentioned in the specification of this composition model. α_1 stands for diversity as variety, such as in the case for educational background, functional background, gender and race in Pelled's model. It can be operationalized by Blau's index as described in section 4. α_2 represents diversity as separation and is usually operationalized with the traditional standard deviation [9]. α_2 is applied to the team member variables age and tenure.

In this model three heuristic mappings χ_1, χ_2 and χ_3 can be identified: χ_1 maps the numeric results of α_1 acting on some team variable and α_2 acting on others to turnover. χ_2 takes the same input and maps it to the judgment of the team leader. χ_3 on the other hand, maps the results of the analytic mappings for the job-related team member variables and group tenure to the judgment of team leader. From Pelled's model only the sign of the slope of the heuristic mappings can be acquired. Hence, the functions χ_i have to be configured with default mathematical functions (linear, quadratic, etc.) and values for slope, axis intercept, etc. However, by applying the model consecutively and monitoring their output/outcome (thus creating training data for the model) these functions can be adjusted accordingly. However, this is not part of the meta model itself but part of the recommendation algorithm acting on the meta model.

5.2 Bouncken and Winkler Model

Another theory for composing teams has been investigated by Bouncken and Winkler in their work about global innovation teams [2]. They conducted a longitudinal study on five large multi-cultural innovation teams (some of them consisting of three sub-teams) over two years. They tried to investigate how communication styles affect innovativeness of teams. The team's domain was food and cosmetics. Additionally they focussed on teams of one firm that were pursuing radical innovations. This could for instance be translated in our meta model through appropriate team external variables (e. g. variables *domain= "food or cosmetics"* and *innovativeness= "radical"*). Although the strings themselves in the absence of a formal ontology do not appear to carry any formal semantics, these team external variables can be used for improving context-awareness in the recommendation process e. g. they may be used to indicate to the project manager for which environment a specific model has been tailored. Moreover, algorithmic support for context-awareness using team external variables is an interesting research question for future work.

Bouncken and Winkler assume that communication styles differ according to the nationality and culture of team members in a team. An important assumption in this study is the association that a different country also implies a different culture. During their study they, for example, observed that the diversity among the culturally different team members with respect to the perception of time (as described by Hall and Hall [8]) will impede creativity and innovation.

Another finding was that in bi-national teams differences in power distance and "context" will lead to lower levels of creativity of the solutions and less innovation. "Context" refers the number of unwritten behaviour rules in a society. If in bi-national teams one half of the team is "low-context" (meaning the number of unwritten behaviour rules in their culture is rather low) and the other "high-context" (for more information on these criteria see [8]) the team will not perform well. Furthermore, if the power distance between team leader and team members is high, this also does not contribute positivity to the team's performance.

Their last proposition states that a highly diverse team by means of culture and nationality can positively influence individual motivation and group cohesion and thus be a source of creativity and innovation. This finding was based on multicultural teams with strong English skills.

We can formalize their approach through the definition of our meta model components team external variables, team member variables, analytic mappings, heuristic mappings and outputs/outcome: As team member variable we propose to choose the nationality, since it is an indicator of "cultural context" and also for the time perception dimension specified by Hall and Hall. Alternatively we could directly use "context" as team member variable. With respect to the decision power we may introduce a corresponding labeled team member variable representing the decision power on a scale from e. g. 1 to 10. Additionally we need the team member variable "language" with an attached weight that indicates the level of the skill. Our set of team member variables TMV':

$$TMV' = \{\textit{nationality, decision power, language}\}$$

where nationality is a string specifying the nationality (e. g. "French", "English", etc.), decision power an ordinal scaled variable ranging from e. g. 1 to 10 and language a set consisting of string-value tuples (e. g. ("English", 3), ("German", 10)).

Bouncken and Winkler's propositions can be adequately mapped to our meta model by using the analytic mapping *separation* (function of team member variable "nationality") and *disparity* (function of "decision power") as proposed in Harrison and Klein's work on diversity [9]. Additionally we need an analytic mapping *exact matching* that e.g. assesses that all the team members speak "English" well (i.e. level of at least 5) representing the proficiency of the respective language.

More precisely, to operationalize separation as our analytic mapping α_1 (a measure with properties similar to the standard deviation) can be used: This analytic mapping must yield a minimum result whenever there is only one category (in our case nationality) represented in the team. When each of the team members falls into a different category, the result of the mapping must be moderate, i.e. the result should be exactly between the minimum and maximum possible result of the mapping. When exactly two categories are represented in the teams such that the team becomes bi-modal, the analytic mapping produces the maximum possible output value. The difference to standard deviation hence barely consists in the scale of measurement which in this specific case is categorical-scaled instead of interval-scaled. How the mapping should behave between these values is not specified in the study and therefore has to be further investigated i.e. the model does not specify whether a single-culture team outperforms a multicultural one.

For operationalizing disparity in our second analytic mapping α_2 either the coefficient of variation (CV) or the Gini coefficient can be used as described in Harrison and Klein's work.

Additionally to the complete characterization of the team, α_3, our *exact matching* mapping uses a team specification called

$$q = \{q_i\} \text{ and } q_i = \{(\text{nationality, decision power, } > (\text{“English”}, 4))\}$$

where i is the desired number of team members. So α_3 matches the team specification q with his actual complete characterization of the team and determines whether they match or not.

A major problem regarding the mapping of this team composition approach to our meta model is the operationalization and identification of a corresponding output. Bouncken and Winkler often mention that the analyzed aspects influence creativity and innovation, although these two outputs are not further defined, described and specified. Possible output/outcome variables in our meta model could, for example, be a product idea's degree of creativity rated by experts on an ordinal scale.

Thus, to complete this model instantiation, we define a heuristic mapping χ_1 that takes the outputs of α_1, α_2 and α_3 as argument and maps them to the desired output/outcome variable. An important contribution of χ_1 is to map the interrelation between the matchings α_1 and α_3 such that α_3 receives more weight whenever α_1 has intermediate values (i.e. team is quite heterogeneous with respect to nationality) as a result.

5.3 Mello and Ruckes Model

A different approach for the analysis of the impact of heterogeneity on performance has been applied by Mello and Ruckes [11]. They used a game theory model to analyze how far heterogeneous teams affect team performance. Performance herein is specified as the cumulated payoff of the team. The scenario they use to analyze the problem in an exemplified manner is a simple two-person hierarchy in which a superior can choose a subordinate who is similar to him (i. e. resulting in a homogeneous team) or different from him (resulting in a heterogeneous team) in terms of team member variables. As an example for these team member variables Mello and Ruckes indicate training (as a source of knowledge) and organizational tenure (as an indicator for the skill of correctly judging opportunities). From their literature review they infer that a team has to have a team leader such as a CEO or a division head. The TMV' set hence consists currently of these team member variables, each containing structured or unstructured elements according to their specification. The set "attended trainings" could for instance be a simple set containing the name of all the training courses offered by the organization. The set "role" on the other hand could be a hierarchical structure of roles. The organizational tenure set "orgtenure", which measures the tenure in years, is equal to a suitable subset of $\mathbb{N}$.

$$TMV' = \{\textit{attended trainings, orgtenure, role}\}$$

Additionally, we can define our first analytic mapping α_1: that is an *exact matching* mapping over the hierarchy level. The related query $q \in \tilde{Q}$ is:

$$\begin{aligned} q &= \{q_1, q_2\} \\ q_1 &= \textit{(attended trainings, orgtenure,} \\ &\qquad \textit{``CEO''} \cup \textit{``division head'')} \\ q_2 &= \textit{(attended trainings, orgtenure,} \\ &\qquad \overline{\text{``}CEO\text{''} \cup \text{``}division\, head\text{''}}\,\textit{)} \end{aligned}$$

In this case just two team members are assumed (as Mello and Ruckes use in their example for reasons of clarity).

They assume that both team members may acquire information about the prospective project's payoff. Depending on the similarity they will have access to the same information sources or not. Mello and Ruckes demonstrate that in cases of high uncertainty and rapidly changing environments a heterogeneous team is more valuable, because the variety of the information collected by a heterogeneous team may be larger than the variety collected by the homogeneous team and thus is more useful to complement the team leader's information. This implies that firms operating in more uncertain environments should tend to fill managerial positions with people from outside the organization. Firms in stable environments should hire managers from inside the organization.

This again leads to the addition of another team member variable and to an extension of the parameter q in our analytical mapping α_1.

$$TMV' = \{\textit{attended trainings, orgtenure, role, inside}\}$$

where *inside* is the set of the logical values "true" and "false" that defines whether the person is inside or outside the organization. The above paragraph raises another important issue with respect to our meta model concerning *team external variables*. Mello and Ruckes distinguish environments that their respective findings can be applied to. Depending on the TEV "environment", a heuristic mapping χ_1 in this specific case will change the sign of the result of the analytic mapping defined above such that matching the query will result in a positive or negative output depending on the set of team member variables. This in turn could be modeled through our meta model by instantiating two different models that have to be applied according to the TEV "environment".

The first model instantiation for uncertain environments would then implement the following query q:

$$\begin{aligned} q &= \{q_1, q_2\} \\ q_1 &= \textit{(attended trainings, orgtenure,} \\ &\quad \textit{``CEO''} \cup \textit{``division head'', ``true'')} \\ q_2 &= \textit{(attended trainings, orgtenure,} \\ &\quad \overline{\text{``}\mathit{CEO}\text{''} \cup \text{``}\mathit{division\,head}\text{''}}\textit{, ``false'')} \end{aligned}$$

the second model instantiation for certain environments would be:

$$\begin{aligned} q &= \{q_1, q_2\} \\ q_1 &= \textit{(attended trainings, orgtenure,} \\ &\quad \textit{``CEO''} \cup \textit{``division head'', inside)} \\ q_2 &= \textit{(attended trainings, orgtenure,} \\ &\quad \overline{\text{``}\mathit{CEO}\text{''} \cup \text{``}\mathit{division\,head}\text{''}}\textit{, ``true'')} \end{aligned}$$

Another interesting proposition of Mello's work is that if the probability that the team leader can obtain information is low, a homogeneous team (also by means of project preferences) may lead to a higher expected payoff (output/outcome variable). The same statement is valid if the probability that the team leader can obtain a lot of information about the project is high. In this case the principal hires the subordinate just because of the effort the subordinate will invest in this project. In the other cases (i. e. when the probability is either not very high or not very low) a heterogeneous team has a greater expected payoff).

For a detailed reasoning of these statements we refer to Mello's article [11].

As described before, our only output variable in this team composition model is the overall payoff. Hence, our model instantiation consists of the previously declared team member variable set TMV', the dependency $\alpha_1 \circ \chi_1$ (χ_1 maps the result of α_1 to the payoff) and the TEV "environment certainty".

6 Conclusions and Future Work

Our translations of existing theories for team composition into our meta model indicate the compatibility of our approach with those theories. Hence, this in turn indicates that the meta model can be used as the foundation of a flexible team recommendation system. Through these translations we identified limits of the meta model and the team composition strategies found in the literature as well.

What our meta model is not inherently capable of is considering the implications of a change of team members during the task (e. g. as it is described in Mello's article). In future work we want to further investigate the applicability of our meta model. Moreover, through a formalization and a subsequent implementation of a corresponding generic recommender system for teams, we intend to empirically evaluate our approach.

Acknowledgment. This work was performed within the scope of the KoPIWA research project supported by the Federal German Ministry for Education and Research (BMBF) under contract no. 01FM07072.

References

1. Ancona, D.G., Caldwell, D.E.: Demography and design: predictors of new product team performance. Tech. rep. (1989)
2. Bouncken, R.B., Winkler, V.: Global innovation teams: cultural team composition, language capacities, and experience as success factors. European Journal of Management (June 2008)
3. Brocco, M., Groh, G.: Team recommendation in open innovation networks. In: RecSys 2009: Proceedings of the third ACM conference on Recommender systems, pp. 365–368. ACM, New York (2009)
4. Brocco, M., Groh, G., Kern, C.: On the influence of social factors on team recommendations. In: Proc. Second International Workshop on Modeling, Managing and Mining of Evolving Social Networks (M3SN), 26th IEEE International Conference on Data Engineering (2010)
5. Chen, S.-J., Lin, L.: Modeling team member characteristics for the formation of a multifunctional team in concurrent engineering. IEEE Transactions on Engineering Management 51(2), 111–124 (2004)
6. Chesbrough, H.W., Vanhaverbeke, W., West, J.: Open innovation: researching a new paradigm, illustrated, reprint ed. Oxford University Press, Oxford (2006)
7. Colucci, S., Noia, T.D., Sciascio, E.D., Donini, F.M., Piscitelli, G., Coppi, S.: Knowledge based approach to semantic composition of teams in an organization. In: SAC 2005: Proceedings of the 2005 ACM symposium on Applied computing, pp. 1314–1319. ACM, New York (2005)
8. Hall, E.T., Hall, M.R.: Understanding cultural differences: Germans, French and Americans. Intercultural Press, Yarmouth (1990)
9. Harrison, D.A., Klein, K.J.: What's the difference? Diversity constructs as separation, variety, or disparity in organizations. The Academy of Management Review 32(4), 1199–1228 (2007)

10. Kurtzberg, T.R., Amabile, T.M.: From guilford to creative synergy: Opening the black box of team level creativity. Creativity Research Journal 13, 285–294 (2000)
11. Mello, A.S., Ruckes, M.E.: Team composition. Journal of Business 79(3), 1019–1040 (2006)
12. Nunamaker, J., Dennis, A., Valacich, J., Vogel, D., George, J.: Electronic meeting systems to support group work. Communications of the ACM 34(7), 40–61 (1991)
13. Pelled, L.H.: Demographic diversity, conflict and work group outcomes: An intervening process theory. Organization Science 7, 615–631 (1996)
14. Powell, W.W., Grodal, S.: Networks of Innovators. In: The Oxford Handbook of Innovation, pp. 56–85. Oxford University Press, Oxford (2005)
15. Schippers, M.C., Hartog, D.N.D., Koopman, P.L., Wienk, J.A.: Diversity and team outcomes: the moderating effects of outcome interdependence and group longevity and the mediating effect of reflexivity. Journal of Organizational Behavior 24(6), 779–802 (2003)

Node Degree Distribution in Affiliation Graphs for Social Network Density Modeling

Szymon Chojnacki, Krzysztof Ciesielski, and Mieczysław Kłopotek

Institute of Computer Science PAS,
J.K. Ordona 21, 01-237 Warsaw, Poland
{sch,k.ciesielski,klopotek}@ipipan.waw.pl

Abstract. The purpose of this article is to link high density in social networks with their underlying bipartite affiliation structure. Density is represented by an average number of a node's neighbors (i.e. node degree or node rank). It is calculated by dividing a number of edges in a graph by a number of vertices. We compare an average node degree in real-life affiliation networks to an average node degree in social networks obtained by projecting an affiliation network onto a user modality. We have found recently that the asymptotic Newmann's explicit formula relating node degree distributions in an affiliation network to the density of a projected graph overestimates the latter value for real-life datasets. We have also observed that this property can be attributed to the local tree-like structure assumption. In this article we propose a procedure to estimate the density of a projected graph by means of a mixture of an exponential and a power-law distributions. We show that our method gives better density estimates than the classic formula.

1 Introduction

Bipartite or affiliation networks describe a situation in which we have two types of nodes and direct links only between nodes of different kinds. One type of nodes could be interpreted as actors and the second as events in which actors take part [1]. From this network we could induce (or unfold) two unipartite graphs containing only nodes of one type. Two nodes are connected in a projected graph when there exists a path of length two connecting them in an affiliation network. This transformation enables us to join movie actors if they costarred in at least one film. It also lets us to link researchers if they are coauthors of one article or to draw a relation between two named entities if they appear in one press topic. It has been shown that despite some similarities, in general the structural properties of a projected and a bipartite representations may differ significantly [2].

More formally, a graph is an ordered pair $G = (V, E)$ comprising a set of vertices (or nodes) V and a set of edges (or links) $E \subseteq \{V \times V\}$. A bipartite network (or bigraph) is a graph $G = (U \cup A, E)$ which vertices can be labeled by two types U and A. The difference with a classic unipartite graph is the fact that V consists of two disjoint sets $V = \{U \cup A, U \cap A = \emptyset\}$ and edges exist only

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 51–61, 2010.

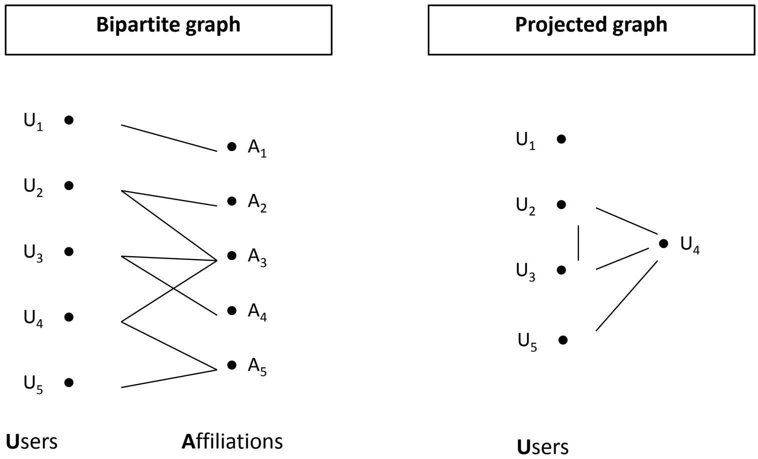


Fig. 1. An original bipartite affiliation network on the left is projected onto a user modality by linking two users in a projected graph if there exists a path of length two connecting them in an underlying affiliation network

between nodes of different types $E \subseteq \{U \times A\}$. In this article we will denote an affiliation graph as $G = (U \cup A, E)$ and its projection onto a user modality as $G' = (U, E')$. We focus on the user modality, however the conducted analysis can be applied also to the group modality.

In Fig. 1 we show a simple example of projecting a bipartite network. The total number of edges in the bigraph is eight, while in the unipartite graph it is four. Node U_1 is isolated in a projected graph as it is connected only to the group A_1, which size is one. Nodes U_2 and U_3 are linked in a projected representation as they share an affiliation A_3 in the underlying bipartite graph. The opposite transformation can be conducted by an analysis of cliques (i.e. complete subgraphs) in the direct neighborhood of a node. However, the problem of identifying the minimal clique covering belongs to the class of *NP - complete* problems.

High density of real social networks is a macro feature that besides of topological micro patterns results in important properties of the networks. In purely random model studied by Erdös [3] all nodes form a single connected component with high probability if a certain density threshold is exceeded. High density coexists with short average path lengths between randomly selected nodes or a small diameter of a graph (i.e. the maximum of shortest paths). The model was criticized for generating Poisson node degree distribution and low clustering coefficients, which is contrary to the empirical evidence. Other models where introduced to explain *small world* paradigm [4] in terms of topological properties of the networks [5] or power-law node degree distribution by means of the preferential attachment and growth mechanisms [6]. It is interesting to measure a density of a graph over time. It has been observed recently that online social networks densify over time [7].

The rest of the paper is organized as follows. Section 2 surveys the related work. In Section 3 we define a random density estimation procedure. Section 4 gives the results of our simulations. We conclude and discuss the implications of our findings in the last fifth section.

2 Related Work

Early analysis of affiliation networks was limited by the size of available datasets. The situation changed with the development of the Internet and new random models for bigraphs were proposed. In a model for bipartite graph generation described in [2] the number of nodes is fixed and we draw a degree for each node from a predefined distribution, in the second step the ends of generated edges are connected randomly. An iterative model for bipartite graph generation was proposed in [8]. During one iteration a node is added, its degree is drawn from a defined distribution and for every edge a decision is made whether to join it to the existing nodes following the preferential attachment rule or join it to a newly created node. A model described in [9] tries to simulate at the same time affiliation network and implicitly given social network.

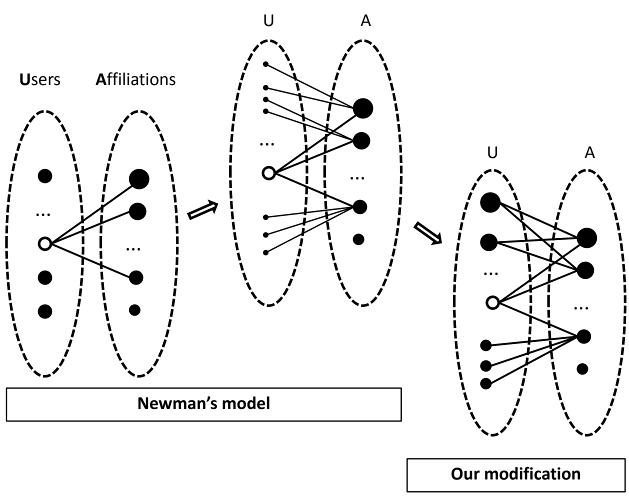


Fig. 2. In the Newman's model a random user's degree is drawn from a predefined distribution. Each edge is connected to a different group. The probability of group selection is proportional to its size. Every edge from one group is linked to a different user, it is assumed that users chosen by various groups do not overlap. In our modification we enable different groups to be connected to same users and probability that a user is selected is proportional to its degree.

The above mentioned models assume that the graph generation follows from a defined mechanism. The model introduced by Newman [10] is much more generic and enables us to analyze any graph with given node degree distributions. However, it is based on an assumption that analyzed graph is locally tree-like. Consequently, the expected user's degree in a projected graph tends to an average user's degree in the bipartite graph times the expected degree of an affiliation group drawn with the preferential attachment minus one. Let's consider a bigraph $G = (U \cup A, E)$, which has $|U|$ users with degrees $u_1 \dots u_{|U|}$ and $|A|$ groups with degrees (or sizes) $a_1, \cdots, a_{|A|}$ then the expected number of edges going out from a group selected randomly with preferential attachment is:

$$\frac{a_1}{\sum a_i}(a_1 - 1) + \cdots + \frac{a_{|A|}}{\sum a_i}(a_{|A|} - 1) = \frac{\sum a_i^2}{\sum a_i} - 1. \tag{1}$$

An average user's degree is $\sum a_i/|U| = |E|/|U|$. The local tree-like structure assumption results in a fact that all paths of length two starting from a random user go to different users (see Fig. 2). Hence, the expected total number of edges in the unipartite graph $\mathcal{E}(|E'|)$ (projected onto a user modality) is:

$$\mathcal{E}(|E'|) = \left(\frac{\sum a_i^2}{\sum a_i} - 1\right)|U|. \tag{2}$$

If we omit the local tree-like structure assumption. The value can be perceived as an upper bound for the expected number of edges in a projected graph. It can be shown by means of the Cauchy-Schwartz inequality [11] that the upper bound $\overline{\mathcal{E}(|E'|)}$ is limited from below:

$$\overline{\mathcal{E}(|E'|)} \geq |U|(|E|/|A| - 1). \tag{3}$$

Moreover, the lower bound can be increased if there is an outlying group among $a_1 \dots a_{|A|}$ (an existence of such group is much higher in the power-law than in the exponential distribution).

Our analysis is motivated by a fact that in many projections of bigraphs an average node degree differs significantly from the theoretical value proposed by Newman. We have shown in [11] that:

- the number of edges in projected graphs obtained from various real-life affiliation networks is higher than the expected theoretical value
- the expected theoretical value is also higher than observed in projections generated for random graphs
- for a bipartite graphs with fixed number of nodes and edges, the density of a projected graph depends on the node degree distributions of both modalities in an affiliation network. The more exponential-like than Zipf-like the group degree distribution, the higher density is observed, but for the user modality the relation is opposite.

This article aims to enrich the analysis presented in [11] with a complete procedure of estimating a density of a projected graph by means of features of the underlying affiliation network.

3 Density Estimation Procedure

The purpose of this section is to describe our random procedure dedicated to estimate the density of a projected graph. The procedure consists of two steps:

1. the real data are used to estimate the parameters of node degree distributions in an affiliation network
2. random projections are simulated and the density in the projected graphs is measured

Firstly, the distributions used in the procedure are defined. Secondly, the estimation techniques are described. At the end of the section we outline the random projection algorithm.

3.1 Degree Distributions

The shape of node degree distributions in social networks is usually modeled by continuous distributions. The predominant role play two distributions: the power-law and the exponential. During our simulations we utilize the discrete counterparts of the distributions i.e. the Zipf and the geometric distributions. Because any pseudo-number is generated by a computer for a limited interval, we need to define truncated versions of the two distributions.

The frequency of an element of rank k in truncated geometric distribution with parameter $p \in (0, 1)$ and maximum rank N is:

$$f_{geo}(k; p, N) = \frac{p(1-p)^{k-1}}{1-(1-p)^{N-1}}. \tag{4}$$

The mean value of a random variable X with the above distribution is given by:

$$\mathcal{E}(X) = \mathcal{E}_{geo}(p, N) = \frac{1 - N(1-p)^{N-1} + (N-1)(1-p)^N}{p(1-(1-p)^{N-1})}. \tag{5}$$

One can verify that the truncation is realized by the division of the classic geometric density function by the value of the cumulative function at N. The expected value can be obtained by summing a chain of a geometric series.

The frequency of an element of rank k in Zipf distribution with scaling exponent $s > 1$ and maximum rank N is:

$$f_{Zipf}(k; s, N) = \frac{k^{-s}}{\sum_{n=1}^{N} n^{-s}}. \tag{6}$$

The mean value of a random variable Y with the above distribution is given by:

$$\mathcal{E}(Y) = \mathcal{E}_{Zipf}(s, N) = \frac{\sum_{n=1}^{N} n^{-(s-1)}}{\sum_{n=1}^{N} n^{-s}}. \tag{7}$$

The mixture with a parameter α of f_{geo} and f_{Zipf} is defined as:

$$f_{mix}(k;\alpha,p,s,N) = \alpha \cdot f_{geo}(k;p,N) + (1-\alpha) \cdot f_{Zipf}(k;s,N). \tag{8}$$

In the following we will analyze a mixture of two distributions with equal expected values.

3.2 Parameter Estimation

Given a real-life bigraph $G = (U \cup A, E)$ with empirical node degree distribution density function represented by $f_U(k)$ (for the user modality) and $f_A(k)$ (for the group modality) and the maximum node degree for both modalities N_U and N_A we estimate the parameters for the both modalities as follows.

The maximum rank in the geometric and the Zipf distributions are set to N_U and N_A. The parameter p and the scaling exponent s are obtained for both modalities by finding numerically the values which guarantee the corresponding expected values (Eq. 5 and Eq. 7) equal to those observed in the data set:

$$\begin{cases} \mathcal{E}_{geo}(\hat{p}_u, N_U) &= \frac{|E|}{|U|} \\ \mathcal{E}_{Zipf}(\hat{s}_u, N_U) &= \frac{|E|}{|U|} \\ \mathcal{E}_{geo}(\hat{p}_a, N_A) &= \frac{|E|}{|A|} \\ \mathcal{E}_{Zipf}(\hat{s}_a, N_A) &= \frac{|E|}{|A|} \end{cases} \tag{9}$$

The values of parameter α_u and α_a are found numerically with simplex method by minimizing the least squares distance between empirical node degree distributions and the parametrized mixture with respect to the condition $0 \leq \alpha \leq 1$:

$$\begin{cases} \hat{\alpha}_u = \underset{\alpha}{\operatorname{argmin}} \sum_k (f_{mix}(k;\alpha,\hat{p}_u,\hat{s}_u,N_U) - f_U(k))^2 \\ \hat{\alpha}_a = \underset{\alpha}{\operatorname{argmin}} \sum_k (f_{mix}(k;\alpha,\hat{p}_a,\hat{s}_a,N_A) - f_A(k))^2 \end{cases} \tag{10}$$

The values of the six estimated parameters $\hat{\alpha}_u$, $\hat{\alpha}_a$, $\hat{p}_u$, $\hat{p}_a$, $\hat{s}_u$, $\hat{s}_a$ are transfered to the random projection algorithm in the next subsection.

3.3 Random Projection Algorithm

In order to assess the number of edges in a projected graph we need to undertake a few steps. A graph is initialized with a number of user and a number of affiliation nodes equal to those in the original graph, $|U|$ and $|A|$ respectively.Then, for every user and every affiliation group node a degree is drawn from an appropriate mixture distribution. In the next step we iterate over all users and each user's edge is

Algorithm 1. Random projection of a bipartite graph

```
input : |U| - number of users, |A| - number of affiliation groups,
        f_mix(k; α̂_u, p̂_u, ŝ_u, N_U) - user node degree probability function,
        f_mix(k; α̂_a, p̂_a, ŝ_a, N_A) - affiliation node degree probability function
output: |E'| - number of edges in a projected graph

Initialize the sets of users and groups;

Users ← {u_1 ... u_|U|} - a set of users;
Groups ← {a_1 ... a_|A|} - a set of groups;
|E'| ← 0 ;

for u ∈ Users do |u| ← DrawUserDegree(f_mix(k; α̂_u, p̂_u, ŝ_u, N_U));
for g ∈ Groups do |g| ← DrawGroupDegree(f_mix(k; α̂_a, p̂_a, ŝ_a, N_A));

Calculate user degrees in the projected graph;

for u ∈ Users do
    NeighborsAll ← ∅;
    SelectedGroups ← ∅;
    for i ← 1 to |u| do
        GroupNeighbors ← {u};
        g ←SelectGroup(Groups \ SelectedGroups) ;
        SelectedGroups ← SelectedGroups ∪ g ;
        // draw a group without replacement
        for j ← 1 to |g| do
            Neighbor ← SelectUser(Users \GroupNeighbors) ;
            GroupNeighbors ← GroupNeighbors ∪ Neighbor;
            // draw a user without replacement for a given group
            // a neighbor user can belong to a group chosen earlier
            if NeighborsAll ∩ Neighbor = ∅ then
                NeighborsAll ← NeighborsAll ∪ Neighbor ;
    |E'| ← |E'| + |NeighborsAll| ;
```

connected to a random group. The probability that a group will be selected is proportional to its degree (size). The groups are drawn without replacement. Hence each user's edge is linked to a different group. For every selected group we draw a set of users that are connected to it. Every group is connected to exactly as many different new users as its degree minus one. However the sets of new users may overlap. Again, a user is drawn with a probability proportional to its degree. In this way we get a number of distinct neighbors of each user and can calculate the number of edges in a graph projected onto a user modality.

4 Experimental Results

We have used three real-life online datasets during the experiments. Basic statistics are contained in Table 1. The CiteULike dataset consists of a set of 5 208

Table 1. The datasets used in the experiments

	Group modality			User modality		
	nodes	maxRank	avgRank	nodes	maxRank	avgRank
CiteULike	2 336	71	3.1	5 208	59	1.4
BibSonomy	93 756	488	2.7	3 617	10 689	70
Movie - IMDB	127 823	294	11.5	383 640	646	3.83

users of http://www.citeulike.org social bookmarking portal. CiteULike is a metaknowledge management system, it enables researchers to save information about favorite publication, attach labels to the resources and see the bookmarks of other users. The users are allowed to create and join groups of interest. The BibSonomy dataset comprises a set of 3 617 users and 93 756 affiliation groups. BibSonomy is also a social bookmarking portal http://www.bibsonomy.org. Its functionality is broader than in case of CiteULike, as it also enables to bookmark websites. However, we did not have an information about users' membership and used tags as groups. In this way two users would be linked in a projected graph if they have used at least one same tag in the past. The Movie database was contains 383 640 users (actors) and 127 823 films (groups). It was downloaded from the Internet Movie Database http://www.imdb.com. Two actors were connected in a projected graph if they costarred at least once in the past.

The estimated values of the parameters of the exponential and the Zipf distributions are presented in Table 2. Also the values of mixture parameter α are given in the table. The parameters were calculated separately for every dataset and every modality. When $\alpha = 0$ then it means that the exponential factor in the mixture distribution can be neglected. When $\alpha = 1$ then the shape of a distribution does not posses power-law features. The six distributions are visualized in Fig. 3.

Table 2. The parameters estimated for the node degree probability density functions

	Group modality			User modality		
	$\hat{p}_a$	$\hat{s}_a$	$\hat{\alpha}_a$	$\hat{p}_u$	$\hat{s}_u$	$\hat{\alpha}_u$
CiteULike	0.322	1.966	0.155	0.71	2.912	0.0
BibSonomy	0.37	2.217	0.0	0.014	1.52	0.576
Movie - IMDB	0.086	1.548	1.0	0.261	2.04	0.0

We have run random projection algorithm 20 times for every dataset. The numbers of edges in the projected graphs are given in Table 3. The theoretical number of edges was calculated with Eq. 2 and is always higher than the real number of edges in the projected graphs. An average number of edges approximated by our method is closer in all cases to the real value. The standard deviation is below 6% of an average.

The theoretical value exceeds the real one in the CiteULike, BibSonomy and IMDB datasets by a factor of 1.67, 12.7 and 2.7. In our method the corresponding

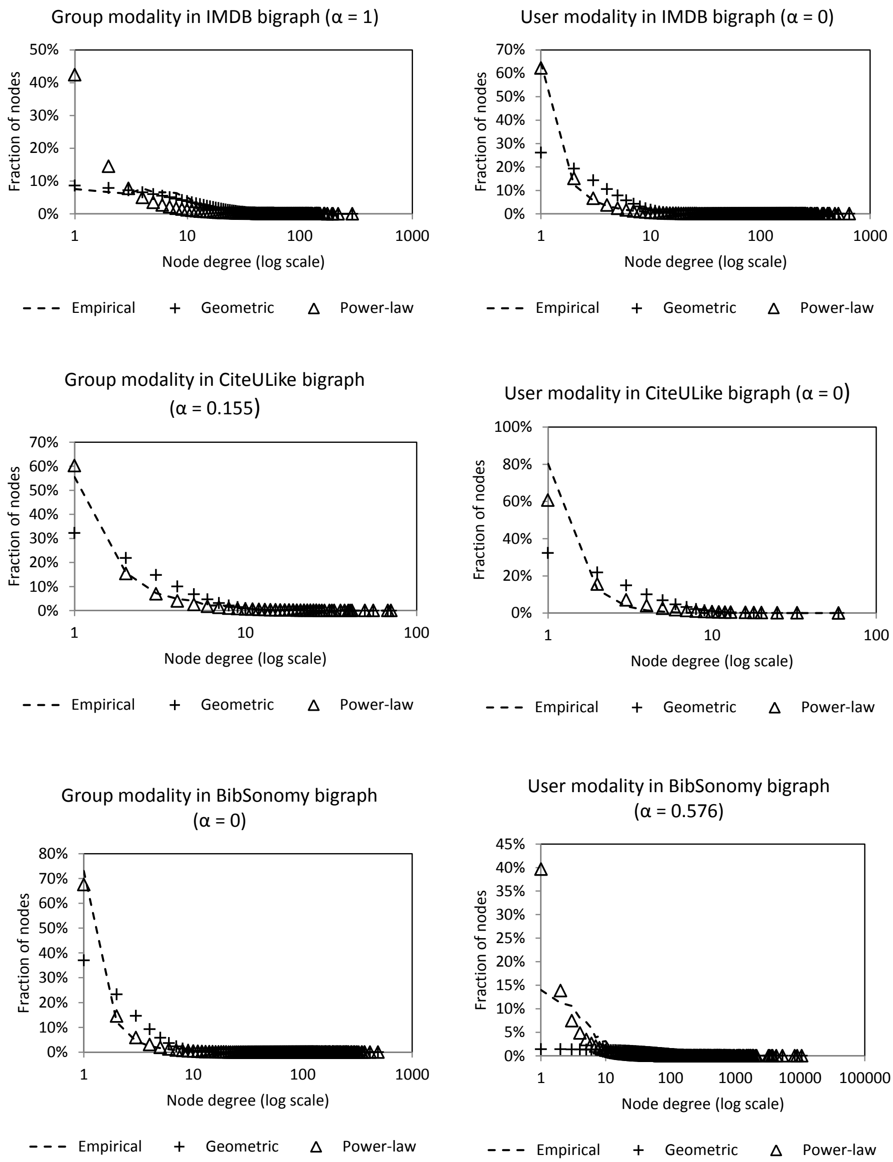


Fig. 3. Empirical node degree distributions of user and group modalities for IMDB, BibSonomy and CiteULike affiliation networks. The geometric and Zipf distributions drawn for the values of p parameter and s scaling exponent satisfying numerically Eg. 5 and Eq. 7. The values of α obtained from least square fitting the mixture of both distributions to the empirical data.

Table 3. The number of edges in both the unipartite and the bipartite representations. The number of edges in Our Method is an average over 20 simulations. The number in brackets stands for a proportion of the standard deviation to the average.

	Bipartite	Projected graphs Real	Theoretical	Our Method
CiteULike	7 196	74 190	124 773	88 567 (6%)
BibSonomy	253 366	1 810 132	23 797 094	2 682 723 (6%)
Movie - IMDB	1 470 404	30 076 166	81 125 160	30 836 072 (1%)

factors are 1.2, 1.5 and 1.03. The improvement is visible for all datasets. However, only in case of the third dataset we managed to find good approximation. It suggests that omitting the local-tree like structure assumption is not always sufficient to calculate the number of edges in a projected graph with high precision. We predict that there are strong correlations between users' groups in real-life affiliation networks which can not be grasped by our random projection algorithm.

5 Conclusion

In this article we analyzed the relationship between the density of a social network and node degree distributions in the underlying affiliation bigraph. The formula proposed by Newman to link the two values is not accurate for real-life datasets. It is based on a local tree-like structure assumption, which is not met in online affiliation networks. We have shown that by omitting the assumption we can approximate the density of a projected graph with higher accuracy. However, in two out of three analyzed datasets there still exists a gap between approximated and real values. This result suggests that there might exist local cliques in real-life bigraphs that are responsible for shrinking the density.

Acknowledgments. This work was partially supported by Polish state budget funds for scientific research within research project *Analysis and visualization of structure and dynamics of social networks using nature inspired methods*, grant No. N516 443038.

References

1. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge (1994)
2. Newman, M., Watts, D., Strogatz, S.: Random graph models of social networks. P. Natl. Acad. Sci. USA 99, 2566–2572 (2002)
3. Erdos, P., Renyi, A.: On the evolution of random graphs. Mathematical Institute of the Hungarian Academy of Sciences (1960)
4. Milgram, S.: The small-world problem. Psychology Today 2, 60–67 (1967)
5. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature 393, 440–442 (1998)

6. Barabási, A., Albert, R.: Emergence of scaling in random networks. Science 286(5439), 509–512 (1999)
7. Leskovec, J., Kleinberg, J., Faloutsos, C.: Graphs over time: densification laws, shrinking diameters and possible explanations. In: KDD 2005: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 177–187. ACM Press, New York (2005)
8. Guillaume, J.-L., Latapy, M.: Bipartite structure of all complex networks. Inf. Process. Lett. 90(5), 215–221 (2004)
9. Zheleva, E., Sharara, H., Getoor, L.: Co-evolution of social and affiliation networks. In: Fogelman-Soulié, F., Flach, P.A., Zaki, M. (eds.) KDD, pp. 1007–1016. ACM, New York (2009)
10. Newman, M., Strogatz, S., Watts, D.J.: Random graphs with arbitrary degree distributions and their applications, vol. 64 (July 2001)
11. Chojnacki, S., Kłopotek, M.: Bipartite graphs for densification in social networks. In: European Conference on Complex Systems (2010)

Achieving Optimal Privacy in Trust-Aware Social Recommender Systems

Nima Dokoohaki[1], Cihan Kaleli[2], Huseyin Polat[2], and Mihhail Matskin[1,3]

[1] Department of Electronics, Computer and Software Systems
Royal Institute of Technology (KTH), 16440, Kista, Stockholm, Sweden
nimad@kth.se, ckaleli@anadolu.edu.tr, misha@kth.se

[2] Department of Computer Engineering, Anadolu University
Eskisehir, 26470, Turkey
polath@anadolu.edu.tr

[3] Department of Information and Computer Science
Norwegian University of Science and Technology (NTNU), Trondheim, Norway

Abstract. Collaborative filtering (CF) recommenders are subject to numerous shortcomings such as centralized processing, vulnerability to shilling attacks, and most important of all privacy. To overcome these obstacles, researchers proposed for utilization of interpersonal trust between users, to alleviate many of these crucial shortcomings. Till now, attention has been mainly paid to strong points about trust-aware recommenders such as alleviating profile sparsity or calculation cost efficiency, while least attention has been paid on investigating the notion of privacy surrounding the disclosure of individual ratings and most importantly protection of trust computation across social networks forming the backbone of these systems. To contribute to addressing problem of privacy in trust-aware recommenders, within this paper, first we introduce a framework for enabling privacy-preserving trust-aware recommendation generation. While trust mechanism aims at elevating recommenders accuracy, to preserve privacy, accuracy of the system needs to be decreased. Since within this context, privacy and accuracy are conflicting goals we show that a Pareto set can be found as an optimal setting for both privacy-preserving and trust-enabling mechanisms. We show that this Pareto set, when used as the configuration for measuring the accuracy of base collaborative filtering engine, yields an optimized tradeoff between conflicting goals of privacy and accuracy. We prove this concept along with applicability of our framework by experimenting with accuracy and privacy factors, and we show through experiment how such optimal set can be inferred.

Keywords: Privacy, Trust, Optimization, Data Disguising, Social networks, Collaborative filtering, Recommender systems.

1 Introduction

Adaptive Web and its myriads of techniques are paving the path towards fulfilling the promise of alleviating classic problem of information overload. Recommenders,

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 62–79, 2010.

one of the most widely adopted and well-anticipated of this stack of technologies, remain the sole leader of this essential advancement. Recommenders intend to provide people with suggestions of products they will appreciate, based upon their past preferences, history of purchase, or demographic information [1]. Most successful recommenders employ well-known collaborative filtering (CF) techniques [2]. CF automates the word-of-mouth process, when asking like-minded friends or family members for their individual opinions on different matters like new movie releases. This process involves finding users similar to the user receiving the recommendation and suggesting her items rated high in the past by similar taste users. Since there are always numerous items and the ratings scored by users are sparse, often the step of finding similar users fails. To alleviate this shortcoming, the former step was replaced by utilizing a trust metric, which enables a trust-based heuristic to propagate and spot users whom are trustworthy with respect to active user *(a)* that we are gathering recommendations for [14]. Recommenders that take advantage of fusion of interpersonal trust with CF heuristics within and across their architectures are collectively referred to as *Trust-Aware Recommender Systems* [16].

Privacy remains a foundational problem in personalization research. In general CF systems usually fail to protect users privacy. Users who remain concerned about their privacy might use false data in the process. Using false data decreases accuracy of CF systems [7]. Users who are concerned about their privacy may employ false data, because data collected for CF can be used for unsolicited marketing, government surveillance, profiling users, misused, and it can be transferred [6]. As a matter of fact, it is more likely that users will give more truthful data if privacy measures are provided. Massa and Avesani [16] study the architecture and design of trust-aware recommender systems and describe how trust-aware recommenders alleviate shortcomings of traditional systems. Trust-aware recommenders are modeled and designed in a decentralized fashion. However, current implementations are either centralized or are not tested in a decentralized fashion [16]. As a result, there is a growing concern about the vulnerability of these systems to shilling attacks [28]. At the same time, as most research invested in analyzing trust-aware recommenders, focuses on improving the recommendations, they fail to clearly address the privacy issues surrounding the architecture and components of these systems. As a result this research work invests on dealing with privacy issues surrounding the architecture and components of trust-aware recommender systems.

To address these shortcomings, during the pace of this manuscript, we extend the architectural landscape of traditional CF techniques and trust-aware recommenders to include building blocks required for realizing a privacy-preserving trust-aware recommender system. As an example of such architecture, we implement a framework for applying data perturbation techniques to user rating profiles. To do this, we introduce a private trust computation process. Then, accordingly, we propose methods for producing private recommendations based on trust-based CF recommender systems. We ground this framework at the top of a social trust recommender system [19], which utilizes *T-index* [31] [32] as its trust metric. We will show how the overall trust estimation can be augmented to accommodate the private trust estimation and prediction generation. We design this framework, having protection and preserving users privacy in mind, while still providing accurate recommendations on masked data using trust-enabled CF schemes. We conceptualize this tradeoff between accuracy and privacy as a Pareto frontier notion. We will show that privacy and trust mechanisms, each with their respective configurations jointly form configurations of the overall framework. According to Pareto optimality perspective, at least a joint setting of both configurations exists which when utilized results in privacy of user data being maintained, while keeping accuracy decent at the same time. To evaluate this framework, we study the accuracy of the recommendations under different masked distributions and compare the results of the computations with original data.

Our experiment results clearly show that the proposed scheme provides recommendations with decent accuracy while preserving users privacy. The rest of the manuscript at hand is organized, as follows: First, a background into the main concepts shaping the foundation of this work is presented. The architecture of the system is presented in the third section, followed by a detailed description of the approach. Experimental evaluation is presented in the forth section, followed by a discussion of results. Finally, a conclusion and future work brings this work to its respective end.

2 Background

2.1 Trust-Aware Collaborative Filtering

CF algorithms generally make recommendations based on similarity between the users tastes. Similarity measure is not sufficient when user rating scores are sparse and insufficient. In the face of these shortcomings, traditional user similarities deem useless and recommenders need new ways to calculate user similarity. As a response to this problem, interpersonal trustworthiness was proposed to replace old similarity measures. Ziegler et al. [18] describe a relationship between how similarity between two users can be interpreted as how much they might trust each other. Golbeck [13, 14] shows the correlation between similarity and trust and how it can elevate movie recommendation accuracy. Taking into account this fact, trust can be considered as a measure for expressing the relationship between two users in recommendation systems. ODonovan and Smyth [17] approach trust-aware recommenders by utilizing a two-mode profiling model that documents the past behavior of users. Massa and Avesani [15, 16] present architecture for a trust-aware recommender system in which trust can be propagated and aggregated for all of the users in a social network setting. Lathia et al. [18] model a variation of *kNN* (*K-Nearest Neighbor*) CF recommender, which allows users to learn who and how much to trust by evaluating the utility of the rating information they receive. One of the problems with frameworks presented above is that the functionality of previous recommenders is dependent on availability of explicit trust ratings in between users to infer other trust relations.

Zarghami et al. [19] introduce a decentralized trust-aware recommender system, which utilizes T-index [33], as a trust metric for filtering trust between users. Unlike previous approaches, a trust network between users can automatically be built from existing ratings between users. Framework increases the probability of finding trustworthy users across the network by creating a Distributed Hash Table (DHT) like list of trustees, TopTrusteeList (TTL) [19] that wraps around the items, which are tasted similarly to those of current user. Our work utilizes this recommender as the foundation of our framework.

2.2 Privacy-Preserving Collaborative Filtering

Privacy remains the most significant problem in the context of CF recommendation systems. Canny [4, 5] proposes privacy-preserving schemes for CF. In his schemes, users control their private data and they are capable of getting personalized referrals produced without disclosing their data. Canny proposes to use homomorphic encryption in his schemes to protect individuals privacy. Polat and Du [6] employ perturbation techniques to offer predictions. In their scheme, users disguise their private data before sending it to central server that collects masked data instead of actual data. Kaleli and Polat [7] study how to produce predictions while preserving individuals privacy while producing na?ve Bayesian classifier (NBC)-based private recommendations. They employ randomized response techniques (RRT) to protect users privacy. Parameswaran [10] presents a data obfuscation technique in which she designs and implements a privacy-preserving shared collaborative filtering framework using data obfuscation algorithm. Berkovsky et al. [11] investigate a decentralized approach, which does not require sending data to a centralized server. Collaborative filtering techniques can be employed in the context of peer-to-peer (P2P) and social networks. Kaleli and Polat [6] propose a solution to produce NBC-based private recommendations in a P2P network. Authors present a solution to produce private referrals in a social network context [9]. Proposed solution requires using data disguising techniques.

Within the context of our framework, we have adopted this approach to provide private recommendations in the context of a trust network of users, where actual user profiles are masked and trust computation process and recommendation procedure are changed accordingly to produce private recommendations.

2.3 Preserving Privacy in Trust-Aware Recommender Systems

Taking measures for preserving privacy during trust calculation and computation has been of great importance. Lack of privacy protection within the context of systems dealing with trust and reputation, can *ease attacks by malicious insiders*, as they might infest the existing trust establishments or alter the trust computation results.

As a result, great deal of research has been invested in analyzing schemes for combining privacy with trust establishments in different fields. In Multi-Agent Systems, preserving privacy during trust negotiations between software agents in any open system is a crucial task because sensitive data is exchanged between strangers without any prior knowledge of each other [20, 21]. In P2P systems, similar concern for privacy is raised about the possibility that malicious users can exploit the peers trust network to spread tampered-with resources [22]. In the context of recommender systems, Lam et al. [23] give an overview of privacy and security problems with recommenders. These problems are twofold: the personal information collected by recommenders raises the risk of unwanted exposure and malicious users can bias or sabotage the recommendations that are provided to other users [23]. While former points out to *privacy of recommenders*, the latter is collectively referred to as *Shilling attacks* [24]. Attacks on recommenders remain a significant security hole in these systems [16, 22]. As popularity of trust-aware recommenders in academic and industrial community increases, problem with attacks on trust-enabled recommenders remains at large. Zhang [25] executes an average shilling attack on a trust-aware recommender system and demonstrates that trust-recommender exhibits more stability over a traditional kNN-based recommender.

Our framework is designed with idea of being capable of withholding shilling attacks in mind. As the main focus of this work is on implementation and design of a privacy-preserving trust recommender, we leave the analysis of framework stability under different attacks for the future work.

3 Recommendation Framework

In this section, we present the framework that we have composed for building a private trust-aware recommender system. To do so, first, a brief introduction into the architecture and design of our trust-aware recommender is presented in the first section, while in the second section; we describe how this architecture can be extended with components needed to build a private trust recommender. This is followed by description of the process of trust estimation and prediction generation of our resulting system. In the last section we present the definition of optimal privacy set and the process of how to infer this set, with respect to the context of this work.

3.1 Architecture of a Private Trust-Aware Recommender System

Massa and Avesani [16] present a generic architecture for a trust-aware recommender system. This architecture is presented in Fig. 1.

In this architecture, gray boxes present modules, while white boxes represent the matrices used as input and outputs of algorithms. Typical inputs of the architecture are: *rating matrix* (rating scores assigned to items by users) and *trust network* [29] (trust statements of users with respect to each other). While rating is the main input of traditional CF recommenders, trust can be inferred and in our case, automatically generated out of the rating matrix. In this architecture, they visualize the anatomy of a traditional CF recommender being composed of two main building blocks: a *similarity metric* and a *rating predictor*.

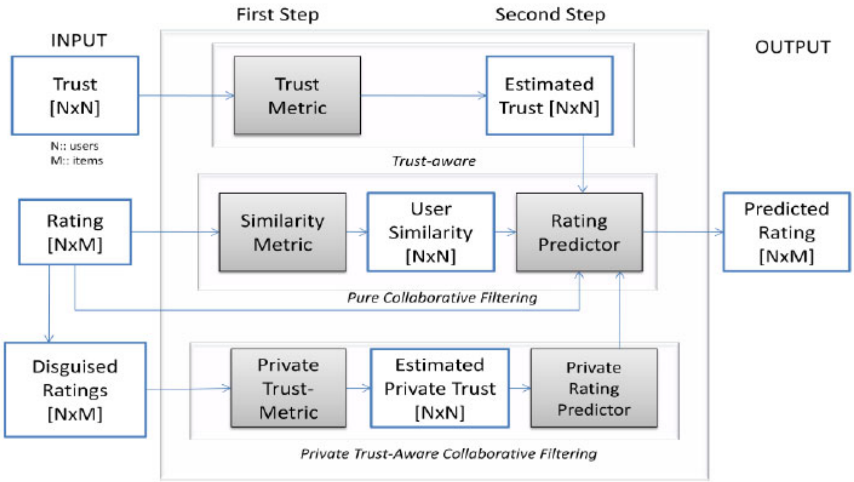


Fig. 1. *Architecture of a Private Trust-Aware Recommender System.* Trust-aware recommenders (top box) can be extended with privacy (bottom box) to enhance traditional similarity driven collaborative filterers (middle box). Computational modules are depicted in gray boxes, while inputs and outputs are depicted in white. Architecture is adapted from Massa and Avesani [16].

Similarity metric, helps finding similar users (or neighbors), which is typically *Pearson Correlation Coefficient* (PCC) [28]. Rating predictor module predicts ratings based on a weighted sum of ratings given by similar users to the items [16]. Architecture of a trust-aware recommender is made up of a *trust metric*. The difference between two architectures is made in how neighbors are discovered and how their weights are identified. This can be done through similarity module or through trust metric module. The combined output, estimated trust network along with user similarity, can be used to generation ratings predictions.

To introduce the notion of privacy within this architecture, we need to justify what we mean by privacy first. As we construct a trust network of users for propagation and aggregation of trust values within our framework, we propose for adoption of notion of social privacy across the network of users.

To define privacy in our terms, we approach the notion of privacy in following terms:

Any user a, who wants prediction [in a Trust Network] does not have to reveal her rating vector during recommendation process and other users in the recommendation process cannot learn any rating value of a and the rated and/or unrated items of user as rating vector.

Taking into account this adaptation of privacy, we have extended existing architecture in Fig.1 to include the architecture of a *private trust aware recommendation system.* As depicted, a privacy-preserving trust-aware CF can be composed of two main modules: a *private trust metric* and a *private rating predictor*. This architecture takes a masked rating set as input and generates a private trust estimation which is used by a private rating predictor, which in turn combined with rating prediction module from the pure CF step can generate predicted rating matrix. Obviously, a private trust recommender is actually composed of both pure and trust-enhanced recommendation modules and inputs. To understand how this architecture can be realized, we adapt this architecture onto our recommendation framework, presented in previous section. In our framework, the private trust metric is realized through *data disguising* and *private trust calculation* steps.

To achieve our privacy aim, we propose to use *z-scores* of user ratings instead of their actual preferences. The z-score [6] of an item indicates how far and in what direction, that item deviates from its distribution's mean, expressed in units of its distribution's standard deviation. In this work we utilize z-score transformation for normalizing data. Since z-score values have zero mean, we can hide their value by adding random numbers from a distribution with zero mean and predefined standard deviation. As a result, users will all make computations with their z-scores instead of their actual ratings. To improve privacy level, we propose to hide unrated items of users, too. Users fill f (related with users density) percent of their unrated items with random numbers having the same perturbation properties as employed for z-score disguising. Since having rating for an item shows that user has purchased this item, to hide which items are really purchased by any user, users fill f percent of their unrated items with random numbers.

The flow of data through the architecture is as follows:

At first, original rating profiles are masked. The decentralized protocol for data disguising is presented in the next section. The matrix generated from data disguising step is fed into the private trust module then. To realize the private trust module, at first, the trust is formalized to adapt to calculation with respect to z-scores. This is followed by disguising z-scores with randomness values. In our approach, users will apply the protocol described in following section to disguise their private vector during the process of trust estimation and computation. When users finish calculation of z-scores and data disguising, we can compute the trust between them. Within this framework, we have adopted Neil Lathias [18] trust formalization for calculating interpersonal trust. The process of calculation is done in a decentralized fashion on each users side. At the end of this step, trust values are returned and stored in the trust network. After this step, we use the private rating predictor module to produce final predictions, which generates the user-item ratings matrix, as the output of the recommender. To do so, we have adopted the *PCC* [28] along with the interpersonal trust values between users from previous step and generate *referrals*. Since users get normalized values from this output, the results are de-normalized. These steps are explained in more detail in the following section.

3.2 Data Disguising and Private Trust Computation

Once any user a, requests a recommendation from social recommender system, a need to disguise his/her rating vector to protect his/her own privacy. Therefore a firstly normalizes his/her rating vector and add randomness to normalized data. In our scheme users follow below protocols and processes for perturbing their data, estimating private trust, and producing private predictions.

Data Disguising Protocol

The procedure for data disguising is as follows:

1. Each user computes their ratings z-score values.
2. Each user u selects a β value and then they uniformly randomly select standard deviation of the random numbers (σ_u) over the range $[0, \beta]$. They also compute number of rated items (num_{rat}) and number of unrated items (num_{unrat}) in their rating vectors.
3. Each user computes her density value d and she selects a random integer value f showing the percentage of unrated items to be filled between 0 and a number associated with d such as $d/2$, d, or $2d$. Each user u randomly selects f percent of their unrated items.
4. Users can utilize uniform distribution or Gaussian distribution to generate random numbers. To select the distribution, users decide a θ value over the range $[0, 1]$ and they uniformly randomly select a random number r_u over the same range with θ.
   a. If $r_u \leq \theta$, the users generate random numbers having *uniform distribution* having interval $[-\delta, \delta], \delta$ can be 1, 2, 3 or 4.
   b. Otherwise, they use *Gaussian distribution* with zero mean and standard deviation σ_u.
5. After selecting distribution, each user generates $(num_{rat} + num_{unrat} * \frac{f}{100})$ random numbers having zero mean and σ_u. To disguise rated items, each user add num_{rat} of random numbers to rated items z-score values and they fill randomly selected $num_{unrat} * \frac{f}{100}$ unrated items with other random numbers.
6. Each user saves their masked z-score vectors.

Private Trust Estimation

As mentioned private trust module allows us to generate private trust values. Assume there are two users; u_a and u_b. We formalize the trust between them as follows:

$$z(u, i) = \frac{R_{u,i} - \overline{R_u}}{\sigma_u} \tag{1}$$

Where $R_{u,i}$ is the true rating of user u on item i, $\overline{R_u}$ is the mean rating of user u, σ_u is the standard deviation of user us ratings and $z(u, i)$ is the z-score value of user u on item i.

$$z'(u, i) = z(u, i) + r_{u,i} \tag{2}$$

Where $r_{u,i}$ is the random number generated by u to disguise z -score of item i and $z'(u,i)$ is the masked value of $z(u,i)$. When users finish calculating z-scores and data disguising, they compute trust among other users using Eq. 3:

$$T'(u_a, u_b) = 1 - \frac{\Sigma_{i=1}^{n} z'_{u_a,i_i} z'_{u_b,i_i}}{z'_{\max} * n} \tag{3}$$

This equation is an adapted formalization of trust proposed by Lathia et al. [18], which is based upon difference of a users rating and its recommenders rating to their common item(s).

Here $T'(u_a, u_b)$ is the estimated (private) trust between respective users; u_a and u_b, $z'(u_a, i_i)$ is the masked z-score of user u_a for item i_i and $z'_{\max}$ is maximum masked z-score.

When u_a and u_b computes trust, they follow the steps below:

1. u_a and u_b decide which half they will operate on.
2. u_a and u_b send the parts that they will not operate on to each other.
3. When users receive related part of other users vector, they compute sub-result of trust using Eq. 3.
4. Each user sends her sub-result to other user.
5. They compute trust value between each other by summing up sub-results.

Private Recommendation Prediction Process

To produce recommendations, Eq. 4 can be used. Since, z-scores are used instead of actual ratings in our scheme, when users finished computing trust; they use Eq. 4 to produce referrals, as follows;

$$p(a,i) = \frac{\Sigma b \in N_{(a,i)} z'_{b,i} * T'(a,b)}{\Sigma b \in N_{(a,i)} T'(a,b)} \tag{4}$$

Users get a normalized rating value when they use Eq. 4. To obtain actual rating value, users need to de-normalize result of Eq. 4 by using Eq. 5.

$$p(a,i) = \overline{R_u} + \sigma_u * p \tag{5}$$

Where $P(a,i)$ is the denormalized prediction for user a and item $i, \overline{R_u}$ is the mean rating for user u, σ_u is the standard deviation of user us ratings and p is the referral value from previous step.

3.3 Defining and Inferring Optimal Privacy Set

It is accepted that privacy and accuracy are conflicting goals in the context of personalization and Collaborative filtering recommenders [12]. This conflict becomes more imminent in the presence of trust. Utilization of interpersonal trust aims at increasing [25], or maintaining the overall accuracy [22]. Trust metrics along with other factors such as neighbors list size at each step of trust estimation increase or maintain the accuracy of predictions. This is while increasing the amount of perturbations leads to further information loss. To protect the private data, the level of perturbation is vital. If the amount is too low, the masked data still discloses considerable amounts of information; if it is too high, accuracy will be very low [12]. If we take into account the configurations that affect the privacy mechanism at one hand, and take into account the configurations affecting trust in another hand, we can argue that an optimal setting can be defined where privacy and accuracy can be both maintained at the same time. From the perspective of achieving an optimal result, problem space can be seen as an optimization design space. Within this design space we have j real parameters corresponding to trust mechanism configurations, while we have k different criteria corresponding to privacy mechanism configurations. In this space we take privacy enhancing mechanism as a function p, which generates privacy configurations set. For example, as we have used perturbation to protect private data, these operators become the distributions we have utilized for adding perturbations to the user rating profiles. We refer to this set as a *Privacy Configuration Set (PCS)*:

$$\prod_{i \in I} pcs_i = \{p : I \rightarrow \bigcup_{i \in I} pcs_i | (\forall i)(p(i) \in pcs_i)\}$$

Since in theory, we can have an infinite number of parameters, we consider that at each time only j parameters are taken into account:

$$\Phi_j : \prod_{i \in I} pcs_i \rightarrow pcs_j$$

$$\Phi_j(p) = \{(pcs_1, pcs_2, \ldots, pcs_j) | pcs_1 \in \Phi \wedge pcs_1 \in \Phi \ldots \wedge pcs_j \in \Phi\}$$

In a similar fashion, trust enhancing mechanism, can be taken as a function t, which generates trust configurations set. For example, configurations that our trust-aware recommender uses to enable social network mediated trust inference are trust metric and the size of the trust lists, at each step. We refer to this set as a *Trust Configuration Set (TCS)*:

$$\prod_{i \in I} tcs_i = \{t : I \rightarrow \bigcup_{i \in I} tcs_i | (\forall i)(t(i) \in tcs_i)\}$$

Since in theory, we can have an infinite number of parameters, we consider that at each time only k parameters are taken into account:

$$\Psi_k : \prod_{i \in I} tcs_i \rightarrow tcs_k$$

$$\Psi_k(t) = \{(tcs_1, tcs_2, \ldots, tcs_k) | tcs_1 \in \Psi \wedge tcs_1 \in \Psi \ldots \wedge tcs_k \in \Psi\}$$

Since we study and analyze configurations from both mechanisms at once, then we need to make a joint set containing members from both sets. As a result we define an ordered set composed from Cartesian product of all privacy configuration sets (PCS), and trust configuration sets (TCS), as follows:

$$\psi = \Phi \times \Psi$$

$$\psi = \{(pcs_1, \ldots, pcs_j, tcs_1, \ldots, tcs_k) | (pcs_j) \in \Phi_j \wedge (tcs_k) \in \Psi_k\}$$

As the goal is achieving acceptable accuracy and respective privacy at the same time then optimization problem becomes multi-objective. As a result, problem of achieving a trade-off between accuracy and privacy in the current context becomes a *Pareto optimization problem.*

Taking into account this fact, we define an Optimal Privacy Set (OPS) as follows:

Definition 1. *Let ψ be the set of all possible joint configurations. There exists a set $\psi\mathbf{i}$, in which all possible joint privacy and trust configurations achieve a decent privacy and accuracy at the same time, in comparison to ψ_i^*, which is the other possible joint configurations. Such set exhibits Pareto optimality. We refer to this set as an Optimal Privacy Set (OPS):*

$$\psi_j \succeq \psi_i^*$$

In other words, among all possible configurations we can always find at least one setting that can either maintain or improve privacy, in the face of accuracy loss. To find such set, following heuristic can be adopted:

Heuristic. *To infer OPS, following heuristic is used:*

1. *Perturbing the overall user data using different PCS settings;*
2. *Observing the framework under variations of TCS;*
3. *Perturbing the sparse user data with PCS inferred from step 2 allows for inferring OPS and finalizing the Pareto optimal setting.*

In the evaluation section, through experiment we show how such set is inferred and justified as the optimal result, which respects the tradeoff we are trying to achieve.

4 Recommendation Framework Evaluation

To evaluate our framework, we have conducted two sets of experiments: First set demonstrates the effect of insertion of random data on accuracy of predictions generated as output of the recommendation system. The second set of experiments demonstrates how filling unrated items with varying f values affect the overall accuracy of recommender system. At the end, we define and infer the optimal privacy set with respect to experiment results. To measure the accuracy of recommendation system, we have utilized MAE (Mean Absolute Error) as respective metric. MAE measures the average absolute difference between predicted rating score made for a specific user and the users actual rating [30]. For these experiments we have used public MovieLens dataset [33]. This dataset contains 943 user rating profiles, with more than 100000 rating values. Rating values are on a 5 point scale. For the first experiment part we have divided the profiles into 80% of data for training purpose and 20% for testing purposes. For the second part we have used 60% of data for training purpose and 40% for testing purpose.

4.1 Accuracy under Overall Masked User Data

We have masked users profiles with different random numbers having Gaussian distribution and Uniform distribution to show effects of distributions on accuracy. To setup the experiment we change two set of parameters: parameters that affect the data disguising operations, and parameters that affect the overall private trust computations. With respect to former, we have tried changing β and δ values while with respect to latter we have tried changing the t (trust metric value), n (neighbors lists size). To compare the results under masked data with results without masked data, MAE for variations of t and n are presented in Fig. 2.

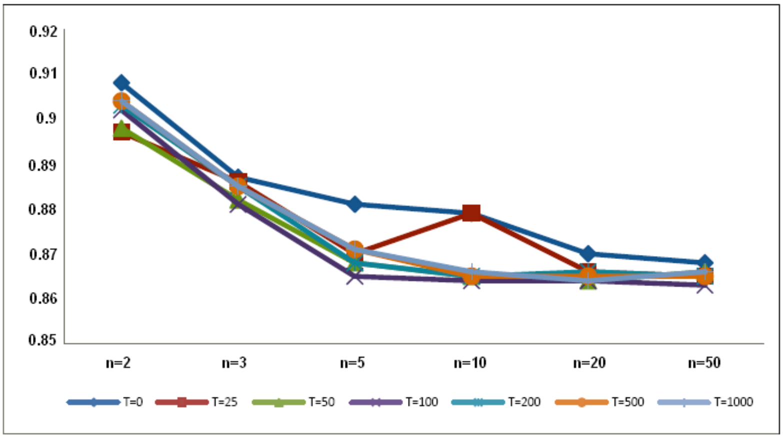


Fig. 2. MAE of recommendation framework, without adding any perturbations [31]

Results of experiments with perturbation on MAE are depicted in Fig.3. Fig 3 plots the effects of random data on MAE with Gaussian distributions (Fig 3.a), and with Uniform distributions (Fig 3.b). In the case of Gaussian distribution we have selected $\{0.5, 1, 2, 4\}$ for β as respective values, and for uniform distributions we have selected $\{1, 2, 3, 4\}$ for δ as respective values. With respect to trust metric t values are selected from $\{0, 100\}$ and for neighbors lists we have tested with list sizes of 2, 3, and 5. In both plots, horizontal axes depict the possible intervals for different distributions of β and δ. Results of MAE experiments clearly state that if we utilize Gaussian distribution for random numbers the higher the β values, the better privacy is achieved and this is due to increasing randomness. We can witness the tradeoff here: The higher the β values the more accuracy we will lose. Results of Uniform distributions also confirm this observation. If we utilize Gaussian distribution for random numbers the higher the δ values, the better privacy is achieved, and the higher the δ values the more accuracy loss we have. With respect to n, we can observe in MAE results that are neither too high and nor too low values for n neighbors list size can give us decent results. This is also the case for t where lowest value ($t = 0$), doesnt give uniform and

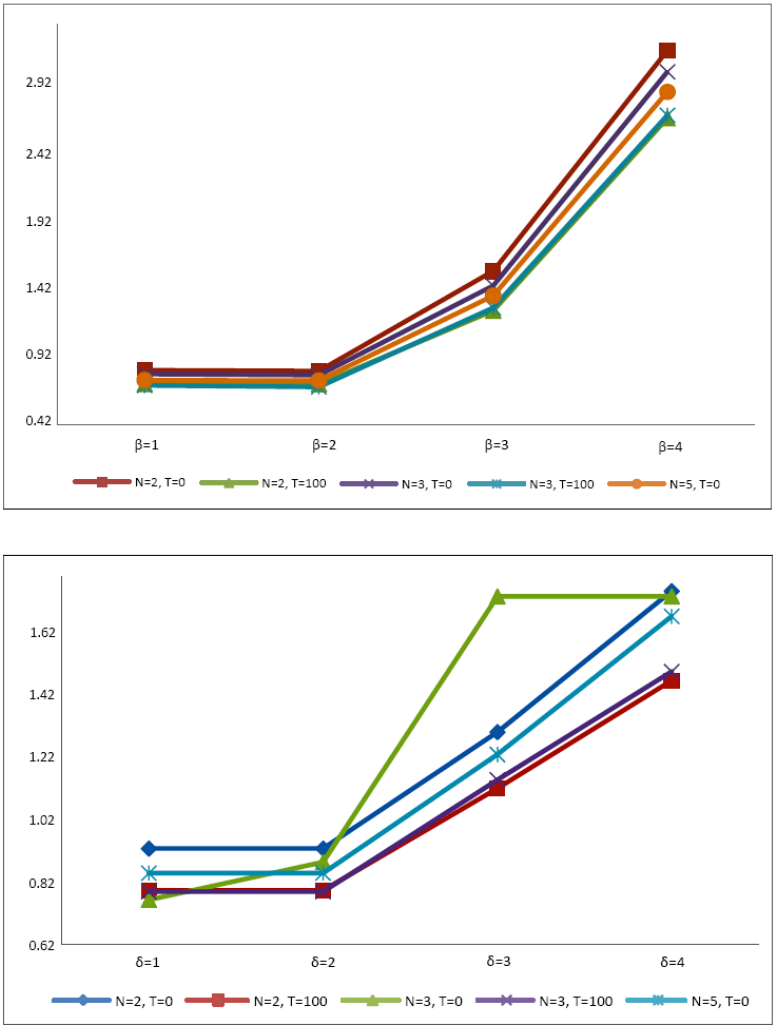


Fig. 3. Effects of adding perturbations on MAE, having Gaussian distribution (a), and having Uniform distribution (b), to user data

consistent result, while highest value for $t(t = 100)$ yields more reasonable MAE results. Overall observation of MAE states that Gaussian distribution seems to be better than uniform distribution for accuracy but they are both useful with selected appropriate β and δ values.

4.2 Accuracy under Sparse Masked User Data

To show effects of filling unrated items with random numbers, we have performed experiments with varying f values.

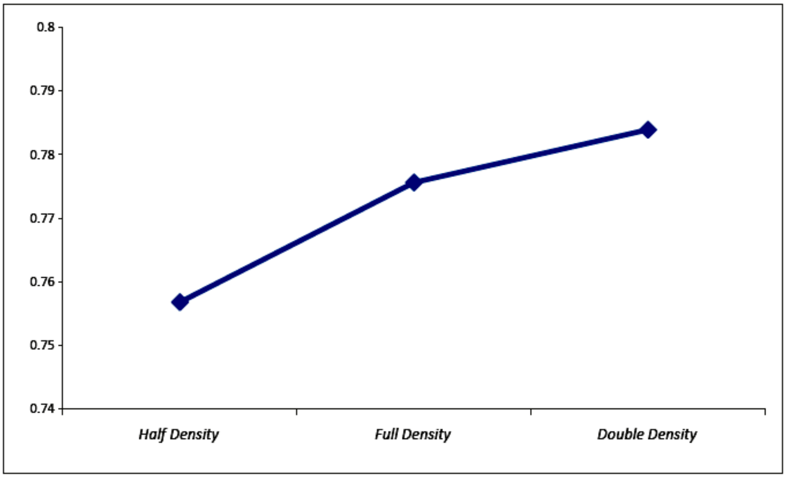


Fig. 4. Filling unrated items with random data having Gaussian distribution with respect to f

In these experiments random numbers having Gaussian distribution with zero mean and standard deviation $\beta = 1$ were used. We selected f values from the intervals $[0, d/2], [0, d]$ and $[0, 2d]$. With respect to f we have depicted resulting MAE in Fig.4.

We can observe from the results that filling unrated items with random numbers provides better privacy, but it decreases accuracy as expected. Also when we increase possible fs interval, we achieve higher privacy level.

4.3 Analyzing Trading-Off between Privacy and Accuracy

Now that the base experiments were presented, we can take into account the privacy and accuracy metrics of current system to define and derive an optimal setting where system exhibits a transparent outcome.

Considering the parameters from both privacy and trust mechanisms, Φ and Ψ are defined as follows:

$$\Phi = \{(\beta, \delta, f)\beta \in [1, 5], \delta \in [1, 5], f \in [0, 2d]\},$$

$$\Psi = \{(n, t)n \in [1, 5], t \in [0, 100]\}.$$

Adopting the formalization introduced earlier, an optimum configuration is a joint setting of $\psi = (\beta, \delta, f, n, t)$ through which we maintain accuracy and privacy at the same time.

As a matter of fact, following the heuristic presented earlier;

1. First we perturb the overall user data using Gaussian and Uniform distributions (δ, β), by comparing the results of MAE of framework under masked

data (Fig.3), we can observe that set of $(\delta, \beta) = (1, 1)$ yield best results as it exhibits the minimal privacy loss. As a result we fix $\beta = 1$, $\delta = 1$ for the next step.
2. In this step we observe the framework under variations of (n, t): With respect to this step, by comparing the results from MAE of framework under masked data (Fig.3), we observe that set of $(n, t) = (3, 100)$, while being fixed on $(\delta, \beta) = (1, 1)$, yields reasonable accuracy, while privacy is maintained. So we fix the current set to $(\delta, \beta, n, t) = (1, 1, 3, 100)$.
3. In the final step we perturb the sparse user data with (δ, β, n, t) inferred from previous step for fine-tuning the privacy. To do so we utilize of different intervals of f with the system being fixed on (δ, β, n, t) configuration from previous step. Through observation of consistent accuracy of different f intervals, we can fine-tune the configuration from previous step and infer an optimum privacy configuration. Taking into account the results (Fig. 4), we observe consistent increase in intervals of f which finalizes the choice of n, t, δ, β and finalizes the results in ordered set of $n = 3, t = 100, \delta = 1, \beta = 1$ and $f = [0, d]$ supporting both accurate and private recommendations:

$$\psi(\delta, \beta, n, t, f) = (1, 1, 3, 100, 0, d)$$

Considering the existing range of ϕ configurations, experiment showed that Pareto optimality holds.

These results were inferred with framework under masked user data. To make sure that optimum result maintains the Pareto optimality effect, we compare the MAE results of non-masked framework (Fig.2) with framework under masked results (Fig.3). In our work we inferred the optimum values for $\beta = 1, n = 3$ and $t = 100$ and for these parameters $MAE = 0.7994$, while for similar parameters without adding perturbations we achieve $MAE = 0.881$, which clearly shows that Pareto optimality holds, while it also shows that we have increased the privacy of the base framework with our architecture.

Further observation shows that our MAE results are still less than results of MAE without adding perturbations. According to (Fig.2), we achieve the best results with $MAE = 0.863$ for $(n, t) = (50, 100)$ and this value is still greater than our optimum value. This observation also states that our result with proposed framework still shows better accuracy than the base framework.

5 Conclusion and Future Work

In this paper we proposed a framework for addressing the problem of privacy in trust recommenders. To overcome this obstacle, first we introduce a framework for enabling privacy-preserving trust-aware recommendation generation. After introduction of architecture, its building blocks and protocols, we pointed out the conflicting goals of privacy and accuracy. Within this context, we showed that a Pareto set can be always be found which can make a tradeoff between these conflicting aims and we presented a heuristic that experimentally infers this set. Through experimentation with predictive accuracy of private trust recommender system, we showed that we can infer such setting that holds even when trust recommender is not under privacy measures. We also showed that privacy increases under proposed framework, while even optimal privacy of our framework is better than the best performance of base framework in its best configurations. As a result privacy can be introduced in trust recommenders and can be optimized to avoid private data loss and at the same time produce accurate recommendations.

As future work, we plan to strengthen our framework against shilling attacks. We will investigate how to extend our scheme when data is collected by a central server

Acknowledgement

This work was partially supported by grant number 621-2007-6565 funded by Swedish Research Council and Grant 108E221 from TUBITAK.

References

1. Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., Riedl, J.: Group-Lens: An open architecture for collaborative filtering of netnews. In: Proceedings of the ACM 1994 Conference on Computer-Supported Cooperative Work, pp. 175–186. ACM, Chapel Hill (1994)
2. Goldberg, D., Nichols, D., Oki, B., Terry, D.: Using collaborative filtering to weave an information tapestry. Communications of the ACM 35(12), 61–70 (1992)
3. Cranor, L.F.: 'I didn't buy it for myself' privacy and e-commerce personalization. In: Proceedings of the ACM Workshop on Privacy in the Electronic Society, pp. 111–117 (2003)
4. Canny, J.: Collaborative filtering with privacy. In: Proceedings of the IEEE Symposium on Security and Privacy, pp. 45–57 (2002)
5. Canny, J.: Collaborative filtering with privacy via factor analysis. In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, pp. 238–245 (2002)
6. Du, W., Polat, H.: Privacy-preserving collaborative filtering. International Journal of Electronic Commerce 9(4), 9–36 (2005)
7. Kaleli, C., Polat, H.: Providing private recommendations using naive Bayesian classifier. Advances in Intelligent Web Mastering 43, 515–522 (2007)
8. Kaleli, C., Polat, H.: P2P collaborative filtering with privacy. Turkish Journal of Electric Electrical Engineering and Computer Sciences 8(1), 101–116 (2010)
9. Kaleli, C., Polat, H.: Providing private recommendations on personal social networks. Advances in Soft Computing 67, 117–125 (2010)
10. Parameswaran, R.: A robust data obfuscation approach for privacy-preserving collaborative filtering. PhD thesis, Georgia Institute of Technology (2006)
11. Eytani, Y., Kuflik, T., Berkovsky, S., Busetta, P., Ricci, F.: Collaborative filtering over distributed environment. In: Workshop on Decentralized, Agent-based and social Approaches to User Modeling, in conjunction with the 10th International Conference on User Modeling, Edinburg, UK (2005)

12. Ziegler, C.-N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: WWW 2005: Proceedings of the 14th international conference on World Wide Web, pp. 22–32. ACM, New York (2005)
13. Golbeck, J.A.: Filmtrust: movie recommendations from semantic web-based social networks. In: Proc. of 3rd IEEE Consumer Communications and Networking Conference, CCNC 2006, Department of Computer Science, University of Maryland, pp. 1314–1315 (2006)
14. Golbeck, J.: Trust and nuanced profile similarity in online social networks, MINDSWAP Technical Report TR-MS1284, University of Maryland, College Park, Tech. Rep. (2007)
15. Massa, P., Avesani, P.: Trust-Aware Collaborative filtering for Recommender Systems. In: On the Move to Meaningful Internet Systems: CoopIS, DOA, and ODBASE, pp. 492–508 (2004)
16. Massa, P., Avesani, P.: Trust Metrics in Recommender Systems. In: Computing with Social Trust, pp. 259–285 (2009)
17. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: IUI 2005: Proceedings of the 10th international conference on intelligent user interfaces, pp. 167–174. ACM, New York (2005)
18. Lathia, N., Hailes, S., Capra, L.: Trust-based collaborative filtering. In: IFIPTM 2008: Joint iTrust and PST Conferences on Privacy, Trust management and Security, Department of Computer Science, University College London, London, UK, p. 14 (2008)
19. Zarghami, A., Fazeli, S., Dokoohaki, N., Matskin, M.: Social Trust-Aware Recommendation System: A T-Index Approach. In: IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, vol. 3, pp. 85–90. IEEE Computer Society, Los Alamitos (2009), doi:10.1109/WI-IAT.2009.237
20. Squicciarini, A., Bertino, E., Ferrari, E., Paci, F., Thuraisingham, B.: PP-trust-X. ACM Transactions on Information and System Security 10(3), 12-es (2007), doi:10.1145/1266977.1266981
21. Singh, A., Liu, L.: TrustMe: anonymous management of trust relationships in decentralized P2P systems. In: Proc. Third Int'l IEEE Conf. on Peer-to-Peer Computing (2003)
22. Lam, S.K., Frankowski, D., Riedl, J.(n.d): Do You Trust Your Recommendations? An Exploration of Security and Privacy Issues in Recommender Systems, pp. 1–15. The New York Times
23. Mobasher, B., Burke, R., Bhaumik, R., Williams, C.: Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. ACM Trans. Internet Technol. 7(4) (2007), doi:10.1145/1278366.1278372
24. Lam, S.K., Riedl, J.: Shilling recommender systems for fun and profit, pp. 393–402. ACM, New York (2004), doi:10.1145/988672.988726
25. Zhang, F.: Average shilling attack against trust-based recommender systems. In: International Conference on Information Management, Innovation Management and Industrial Engineering, vol. 4, pp. 588–591 (2009)
26. Massa, P., Avesani, P.: Controversial users demand local trust metrics: An experimental study on epinions.com community. In: Proceedings of the National Conference on artificial Intelligence, vol. 20, p. 121. AAAI Press/MIT Press, Menlo Park/Cambridge (1999)
27. Hirsch, J.E.: An index to quantify an individual's scientific research output. PNAS 102(46), 16569–16572 (2005)

28. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the l4h Conference on Uncertainty in Artificial Intelligence (UAI 1998), San Francisco, July 24-26, pp. 43–52 (1998)
29. Dokoohaki, N., Matskin, M.: Effective design of trust ontologies for improvement in the structure of socio-semantic trust networks. International Journal On Advances in Intelligent Systems 1(1942-2679), 23–42 (2008)
30. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22(1), 5–53 (2004)
31. Fazeli, S., Zarghami, A., Dokoohaki, N., Matskin, M.: Mechanizing Social Trust-Aware Recommenders with T-index Augmented Trustworthiness. In: Proceedings of the 7th International Conference on Trust, Privacy & Security in Digital Business (Trustbus 2010), in conjunction with the 21st International Conference on Database and Expert Systems Applications (DEXA 2010), Bilbao, Spain (2010)
32. Fazeli, S., Zarghami, A., Dokoohaki, N., Matskin, M.: Elevating Prediction Accuracy in Trust-aware Collaborative filtering Recommenders through T-index Metric and TopTrustee lists. The Journal of Emerging Technologies in Web Intelligence, JETWI (2010)
33. Riedl, J., Konstan, J.: Movielens Dataset (1998)

First-Use Analysis of Communication in a Social Network

Satoko Itaya[1], Naoki Yoshinaga[1], Peter Davis[2], Rie Tanaka[1], Taku Konishi[1], Shinich Doi[1], and Keiji Yamada[1]

[1] NEC C&C Innovation Research Laboratories
8916-47, Takayama-cho, Ikoma, Nara 630-0101, Japan
{s-itaya,n-yoshinaga}@bp.jp.nec.com
[2] Telecognix Corporation
58-13 Yoshida Shimooji-cho Sakyo-ku, Kyoto, 606-8314 Japan
davis@telecognix.com

Abstract. The study of communication activity in social networks is aimed at understanding and promoting communications in groups, organizations and communities. In this paper, we propose a method for the analysis of communication records to extract content-based network activity, with a focus on first-use. Links between people in a social network are defined based on content and temporal relation of messages sent and received. We introduce the notion of first-use, first-use paths, and classes of users based on first-usage. First-use is defined with respect to a specific time period and specific communication content. It refers to the sending of messages containing the specified contents for the first time before being receiving them from any other user in the specified time period. First-use paths are defined as sequences of first-use events in communication networks, and m-ary classes of users are defined recursively as users who receive for the first time from (m-1)-ary users. We present an example of application of the analysis to the email records of a large company.

Keywords: social network, information propagation, e-mail analysis.

1 Introduction

In recent years, there has been much research of communication activity in social networks for the purpose of better understanding communication and developing tools to assist and enhance communication [1][2][3][4]. We are particularly interested in developing methods for analyzing and predicting how communication activity depends on the content of the information, and how communication activity emerges and evolves over time in a social group. In this paper, we introduce a method for the analysis of communication records to extract content-based network activity, with a focus on the first-use of contents. In particular, we apply the method to the study of the well-known email corpus of the Enron company.

Previous works have explored the structure of communications in social groups, and the Enron email corpus in particular. In particular, Kossinets et al [4] have

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 80–88, 2010.

attempted to clarify the structure of the flow of information in a social network by examining the temporal relations of communications to identify potential paths of fastest flow. They applied the method to an analysis of e-mail communication among the faculty and staff of a large university over a two-year period. However, previous works have not examined the issue of first-use of contents and how the evolution of communication activity depends on the content of the communications.

In this paper, we first describe a general framework for analysis of content propagation from a communication record. Communication links between agents are defined based on content and temporal correlation of send and receive of messages, and used to construct networks. We introduce the notions of first-use, first-use paths, and classes of users based on first-usage. Then we apply the analysis to the email records of the Enron company, using keywords to classify content types. We describe and compare communication activity for different types of keywords used in Enron emails.

2 Analysis Model

In this section we describe a basic model for the analysis of a communication record, as a sequence of discrete events ordered in time. We first describe a general model, which could be applied to a variety of different communication media, including mail, telephone calls, email and file sharing. We introduce the following definitions.

Communication Record

A standard communication record is list of directed communication events recorded as tuples: "to", "from", "time", "type". ("to" may be singular or plural, including a "broadcast address".) A standard communication record can be obtained by extracting essential information from a full communication record, such as in a corpus of email files. For simplicity, we shall call a directed-communication event a "message". However, in general, a directed communication could include events such as commenting on a blog or up/downloading videos via a website.

Table 1. Standard communication record format

Time	From	To	Content Type
1	A	B	X,Y,Z
2	B	C,D	X
3	C	A	X,Y
4	D	B	X,Z

Users: Senders and Receivers

A sender (receiver) is someone that sends (receives) a message of a particular type. A user is someone who is a sender or a receiver.

Time-Content Graph (Network)

Users are represented by nodes in a graph. A graph is defined with respect to a content type X and an observation time window. A directed edge, also known as a link, exists between two nodes if there is a message of the specified content X sent from the sender to the receiver during the specified time window.

First-Use Path (First Path)

A first-use path, or simply first path, in a network is a sequence of links corresponding to a temporally-ordered sequence of messages, such that the receiver of each message is the sender of the next message, and each message is the first message sent by the sender after receiving the previous message in the sequence.

A first-use path is a useful feature because it is a bound on the propagation of new contents. If content propagates between two nodes it does not propagate earlier than the first-use path. The first path between any two nodes, if it exists, is unique. If the first path between A and C is via B, then it will always include the first path between A and B, but not necessarily the first path between B and C.

First-Use User Class

We classify the users according to their first use of contents. We call a sender a "primary" sender if they send a message *before* they receive one. Further, we define an 'm-ary' sender ($m > 0$), where $m = 1$ corresponds to a primary sender, and an m-ary sender for $m > 1$ can be defined recursively, as a sender who receives a message from a ($m - 1$)-ary sender *before* they send their *first* message.

Figure 1 shows a communication diagram which illustrates the meaning of first-use in a certain time period. In (a) bold arrows show a temporally ordered sequence of messages between A and D which is the first path from A to D. In (b) chain-arrows show two first-paths, one between A and B, and another between B and D via C.

In the example of Figure 1, nodes A and B are primary nodes, B is secondary, and D is tertiary, Notice that B is a primary sender (1-ary) in this time period, because it sends a message (to C) before it receives from anyone.

Figure 1. First-use paths (a) Bold arrows show the first path from A to D. (b) Chain-arrows show two separate first paths, one between A and B, and another between B and D via C. Labels at the bottom show the first-use class of the nodes.

Finally, we comment that we can generalize the definitions of first-use classes to specify them with respect to a particular time period T, and a particular time window length W. For example, we can say that a user is a primary user in time period T if they did not receive a message in the time interval W before they sent a message for the first time. (In the limit of zero-length time window $W = 0$, all senders are primary senders.)

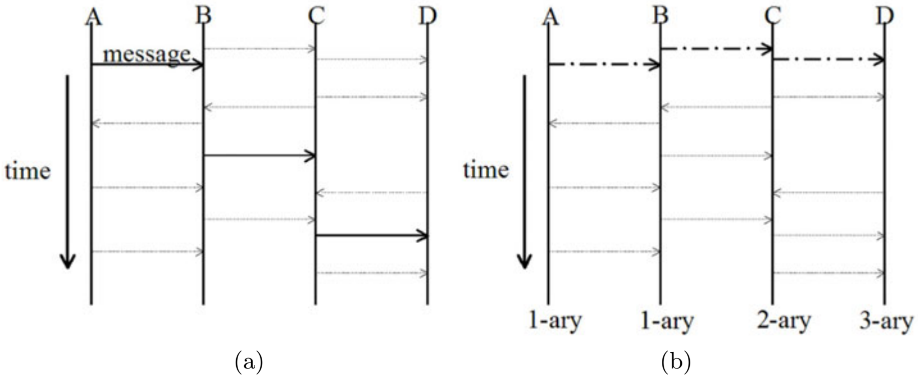


Fig. 1. First paths (a) Bold arrows show the first path from A to D. (b) Chain-arrows show two separate first paths, one between A and B, and another between B and D via C. Labels at the bottom show the first-use class of the nodes.

3 Analysis of Enron Mail Corpus

We now apply the first-use analysis to the email corpus of the Enron company [1][5]. The Enron mail corpus is a unique public source of email records, and a popular tool for research in social network analysis. Enron was an American energy company that collapsed at the end of 2001. The Enron email corpus contains e-mails sent and received by Enron employees during a period of three years, from 1999 to 2001. Part of the e-mail corpus was made public by the FERC (Federal Energy Regulatory Commission) during the ensuing legal investigation. A distilled version has been made available on the web at CMU [5], and this is the version that we use for our study.

First, we extract a message record from the Enron corpus, in the general form as in Table 1. Corresponding to each email, the message record contains a set of time-stamp, sender, receivers and content type characterized by keywords.

The timestamp just corresponds directly to the send or receive timestamp on the Enron email as seen by the user.

The keywords selected are a set of popular keywords - popular in the sense that they are used by a relatively large number of users in a relatively short period of time. We extracted keywords using a method of TF-IDF (Term Frequency-Inverse Document Frequency) [6][7]. We concentrate on the values of the IDF first and then we look at the TF values and their evolution in time. The threshold for the IDF is set to extract words that appear more rarely. Then we examine the TF values of these words during the total time period and select those words that are localized in time. By localized in time we mean that the frequency of use is higher for a short period of time. It is likely that the increase and decrease of activity in such a period is related to correlated activity and propagation effects.

Each email in the communication record can contain any number of keywords. The content type of the email can be classified using one or more keywords. Here

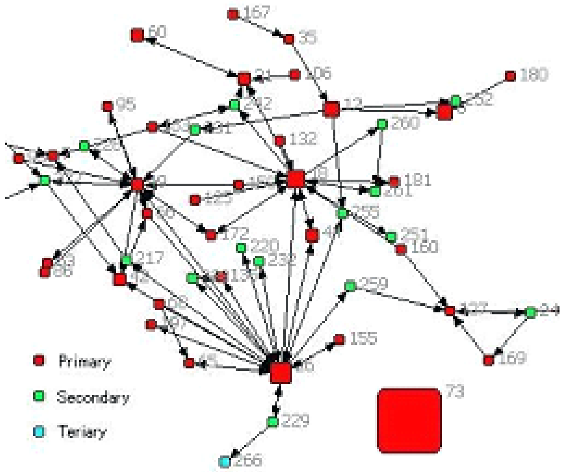


Fig. 2. Part of the network of emails with keyword 'estoppel'

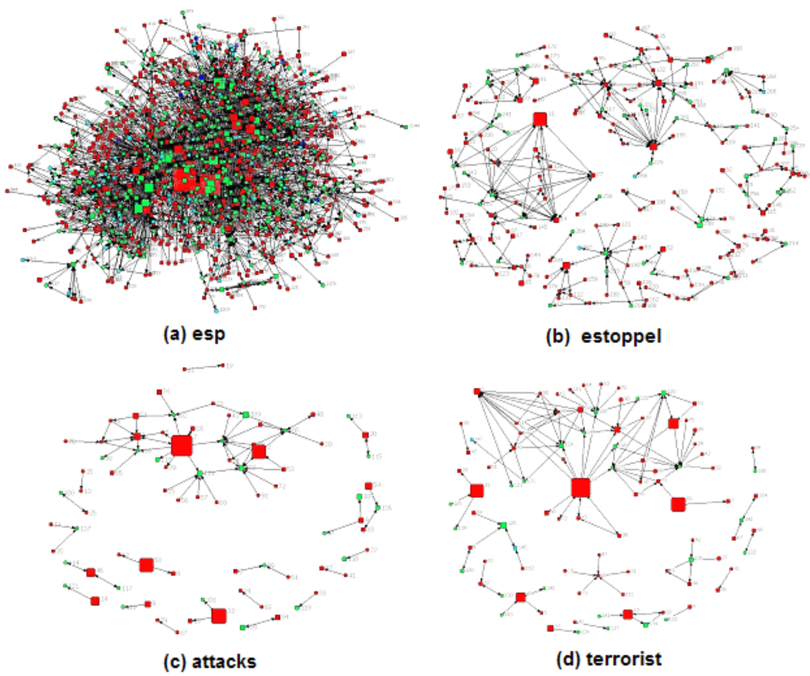


Fig. 3. Comparison of networks for different keywords

in this paper we present examples assuming the content is defined by one keyword. The generalization to cases where the content is defined by sets of keywords is straightforward. Figures 2 and 3 show examples of networks for various different keywords, with nodes classified according to their first-use class.

In Fig. 2 we show an example of a close-up of part of a network, for the keyword 'estoppel'. The word 'estoppel' is a legal term used in disclaimer signature templates in certain types of official Enron emails, particularly emails to persons outside Enron. Since its usage is limited it is a useful reference example. The type of icon in used for nodes in Fig. 2 indicates the first-use type of the user, and the size indicates the number of emails sent. The integer label is the user ID. We can see that the network has primary, secondary and tertiary nodes. We can also see that the network is not fully connected, with one node in particular (node 73) sending a large number of emails to destinations outside of Enron, and none to destinations within Enron. A cross-check with the Enron employee database shows that the large-size primary-user nodes correspond to persons whose job responsibility includes communicating with outside Enron.

Next, in Fig. 3 we show a comparison of examples of networks corresponding to?four different keywords, 'esp', 'estoppel', 'attacks' and 'terrorist', respectively. The word 'esp' means Energy Service Provider and is closely related to business in Enron. The words 'attacks' and 'terrorist' are relevant to the terrorist attacks which occurred in September 2001. It can be immediately seen that there are many more mails with the business-related keyword 'esp' than the other keywords. Below we proceed to analyze the differences in more detail.

One useful measure is the ratio of emails sent by non-primary users. A large proportion of non-primary users indicates the possibility of a large number of users sending as a response to receiving, rather than sending spontaneously. Figure 4 shows the results for a set of 16 keywords corresponding to a wide range of different values of non-primary user ratio. The number of non-primary users is relatively small for all keywords studied, less than 50% with an average value of just 23%. However, the difference in numbers is significant and seems to correlate with the type of contents. The two keywords 'esp' and 'pge' in particular are more than a standard deviation above the average, and the words 'estoppels' and 'enforceable' are more than a standard deviation below the average, so the difference between these two pairs is clearly significant. We can see that keywords known to be related to similar types of content are grouped together in the histogram - for example keywords such as 'esp' and 'pge' which are specifically related to the content of Enron business, keywords such as 'affiliates', 'binding' 'estoppel' and 'enforceable' which are used in email templates, and keywords such as 'terrorist' and 'attacks' which are related to external events and commonly used by people outside Enron. The keywords related to Enron business have the highest proportion of non-primary emails. This indicates a higher tendency for these words to be used after receiving. On the other hand, the keywords that are used in the email signature templates have the smallest ratio of non-primary emails, ie. most users of these keywords are primary users. The keywords 'attacks' and 'terrorist' have also a small ratio of non-primary emails. This seems

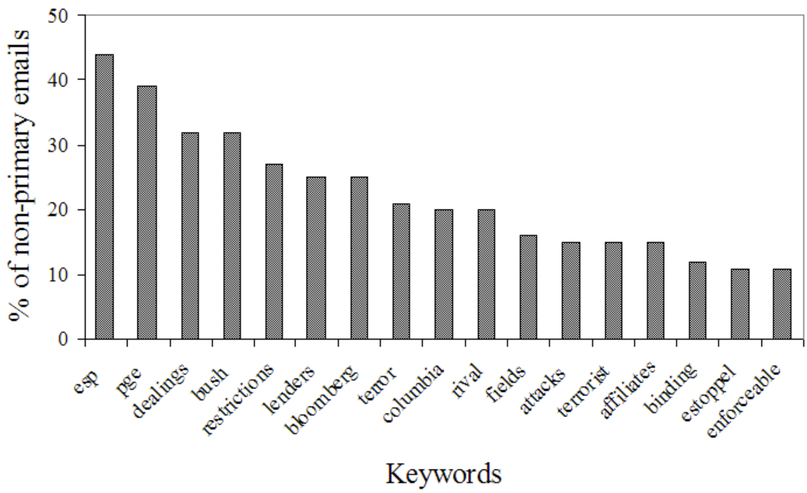


Fig. 4. Dependence of ratio of non-primary emails on content keyword

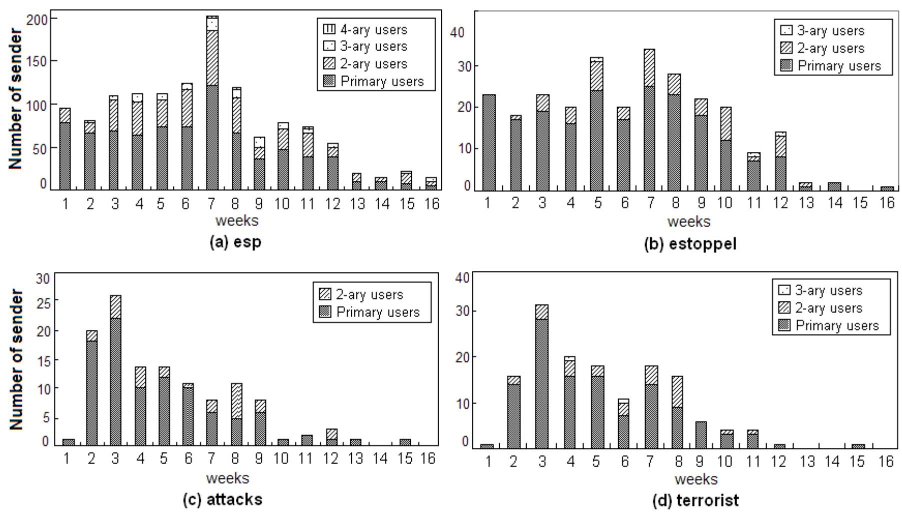


Fig. 5. Temporal behavior of first-use numbers in Enron mail

consistent with the expectation that if words are also used by many people outside the company, it is more likely that users within the company use this keyword before receiving from someone else within the company.

Overall, we conclude that the dependence of ratio of the number of emails sent by non-primary users seems to be a useful measure which correlates with the type of contents.

In order to see changes in the number of first-uses over time, we set a short time unit, such as an hour, a day, a week or a month, and examine the number of new users in each time unit. Each user is counted only once, in the time unit that he sends an email for the first time. If the use of a keyword is random with a constant rate, then we only expect a monotonic decrease of the number of first uses over time. The observation of an increase in number of first uses clearly indicates a localized burst of communication activity related to a keyword.

Figure 5 shows examples obtained from an analysis of a particular 16 week period of the Enron email data, using one week as the time unit. During the period the overall activity in each case first increases and then decreases to a small value, indicating the use is localized in time. However, the difference in the timing of the peaks appears to correspond to differences in the content type. The activity of keywords 'attacks' and 'terrorist' seem roughly similar, with peaks in the third week. The activity of keywords 'esp' and 'estoppel', which are both related to Enron business, also appear correlated in time, with a peak at a similar point in time, around 7 weeks. The keyword 'esp', which has a smaller ratio of primary users, shows a sharper peak of activity. We observe that in general the keywords with the larger ratio of non-primary emails, tend to show activity more localized in time.

4 Discussion

The analysis of the Enron email communications shows that the first-use analysis is able to reveal different patterns of communication activity depending on the type of the contents. In the Enron data, we can see distinct differences in communication activity related to internal business related contents and more general contents. In particular, the analysis shows significantly higher values of non-primary user ratio for job-related contents. This is consistent with a higher ratio of correlated 'responses', which distinguishes job-related communications from content that appears to be mainly random or broadcast activity.

We note that for the application example in this paper we used a simple classification of content based on the use of common keywords, rather than more comprehensive matching of contents. It can be expected that more detailed features of communication activity could be seen with more detailed classification of contents. However, the more detailed the content classification, the smaller the number of emails in each category, making statistical analysis more difficult. The tradeoff between the specificity of communication content and the statistical significance of analysis results is an ongoing challenge for the future.

Finally we comment that the first-use analysis is also useful for detecting and identifying people with high communication potential and the dependence of communication potential on content. For example, the primary use of 'estoppel' is highly correlated with formal communication roles in the company, while the use of the word "terrorist" is not. One area where the analysis of communication potential is now popular is in viral marketing of new products. Communication potential is an important aspect of any social group, and an important area for future applications of first-use analysis.

5 Conclusions

In this paper, we propose a method for the analysis of communication activity in a group of people based on a communication record. In particular, we proposed a method to analyze content-based activity, based on the first-use of contents in communications. Specifically, we introduced the notions of first-use, first use paths, and classes of users based on first-usage, and explained how they can be obtained from communication records.

As an example of application of the analysis based on first-use of contents, we presented results from the analysis of email communications in the Enron company. First, we characterized the contents of email with popular keywords (that is, keywords that are used in many emails in short periods of time), constructed a content-dependent network, and then used the first-use analysis to classify nodes in the network, which revealed patterns of propagation of usage activity. We also showed that the proportions of first-use users was correlated with the type of content, such as email template contents, work-related topics, and general topics.

This application example demonstrates that the approach introduced in this paper is promising for analyzing the features of the dynamics of communications within a social network. It is expected that the ability to identify dynamical features of communication activity will facilitate better communication and cooperation in social groups in the future.

References

1. Diesner, J., Carley, K.M.: Exploration of communication networks from the Enron email corpus. In: Proceedings of SIAM International Conference on Data Mining, Workshop on Link Analysis, Counterterrorism and Security, pp. 3–14 (2005)
2. Jackson, M.O., Yariv, L.: Diffusion on social networks. Economie Publique 16, 69–82 (2006)
3. Lind, P.G., Hermann, H.J.: New approaches to model and study social networks. New Journal of Physics 9, 228 (2007)
4. Kossinets, G., Kleinberg, J., Watts, D.: The structure of information pathways in a social communication network. In: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 435–443 (2008)
5. Cohen, W.: Enron Email Dataset, CMU (2004), http://www-2.cs.cmu.edu/~enron/
6. Salton, G., McGill, M.J.: Information to Modern Information Retrieval. McGraw-Hill, New York (1983)
7. Aizawa, A.: An information-theoretic perspective of tf-idf measures. Information Processing & Management 39, 45–65 (2003)

Label-Dependent Feature Extraction in Social Networks for Node Classification

Tomasz Kajdanowicz, Przemysław Kazienko, and Piotr Doskocz

Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
{tomasz.kajdanowicz,kazienko,piotr.doskocz}@pwr.wroc.pl

Abstract. A new method of feature extraction in the social network for within-network classification is proposed in the paper. The method provides new features calculated by combination of both: network structure information and class labels assigned to nodes. The influence of various features on classification performance has also been studied. The experiments on real-world data have shown that features created owing to the proposed method can lead to significant improvement of classification accuracy.

Keywords: feature extraction, label-dependent features, classification, social network analysis, AMD social network.

1 Introduction

Classification is one of most important concepts in Machine Learning. It is usually based on the data that represents relationships between a fixed set of attributes and one target class. These relations describe each object independently that means no direct correlations between objects in the classification phase are taken into account. An exception may be additional input features, which aggregate information about the entire group a given object belongs to. However, it requires any clustering process be launched before. There are some applications and research methods, especially related to social networks, which are able to produce data with dependencies between labels of interconnected objects, referred as relational autocorrelation [16]. Based on these connections additional input information should be added to the classification process. If the considered objects are humans and the classification is utilized on their profiles then the social network can be extracted from complementary data (different from people's profiles) about common activities and mutual communication [9, 10, 15]. Overall, a social network is a set of nodes (human entities, objects) and node-node relationship between pairs of nodes [18]. According to [17], all network objects may be described by three distinct types of information that can be easily used in label classification: correlation between the object's label (class) and its attributes, correlation between the object's label and the observed (known) labels of other objects in its neighborhood and, consequently, correlation between the object's label and unobserved (unknown) labels of other objects in its neighborhood.

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 89–102, 2010.

Basic task of within-network classification [1, 12] is to assign the correct labels to the unlabeled nodes from a set of the possible class labels. For example, based on the network of communication interactions, it could be determined whether a given company's employee is either an executive or a performer. Willing to obtain the best possible results of classification, all three types of information should be evaluated: nodes attributes (profiles), node-node network relations to the known labels in the neighborhood (labeled neighbors) and relations to the neighboring objects with unknown labels. Main difficulty here is to extract the set of most discriminative features from the network nodes and their connections to achieve the best classification model.

A new approach for network feature extraction is proposed in further sections. Some of these structural features have discriminative distribution, which may directly influence classification performance.

Section 2 covers related work while in Section 3 appears main part of the paper, where a new method for network feature extraction is presented. Sections 4 and 5, contain descriptions of the experimental setup and the obtained results, respectively. The paper is concluded in Section 6.

2 Related Work

In recent years, there has appeared a great number of works describing models and techniques for classification in network data. Analogously to classical machine learning problems, classification in network data requires specialized solutions for feature extraction, high performance supervised and unsupervised learning algorithms, sparse data handling, etc.

In general, network classification problems, may be solved using two main approaches: by within-network and across-network inference. Within-network classification, for which training entities are connected directly to entities, whose labels are to be classified, stays in contrast to across-network classification, where models learnt from one network are applied to another similar network [11]. Overall, the networked data have several unique characteristics that simultaneously complicate and provide leverage to learning and classification. More generally, network data allow the use of the features of the node's neighbors to label them, although it must be performed with care to avoid increase of variance estimation [7].

There have been developed many algorithms and models for classification in the network. Among others, statistical relational learning (SRL) techniques were introduced, including probabilistic relational models, relational Markov networks, and probabilistic entity-relationship models [2, 6, 13, 16]. Two distinct types of classification in networks may be distinguished: based on collection of local conditional classifiers and based on the classification stated as one global objective function. The most known implementations of the first approach are iterative classification (ICA) and Gibbs sampling algorithm (GS), whereas example of the latter are loopy belief propagation (LBP) and mean-field relaxation labeling (MF) [17]. Generally speaking, there exist many pretty effective algorithms of collective classification as well as graph-based semi-supervised learning methods. It refers, especially logForest, a logistic model based on links, wvRN, a relational neighbor model, SSL Gaussian random field model, ghostEdge, combination of statistical relational learning and semi-supervised learning for sparse networks and theirs collective classification supplements [5].

One of the most crucial problems in the network classification is feature extraction. According to [4] the derived features are divided into two categories: label-dependent (LD) and label-independent (LI). Features LD use both structure of the network as well as information about labels of the neighboring nodes labels, e.g. number of neighbors with given class label. Features LI, in turn, are calculated using the network structure only, e.g. betweenness of a node. The LI like features, therefore, are independent from the distribution of labels in the network and might not be informative. However, they can be perfectly calculated regardless of the availability of labels. What is worth mentioning, most of the proposed network classification methods were usually applied to the data sets with very limited access to labels. Their authors assumed that their applications need to deal even with only 1% labeled nodes. This problem is known as classification in sparsely labeled networks [4, 5].

It appears that the majority of network-based structural measures used as features in network classification may be useful and may potentially improve classification performance.

Social networks, being a network representation of interactions between people is a subject of research in terms of classification in networks as well [4].

3 Features Extraction from the Social Network

3.1 General Terms

Let us suppose that a social network is a graph $G = (V, E, X, L, Y, W)$, where V is a set of nodes (objects, social entities); E is a set of edges (connections) e_{ij} between two nodes v_i and $v_j, E = \{e_{ij} : v_i, v_j \in V, i \neq j\}$; X is a set of attribute vectors x_i, a separate one for each node v_i (a profile of v_i), $X = \{x_i : v_i \in V \Leftrightarrow x_i \in X\}$; L is the set of distinct labels (classes) possible to be assigned to nodes; Y is a list of actual labels assignments to nodes, $Y = \{< v_i, y_i >: v_i \in V \wedge y_i \in L\}$; W is a set of edge weights, $\forall w_{ij} \in W \, w_{ij} \geq 0$ and w_{ij} indicates the strength of edge e_{ij}.

Having known the values of y_i for a given subset of nodes $V^K \subset V$, classification may be described as the process of inferring the values of y_i for the remaining set of nodes $V^U, V^U = V \backslash V^K$.

The first step in the process of node classification is a translation of network data into a set of unified vectors, one for each node. A single vector corresponding to node v_i contains all information from x_i as well as some additional information (new attributes) derived by feature extraction methods based on the network profile. Next, the obtained set of vectors is used in classical, supervised classification.

3.2 Features Extraction

Feature extraction from social networks is a general term for methods of constructing variables from the connectivity graph, expressing the position and importance of each node with respect to the others. As mentioned in Section 1, the generated features may be label-independent or label-dependent. For clarity, while describing label-dependent features, it is made a basic assumption in the paper that feature extraction is based only on correlation between the object's label and the observed labels of other objects in its neighborhood see Fig. 1.

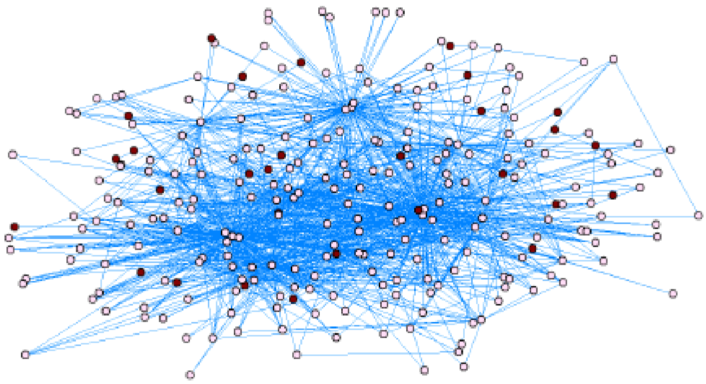


Fig. 1. Example social network with 10% of unlabeled nodes (black circles denote labeled nodes)

Three examples of basic label-independent and three label-dependent features are presented in the following sub-sections, as well as generalization for label-dependent features extraction.

Label-Independent Features

Betweennes Centrality

Betweenness centrality of node v_i pinpoints to what extent v_i is between other nodes. Nodes with high betweennes are very important in the network as other nodes are connected with each other mainly through them. Betweenness centrality $B(G, v_i)$ of node v_i in graph G can be calculated according to the following equation:

$$B(G, v_i) = \sum_{v_j, v_k, v_j \in G(V); j \neq k \neq i} \frac{P(G, v_j, v_k, v_i)}{P(G, v_j, v_k)}, \tag{1}$$

where: $P(G, v_i, v_j)$ - a function returning the number of shortest paths between v_k and v_j in graph G;

$P(G, v_j, v_k, v_i)$ - a function that returns the number of shortest paths between v_i and v_j that pass through v_i in graph G.

Obviously, Equation 1 is calculated only for pairs v_j, v_k, for which there exists a path from v_j to v_k to prevent the denominator from equaling 0.

Degree Centrality

Degree centrality is defined as the number of connections (edges) incident upon a given node. It is the simplest and most intuitive measures that can be used in the network analysis. Nodes with the high degree centrality are recognized as a crucial cog that occupies a central location in the network. Degree centrality $D(G, v_i)$ of node v_i in graph G can be computed using Equation 2:

$$D(G, v_i) = \frac{card(n(G, v_i))}{card(V) - 1}, \tag{2}$$

where: $n(G, v_i)$ - a set of neighboring nodes of node v_i in graph G.

Local Clustering Coefficient

The local clustering coefficient $CC(G, v_i)$ of a node v_i in graph G quantifies how close v_i's neighborhood is to a complete graph, see Equation 3.

$$CC(G, v_i) = \frac{card(R(n(G, v_i)))}{card(n(G, v_i))(card(n(G, v_i)) - 1)}, \tag{3}$$

where: $R(V)$ - an operator returning the number of all connections between nodes from set V.

Label-Dependent Features

While introducing label-dependent features two manners of their formation are proposed. Both of them relay on the idea of selective definition of sub-networks based on the labels assigned to each node. It means that a sub-network for a given label l consists of only those nodes that share label (class) l together with all edges connecting these selected nodes. For that purpose, a new selection operator $O(G, l)$ for graph G and label l is defined. It returns a sub-network G_l labeled with $l : G_l = (V_l, E_l, X_l, \{l\}, Y_l, W_l)$ such that $V_l = \{v_i :< v_i, l >\in Y_l\}$, $Y_l = \{< v_i, y_l >: v_i \in V \wedge y_l = l\}$, $E_l = \{e_{ij} : v_i, v_j \in V_l \wedge e_{ij} \in E\}$, $X_l = \{x_l : v_l \in V_l \Leftrightarrow x_l \in X\}$.

Afterwards, for each sub-network G_l (each label l), new features are computed.

First group of label-dependent features composition is based on new custom measures derived from the interaction between a given node and its neighboring nodes only. The measures take into consideration either the number of connections or their strengths.

Normalized Number of Connections to the Labeled Neighbors

The measure for the normalized number of connections to the labeled neighbors $NCN(G, l, v_i)$ represents the proportion of the number of connections to the neighboring nodes in the sub-network with label $l(G_l)$ by the number of connections to the labeled neighbors in the whole primary graph G (with all labels).

The measure $NCN(G, l, v_i)$ is defined as follows:

$$NCN(G, l, v_i) = \frac{card(n(O(G, l), v_i))}{card(n_L(G, v_i))}, \quad (4)$$

where: $n(O(G, l), v_i)$ - a set of the neighboring nodes for node v_i in sub-network $O(G, l)$, $n_L(G, v_i)$ - a set of v_i's labeled neighbors in graph G, each neighbor must be labeled with any label $l \in L$.

Note that the value of $card(n(O(G, l), v_i))$ is the same as the number of connections between v_i and v_i's neighbors (each v_i's neighbor has one connection with v_i). Similarly, the value of $card(n_L(G, v_i))$ equals the number of connections between v_i and all v_i's labeled (and only labeled) neighbors.

The measure $NCN(G, l, v_i)$ is computed separately for each label l and in general, for two labels l_k and l_m, the value of $NCN(G, l_k, v_i)$ may differ from $NCN(G, l_m, v_i)$.

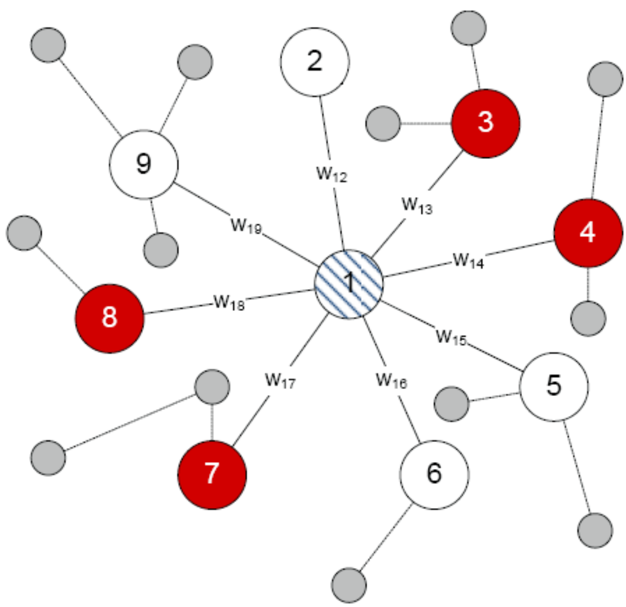


Fig. 2. Feature calculation based on label dependent neighborhood. For each of label class {white, red}w is calculated.

For the example network from Fig. 2, and the measure $NCN(G, 'red', v_1)$ calculated for node 1 in the sub-network with nodes labeled with the $'red'$ class, the value of $NCN(G, 'red', v_1)$ is 4 divided by 8 (total number of nodes in graph G).

Normalized Sum of Connection Strengths to the Labeled Neighbors

The value of the normalized sum of connection strengths to the labeled neighbors $NCS(G, l, v_i)$ is the proportion of node v_i's activity towards v_i's neighbors (measured by the aggregated connection strengths) in the sub-network with label $l(G_l)$ normalized by the equivalent value of strengths to the neighbors with any label in the whole graph G. The value of $NCS(G, l, v_i)$ for graph G and label l is expressed in the following way:

$$NCN(G, l, v_i) = \frac{\sum\limits_{v_j \in n(O(G,l), v_i)} w_{ij}}{\sum\limits_{v_j \in n_L(G, v_i)} w_{ik}}, \tag{5}$$

Similarly to $NCN(G, l, v_i)$, the measure $NCS(G, l, v_i)$ is evaluated separately for each label l and differs for different labels l.

For the network from Fig. 2, the measure $NCS(G, 'red', v_1)$ is computed for node 1 and label (class) $'red'$, as the sum of w_{13}, w_{14}, w_{17}, and w_{18} normalized by sum of all eight connection strengths.

General Method for Label-Dependent Features Extraction

In the domain of social network analysis (SNA), a number of measures characterizing network nodes have been introduced in the literature. Majority of them is label-independent and it is possible to define many methods that will extract label-dependent features based on them. A general concept of creation of any label-dependent feature $M_l(G, l, v_i)$ for label l and node v_i in the social network G applies label-independent feature M to the appropriate labeled sub-network $G_l = O(G, l)$, as follows:

$$M_l(G, l, v_i) = M(G_l, v_i), \tag{6}$$

where: $M_l(G_l, v_i)$ - denotes any structural network measure for node v_i applied to sub-network $G_l = O(G, l)$, e.g degree, betweennes or clustering coefficient;

Obviously, $M_l(G, l, v_i)$ is computed separately for each label l using the appropriate sub-network $G_l = O(G, l)$. In other words, Eq. 6 provides a method for construction of label-dependent version of certain metric.

As an example, the label-dependent clustering coefficient (CC_l) is defined in accordance with Equation 3 as:

$$CC_l(v_i) = \frac{card(R(n(G_l, v_i)))}{card(n(G_l, v_i))(card(n(G_l, v_i)) - 1)}. \tag{7}$$

4 Experimental Setup

4.1 Data Set

The data set used for experiments, "Attendee Meta-Data" (AMD), was downloaded from UCI Network Data Repository (`http://networkdata.ics.uci.edu/`

data.php?d=amdhope). The AMD data set was an output of a project, which used RFID (Radio Frequency Identification) technology to help connect conference participants at "The Last HOPE" Conference held in July 18-20, 2008, New York City, USA. All attendees received an RFID badges that uniquely identified and tracked them across the conference space. The data set contains descriptions of interests of participants, their interactions via instant messages, as well as their location over the course of the conference. Conference attendees were asked to "tag" themselves based on a diverse set of interests. Thanks to location tracking, a list of attendances was extracted for each conference talk. Additionally, participants could email or send a text message to "ping" the people who had similar interests.

In general, the data set contains information about conference participants, conference talks and presence on talks. Initial import contained 767 different persons, 99 talks, 10,110 presences reported during talks. In the cleaning process, these contributors who did not give any information about their interests were excluded from further studies. As a result, 334 persons with 99 lectures and 3,141 presences have left after cleaning.

Afterwards, the social network was build. Ties in the network were constructed based on the fact that participants were present on the same talks. Moreover, strengths of the connections between each pair of contributors were calculated as the proportion of number of talks attended by both participants by the total number of talk presences of the first participant. It provided 68,770 directed, weighted connections, with histogram presented in the Fig. 3.

The raw data contained 4 attributes: 3 nominal (sex, cell phone provider, country) and 1 numerical (age). Additionally, each participant was described by unordered set of interests that in our experiments was chosen as the classification target. Since each network node (participant) could have multiple interests assigned, it was decided to construct 20 separate experimental data sets that formed a binary assignment of each interest. Example networks are presented in

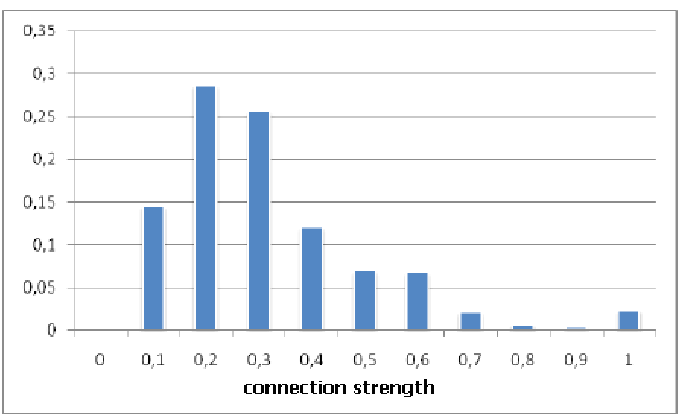


Fig. 3. Histogram of calculated weights in the AMD social network

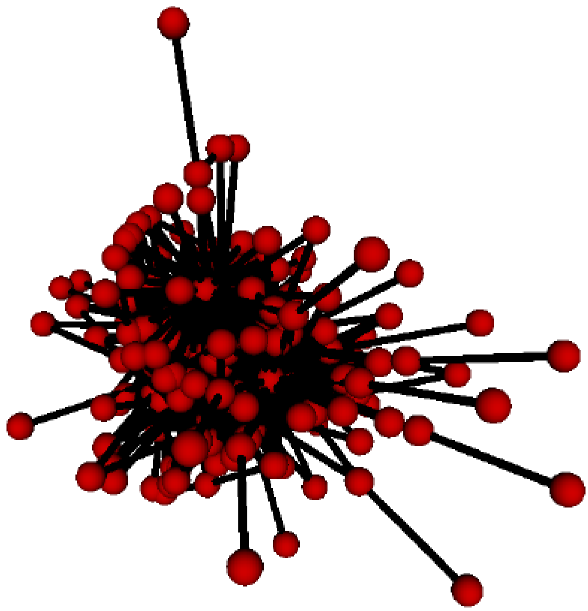


Fig. 4. Visualization of the social network for the activism interest data set based on the class '0' neighborhood using Force-Directed Placement Algorithm [3]

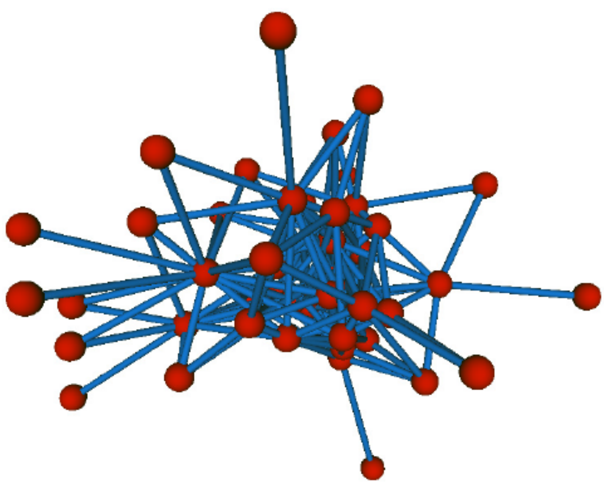


Fig. 5. Visualization of the social network for the activism interest data set based on the class '1' neighborhood using Force-Directed Placement Algorithm [3]

Fig. 4 and 5. For the clarity of the experiment, the binary classification problem was established as it did not contrive a loss of generality of the proposed feature extraction approach.

4.2 Extracted Features

According to the methodology presented in Section 3, 17 attributes were calculated in the experiments, see Table 1.

Table 1. Features used in experiments

No.	Feature	Feature Set
1	age	
2	gender	1
3	county	
4	phone provider	
5	betweenness, Eq. 1	
6	degree, Eq. 2	2
7	clustering coefficient, Eq. 3	
8	normalized sum of connection strengths to the neighbors labeled as '0', Eq. 5	
9	normalized sum of connection strengths to the neighbors labeled as '1' neighbors, Eq. 5	
10	normalized number of connections to the neighbors labeled as '0' neighbors, Eq. 4	
11	normalized number of connections to the neighbors labeled as '1', Eq. 4	
12	betweenness on neighborhood with class '0'	3
13	betweenness on neighborhood with class '1'	
14	degree on neighborhood with class '0'	
15	degree on neighborhood with class '1'	
16	clustering coefficient on neighborhood with class '0'	
17	clustering coefficient on neighborhood with class '1'	
	all above (1-17)	4

Extracted features were grouped in 4 sets. The first contained raw data attributes. In the second there were label-independent network based features. In the third group label-dependent features obtained from proposed method were introduced. The last, fourth group attach all previously introduced features. Finally, the obtained 20 data sets, used in the experiment, may be downloaded from `http://www.zsi.pwr.wroc.pl/~kazienko/datasets/amd/amd.zip` in the arff format. The outcome of performed classification (classification target) was established to predict an interest that a particular person has assigned.

4.3 Classification

Experiments were conducted for 20 data sets using 3 classification algorithms, AdaBoost, Multilayered Perceptron, SVM, with settings presented in Table 2,

Table 2. Features used in experiments

Algorithm	Setting	Value
AdaBoostM1	weight threshold	100
	number of iterations	10
	base classifier	Decision Stump
Multilayer Perceptron	learning rate	0.3
	momentum	0.2
	training time	500
	validation threshold	20
	hidden layers	5
SVM	complexity	1.0
	tolerance	0.0010
	epsilon	10^{-12}
	kernel	polynomial kernel
	exponent	1.0

the same for each of four feature groups (Table 1). Classification was performed in 10% - 90% proportion of labeled and unlabeled nodes, respectively, using 10-cross fold validation.

5 Results

The obtained results have revealed that the average accuracy of classification using various feature sets really differs. As presented in Fig. 6, the average accuracy is greater by about 23% for feature set 3 and 4 compared to set 1 and 2. Simultaneously, F-Measure and precision improves by usage of label-dependent feature sets (set 3 and 4) by 33% and 35%, respectively, see Table 3.

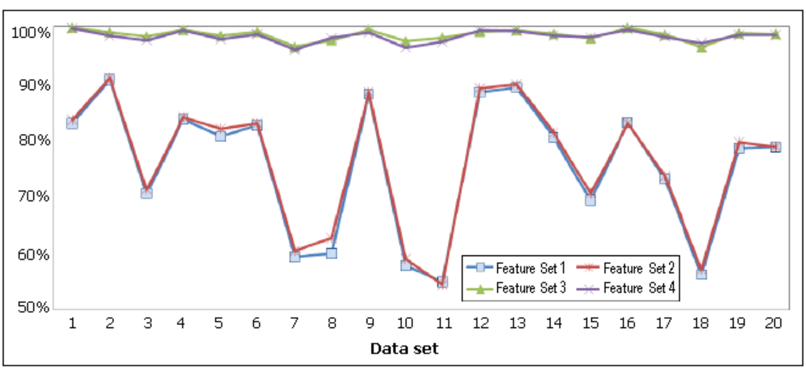


Fig. 6. Average accuracy for 20 data sets using 4 different feature sets

Table 3. Average results of experiments for 20 data sets

Algorithm	Feature Set 1	2	3	4	Measure
AdaBoost	0.76	0.76	0.99	0.98	Accuracy
	0.62	0.63	0.99	0.99	Precision
	0.67	0.68	0.99	0.98	F-measure
Multilayer Perceptron	0.74	0.76	0.99	0.98	Accuracy
	0.67	0.63	0.99	0.98	Precision
	0.69	0.68	0.99	0.98	F-measure
SVM	0.76	0.76	0.98	0.98	Accuracy
	0.64	0.61	0.98	0.98	Precision
	0.69	0.67	0.98	0.98	F-measure

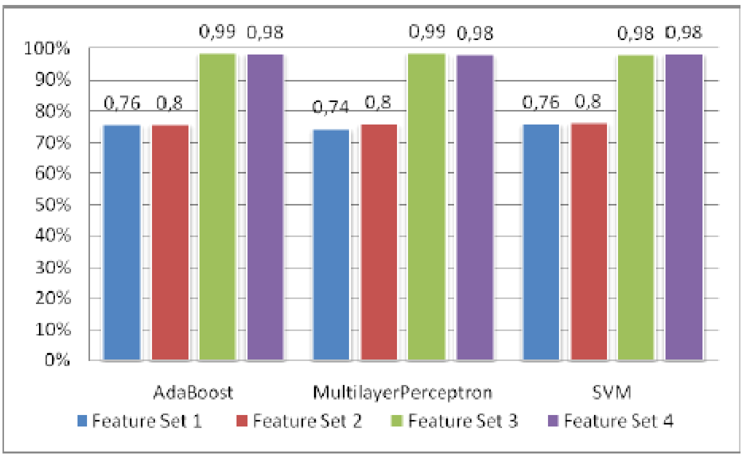


Fig. 7. Average classification accuracy for 4 different feature sets and 20 data sets

Irrespectively of the used feature data set, all utilized classification algorithms: AdaBoost, Multilayered Perceptron, SVM, provide similar results (see Fig. 7).

As shown in Fig. 6, classification based on feature set 3 and 4 seems to be more stable than for feature set 1 and 2. In particular, standard deviation of accuracy for 20 data sets in first case equals 1% and in the second 12%.

Additionally, experiments have revealed that classification based on feature set 4 returns in average worse accuracy than classification based on feature set 3 (see Table 3). Let remind that feature set 4 contains all features from sets 1, 2 and 3. Worse classification performance might be an effect of too many relative poor input features, from which some weaken classification and have contrary discriminative distributions. It refers features from set 1 and 2 that degrade high correlation between output and label-dependent features from set 3. It means that the features extracted from the social network are so good that regular profiles of the tested cases only decrease classification performance and should not be even taken into account.

Owing to the carried out experiments, it is visible that the proposed label-dependent features used in classification undoubtedly provide the best results.

6 Conclusions and Future Work

A new method for label-dependent feature extraction from the social network was proposed in the paper. The main principle behind the method is based the selective definitions of sub-graphs for which new features are defined and computed. These new features provide additional quantitative information about the network context of the case being classified.

According to collected experimental evidences, the proposed label-dependent feature extraction appears to be significantly more effective and improves classification performance in high extent. Obtained, so good, results were even surprising to authors. These results have shown that the new approach to classification extended with features derived from the social network may return very satisfactory and promising outcomes.

It may even happen that the regular features only decrease classification indicators and should be removed from the input feature set. This phenomenon comes probably from the general background of both feature sources. Human profiles are, in fact, the voluntarily collected data whereas social networks are created upon real people activities. There is a crucial difference between a statement "I am interested in mountains" and real information about the mountain climbing. The second is more reliable.

Feature work will focus on further experimentations on the method, especially in terms of its validity for variety of local network measures. Additionally, the proposed feature extraction method will also be examined against the usage of global objective functions for classification. Yet another direction of future studies will be development of new ensemble algorithms, which would have network measures already incorporated, especially based on boosting concept [8].

Acknowledgments. This work was supported by The Polish Ministry of Science and Higher Education, the development project, 2009-11.

References

1. Desrosiers, C., Karypis, G.: Within-network classification using local structure similarity. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009. LNCS, vol. 5781, pp. 260–275. Springer, Heidelberg (2009)
2. Friedman, N., Getoor, L., Koller, D., Pfeffer, A.: Learning probabilistic relational models. In: Proceedings of the International Joint Conference on Artificial Intelligence IJCAI 1999, pp. 1300–1309 (1999)
3. Fruchterman, T., Reingold, E.: Graph Drawing by Force-directed Placement. Software - Practice and Experience 21, 1129–1164 (1991)

4. Gallagher, B., Eliassi-Rad, T.: Leveraging Label-Independent Features for Classification in Sparsely Labeled Networks: An Empirical Study. In: Proceedings of the Second ACM SIGKDD Workshop on Social Network Mining and Analysis (SNA-KDD 2008), Las Vegas, NV (2008)
5. Gallagher, B., Tong, H., Eliassi-Rad, T., Faloutsos, C.: Using ghost edges for classification in sparsely labeled networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 256–264 (2008)
6. Getoor, L., Friedman, N., Koller, D., Taskar, B.: Learning probabilistic models of link structure. Journal of Machine Learning Research 3, 679–707 (2002)
7. Jensen, D., Neville, J., Gallagher, B.: Why collective inference improves relational classification. In: The Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 593–598 (2004)
8. Kajdanowicz, T., Kazienko, P., Kraszewski, J.: Boosting Algorithm with Sequence-loss Cost Function for Structured Prediction. In: Graña Romay, M., Corchado, E., Garcia Sebastian, M.T. (eds.) HAIS 2010. LNCS (LNAI), vol. 6076, pp. 573–580. Springer, Heidelberg (to appear, 2010)
9. Kazienko, P., Musiał, K., Kajdanowicz, T.: Profile of the Social Network in Photo Sharing Systems. In: 14th Americas Conference on Information Systems, AMCIS 2008, Social Network Analysis in IS Research, Association for Information Systems (AIS), Mini-track (2008) ISBN: 978-0-615-23693-3
10. Kazienko, P., Musiał, K., Kajdanowicz, T.: Multidimensional Social Network in the Social Recommender System. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans (accepted, in press)
11. Lu, Q., Getoor, L.: Link-based classification. In: Proceedings of the 20th International Conference on Machine Learning ICML 2003, pp. 496–503 (2003)
12. Macskassy, S., Provost, F.: A brief survey of machine learning methods for classification in networked data and an application to suspicion scoring. In: Airoldi, E.M., Blei, D.M., Fienberg, S.E., Goldenberg, A., Xing, E.P., Zheng, A.X. (eds.) ICML 2006. LNCS, vol. 4503, pp. 172–175. Springer, Heidelberg (2007)
13. Macskassy, S., Provost, F.: Classification in networked data: A toolkit and a univariate case study. Journal of Machine Learning Research 8, 935–983 (2007)
14. McPherson, M., Smith-Lovin, L., Cook, J.: Birds of a feather: Homophily in social networks. Annual Review of Sociology 27, 415–444 (2007)
15. Musial, K., Bródka, P., Kazienko, P., Gaworecki, J.: Extraction of Multi-layered Social Networks from Activity Data. Journal of Global Information Management (to appear)
16. Neville, J., Jensen, D.: Relational dependency networks. Journal of Machine Learning Research 8, 653–692 (2010)
17. Sen, P., Namata, G., Bilgic, M., Getoor, L., Gallagher, B., Eliassi-Rad, T.: Collective classification in network data. Artificial Intelligence Magazine 29(3), 93–106 (2008)
18. Wasserman, S., Faust, K.: Social network analysis: Methods and applications. Cambridge University Press, New York (1994)

Computing α-Efficient Cost Allocations for Unbalanced Games*

Mariusz Kaleta

Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw, Poland
M.Kaleta@ia.pw.edu.pl
http://www.ia.pw.edu.pl

Abstract. We consider a network of agents involved in a common project. Resulting project common cost allocation problem can be modeled as a cooperative game with empty core possible. From social point of view, achievement of subsidy-free allocation may play important role, even at a cost of allocation efficiency. Subsidy-free and α-efficient allocation can be obtained by solving linear programme MASIT. However, to find an unique MASIT solution we use notion of equitable rational preference relation and apply column generation technique. We also show, that there are interesting cases of unbalanced games, and for one of them, TP-game, we present numerical results of our approach.

1 Introduction

We consider a social network made up of $\mathcal{N}$ entities (agents), which are involved in a common project. In this case, usually a total cost of the project is known or can be easily computed. However, assignment of an unique, individual cost share to each involved entity $n \in \mathcal{N}$, may be not evident and easy to calculate. Denote by $c(\mathcal{N})$ a total joint cost of the common project, and by $c(S)$ a joint cost of the project assuming that it is undertaken by subset $S \subseteq \mathcal{N}$ of agents. Then, the problem of a project joint cost allocation consists in determining an allocation vector $x = (x_1, \ldots, x_N)$, where x_n is a cost that entity n has to cover.

With the growing number of social networks, in which the agents usually deal with some common resources, the problem of common cost distribution becomes even more important. A wide spectrum of application is related to the world-wide market liberalization and deregulation processes in many network infrastructure sectors, including power systems, telecommunication, computer, rail and transport networks, water, urban systems and others. In the mentioned areas, the networks of agents supported by market mechanisms that are being implemented, are supposed to realize some social goals, e.g. social welfare and fairness maximization. In such systems, the processes usually are highly complex and are supported by information systems, especially computer decision support

* The research was supported by the Polish National Budget Funds 2010-2013 for science under the grant N N514 044438.

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 103–112, 2010.

systems. An infrastructure, usually modeled as a network, ties the agents and provides common resources needed by the agents to realize they individual goals and also superior, system goals. Thus, one of the key functional elements of decision support systems for infrastructure market mechanisms, is a solution for common resources cost distribution.

Also, a wide spectrum of non market-driven problems related to cost allocation in social network of agents is addressed in the literature and can be met in practice, e.g. in supply chains area [1], natural environment protection, or grid computing [7].

The problem can be modeled as a cooperative TU-game[1] $\langle \mathcal{N}, c \rangle$ with finite set of agents $\mathcal{N} = \{1, .., N\}$, and characteristic function $c : 2^N \to \mathbb{R}$. The cost vector x can fulfill a set of rationalities which allow to determine the characteristic of cost assignment and understanding of the allocation by each agent. First is a *global rationality* (efficiency), which means that the sum of the individual assignments must be equal to the total joint cost:

$$\sum_{i=1}^{N} x_i = c(\mathcal{N}) \ . \tag{1}$$

Group rationality means that for any coalition S, sum of assigned costs of agents that belong to S, must be not higher than $c(S)$ total cost of the project performed by coalition S:

$$\sum_{i \in S} x_i \leq c(S), \forall S \subseteq \mathcal{N} \ . \tag{2}$$

Let us mention that if set S consists in one agent, then the condition (2) becomes an *individual rationality* since it does not allow to allocate on this agent more than he should pay under no cooperation. Both conditions (1) and (2) define the core of the game [3]. Group rationality can be also formulated for dual game, which is profit game $\langle \mathcal{N}, v \rangle$ with characteristic function $v(S) = c(\mathcal{N}) - c(\mathcal{N} \setminus S)$. It results with following inequalities

$$\sum_{i \in S} x_i \geq c(\mathcal{N}) - c(\mathcal{N} \setminus S) \tag{3}$$

which is equivalent to (2) if efficiency condition is satisfied. Inequalities (3) are also known as the *incremental cost test* which is used for testing whether a given allocation is subsidy-free [11]. By satisfying the incremental cost test it is assured that no coalition pays less than a cost increase caused by joining this coalition to the project.

In the literature, a wide stream of works is devoted to problems with nonempty cores and many of researches are focused on proving non-emptiness of the core for particular problems. However, in the paper we show practical and theoretical examples of cost division problems, where related game is unbalanced and its core is empty. In this situation, both conditions for efficiency and group rationality cannot be held simultaneously. Our contribution is (i) in proposing core

[1] A game with transferable utilities.

relaxation which results in α-efficiency, (ii) in linear programming model for finding minimal α-efficient allocation and finally (iii) in an efficient computational algorithm which can find α-efficient allocation for huge number of possible coalitions (we show an example of 50 agents, that is, 2^{50} coalitions). Allocations with relaxed efficiency conditions were considered in some papers, e.g. in [2,4] in the context of telecommunication. However, α-efficiency was incorporated only due to complexity of this constraint and to facilitate computations, thus a deviation from the efficiency was under strong minimization. In this paper we relax the core conditions to satisfy incremental cost test and to obtain a subsidy-free allocation.

2 Unbalanced Cost Games and Core Relaxation

Although in a case of an unbalanced game[2] not every coalition $S \subseteq \mathcal{N}$ has incentives to form, analysis of unbalanced game $\langle \mathcal{N}, c \rangle$ still makes sense. In some situations, some agents can be forced to stay in a grand coalition due to political, legal or other reasons, or they may just have no choice. An example of the last case is a problem of connecting householders to a power distribution grid. Let us consider two householders that need to be attached to the distribution grid. We assume that there is only one provider of electricity transmission services. The situation is illustrated in the Fig. 1.

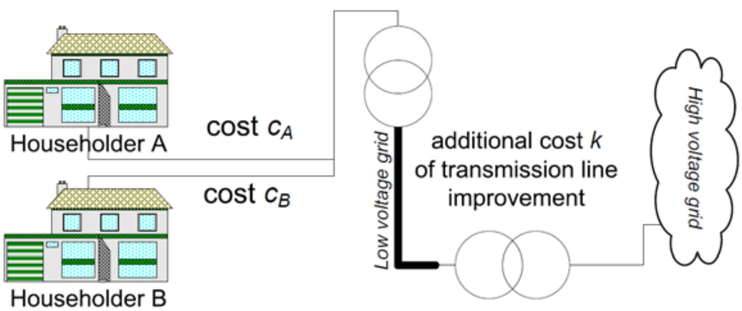


Fig. 1. Power distribution grid problem

Denote by A and B two householders and by c_A and c_B cost of attachment householder A and B respectively. We assume, that if both of the householders are to be connected, then a capacity of low voltage grid is insufficient and must be increased. Cost of the low voltage grid enhancement is denoted by k. Thus, if both householders are attached they must pay total cost of $c_A + c_B + k$. Since this is more costly than hypothetical individual coalitions, the grand coalition is theoretically not stable. However, it is still important problem how the total cost should be allocated. Let us observe that incremental cost test cannot be satisfied

[2] Unbalanced game is a game with empty core.

for $k > 0$ if the allocation is efficient, since we obtain the following contradictory system:

$$x_A \geq c(A) + k \tag{4}$$

$$x_B \geq c(B) + k \tag{5}$$

$$x_A + x_B = c(A) + c(B) + k \tag{6}$$

Second example is a classical transportation problem. We consider a situation presented in the Fig. 2. Producer P_1 has capacity of 5 and unit transportation costs of 2 and 1 for demands D_1 and D_2 respectively. Producer P_2 has capacity of 4 and unit transportation costs of 4 and 5 for demands D_1 and D_2 respectively. D_1 requires 4 units and D_2 requires 3 units of a flowing commodity. Assume, that total cost of satisfying the demands should be allocated on D_1 and D_2.

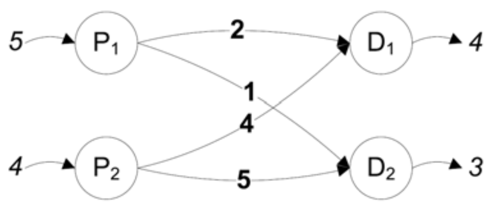


Fig. 2. Transportation problem game

Total cost resulting form transportation cost minimization is equal to 15. Finally, we obtain the following system:

$$x_{D_1} \geq 15 - 3 = 12 \tag{7}$$

$$x_{D_2} \geq 15 - 8 = 7 \tag{8}$$

$$x_{D_1} + x_{D_2} = 15 \tag{9}$$

which is obviously contradictory ($15 \neq 12 + 7$).

This simple example states a basis for modeling many cost allocation problems in a network of agents in area of supply chains. Also other models from social informatics area, resulting in a unbalanced games, can be based on transportation problem case. For example, an extension of work distribution problem in a computational grid, addressed in [7], by a possibility to choose procesor for a given job, may lead to an unbalanced game. In this case, some simply interpretation of Fig. 2 could be as follows: agent D_1 produces four jobs, each job can be performed on procesor P_1 at cost 2 or procesor P_2 at cost 4 and takes one unit time, agent D_2 produces 3 jobs, each job can be performed at procesor P_1 at cost 1 or at procesor P_2 at cost 5 and also takes one unit time. However, computational time on the processors are limited to 5 and 4 respectively. To make the agents feel not to subsidy each other, they must cover the total cost of 19.

In a case of empty core, typical approach is to relax core conditions. Shapley and Shubik [8] have introduced ϵ-core which is based on group rationality relaxation. Nevertheless, relaxation of group rationality means that final allocation becomes not subsidy-free. There is some trade-off between subsidy-free conditions and efficiency. Here we focus on fully subsidy-free allocations, thus we must relax efficiency axiom. It leads us to a concept of α-efficient allocation, in which efficiency condition is substituted by the following one:

$$C(\mathcal{N}) \leq \sum_{i \in \mathcal{N}} x_i \leq \alpha C(\mathcal{N}), \alpha \geq 1 \ . \tag{10}$$

3 MASIT Approach

Because a majority of methods known in the literature assumes that allocation must be efficient, they do not allow to find subsidy-free allocation in a case of unbalanced game. Under the assumption, that incremental cost test (3) must be satisfied for each coalition, we obtain the following system assuring subsidize-free allocation:

$$\sum_{i \in S} x_i \geq C(\mathcal{N}) - C(\mathcal{N} \setminus S) \quad \forall S \subseteq \mathcal{N} \ . \tag{11}$$

Each agent expects the lowest possible charges allocated to him. Thus, the following, introduced by Kaleta in [5], multi-criteria problem MASIT (*Minimal Allocation Satisfying Incremental cost Test*) can be formulated.

Problem **MASIT***:*

$$\min_{x_1, x_2, \ldots, x_n} (x_1, x_2, \ldots, x_n), \tag{12}$$

subject to

$$\textstyle\sum_{i \in S} x_i \geq C(\mathcal{N}) - C(\mathcal{N} \setminus S) \quad \forall S \subseteq \mathcal{N} \tag{13}$$

$$x_i \geq 0 \quad \forall i \in \mathcal{N} \tag{14}$$

An optimal solution of minimization function (12) over simplex formed by set of incremental cost test inequalities usually is not unique. Simple scalarizing function, e.g. minimization of mean or maximum allocation, can result with an allocation easy to question from the point of view of general acknowledged allocation fairness. Particulary, two agents with the same impact on the global costs may be charged different values which minimize scalarizing function. If optimal solution equally treating both of the agents exists, it should be strictly preferred. This leads to an equitable rational preference relation concept based on equitable Pigou-Dalton shifts axiom[3]. Equitable shift consists in worsening better (lower) allocation x_i and simultaneously decreasing higher allocation x_j by relatively small value $\varepsilon > 0$. Allocation vector $x - \varepsilon e_i + \varepsilon e_j$ arising from equitable shift is strictly

[3] Summary of the theory of equitable optimization and various applications relevant to social informatics can be found in [9].

preferred than original vector x. Allocation vector x' equitable dominates vector x'' if it is strictly preferred according to the rational equitable preference relation $x' \succ_w x''$.

Let $\Theta : \mathbb{R}^N \rightarrow \mathbb{R}^N$ denotes an operator of non-decreasing ordering the coordinates of vector x, it means that $\Theta(x) = (\Theta_1(x), \Theta_1(x), \ldots, \Theta_N(x))$, where $\Theta_1(x) \leq \Theta_2(x) \leq \ldots \leq \Theta_N(x)$. Let $\bar{\Theta} = (\bar{\Theta}_1, \bar{\Theta}_2, \ldots, \bar{\Theta}_N)$ be an operator of cumulative ordering, where $\bar{\Theta}_i = \sum_{l=1}^{i} \Theta_l(x)$ for $i = 1, 2, \ldots, N$. Successive coordinates of vector $\bar{\Theta}(\bar{x})$ are the highest allocated value, sum of two highest allocated values, sum of three highest allocated values and so on. Then admissible solution x of task (12)-(14) is equitable effective if and only if it is a solution of the following multi-criteria problem [6]:

$$\min\{\bar{\Theta}(x)\} \tag{15}$$

subject to constraints (13), (14). This problem can be solved by transforming it into single criteria problem by weighing the criteria. This approach is equivalent to OWA aggregation (*Ordered Weighted Average*) applied to task of $\Theta(x)$ maximization subject to (13) and (14) [10]. OWA aggregation can be depicted in computational convenient form of maximization linear combination of cumulative ordered criteria which can be expressed by linear formulas. Finally, the following linear programme can be formulated:

$$\max \sum_{k=1}^{N} w_k (k v_k - \sum_{i=1}^{N} d_{ki}) \tag{16}$$

$$\sum_{i \in S} x_i \geq C(\mathcal{N}) - C(\mathcal{N} \setminus S) \quad \forall S \subseteq \mathcal{N} \quad |\lambda_S \tag{17}$$

$$v_k + x_i \leq d_{ki} \quad \forall k, i \in \mathcal{N} \quad |\lambda_{k,i} \tag{18}$$

$$d_{ki}, x_i \geq 0 \quad \forall k, i \in \mathcal{N} \tag{19}$$

where w_k are nonnegative coefficients, v_k are unlimited variables and d_{ki} are nonnegative variables which represent bottom deviation from v_k.

Notice, that in the consequence of equitable rational preference relation properties, any solution of (16)-(19) satisfies allocation symmetry (anonymous) condition, thus, the allocation is not sensitive for agents renumbering. Constraints (17) result in incremental cost test satisfaction, therefore a solution of problem (16)-(19) is an allocation free from subsidizing. Also properties of positive cost allocation on influential agent and no cost allocation on insignificant agent (*dummy agent*) are satisfied.

MASIT_OWA formulation has huge number of constraints (17) – for every subset of $\mathcal{N}$. It can be observed that for a given allocation usually majority of constraints are not active and stay in a shadow. Only relatively narrow subset of coalitions are important for the final allocation. Thus, column generation-like techniques can be used here. Let us mention that similar attempts have been done for balanced games, e.g. recent one by Drechsel and Kimms based on row generation algorithm [1], however it exercises the efficiency condition and cannot be directly applied to the problem we consider.

Let us denote by λ_S and $\lambda_{k,i}$ the dual variable related to (17) and (18) respectively. Then, dual problem to MASIT_OWA is as follows:

$$\min \sum_{S \subseteq \mathcal{N}} [C(\mathcal{N}) - C(\mathcal{N} \setminus S)]\lambda_S \tag{20}$$

$$\sum_{i \in S} \lambda_{k,i} = kw_k \quad \forall k \in \mathcal{N} \tag{21}$$

$$\lambda_{k,i} \leq w_k \quad \forall k, i \in \mathcal{N} \tag{22}$$

$$\sum_{S \subseteq \mathcal{N}} \lambda_S - \sum_{k \in \mathcal{N}} \lambda_{k,i} \leq 0 \quad \forall i \in \mathcal{N} \quad\quad |\pi_i \tag{23}$$

$$\lambda_S \geq 0 \quad \forall S \subseteq \mathcal{N} \tag{24}$$

$$\lambda_{k,i} \geq 0 \quad \forall k, i \in \mathcal{N} \tag{25}$$

In the dual problem, which is a subproblem in column generation technique, there is great number of columns related to λ_S variables. Let π_i denotes dual variables for constraint (23). In a given iteration of column generation algorithm, only λ_S which are related to introduced coalitions occur in the problem (16)-(19). The variables λ_S occur only in the objective (20) and constraints (23), hence, in the master problem we only need to know prices for constraints (23).

4 Numerical Results

In the subproblem, a new column improving master problem objective and representing some coalition, is searched. Master problem objective can be improved if the expression $\pi^T \lambda_S - c(\mathcal{N}) + c(\mathcal{N} \setminus S)$ is positive for a new column λ_S. For a considered transportation problem case, we formulate following subproblem:

$$\max_{S \subseteq \mathcal{N}} [\pi^T \lambda_S - C(\mathcal{N}) + C(\mathcal{N} \setminus S)] \tag{26}$$

$$\sum_i p_{ij} \leq \overline{P}_j, \quad \forall j \in \mathcal{P} \tag{27}$$

$$\sum_j p_{ij} \geq \overline{D}_i - (1 - z_i)M, \quad \forall i \in \mathcal{N} \tag{28}$$

$$p_{ji} \geq 0, \forall j \in \mathcal{P}, \forall i \in \mathcal{N} \qquad z_i \in \{0,1\}, \forall i \in \mathcal{N} \tag{29}$$

Variable p_{ji} represents commodity flow between producer j and demand i, c_{ji} is a unit cost of flow, $\bar{P}_j$ denotes production capacity of producer j, $\mathcal{P}$ is set of all producers, $\bar{D}_i$ is a level of D_i demand, and M is a great number. There are also binary variables z_i, which indicates whether agent i is present in coalition S. Optimal solution of subproblem (26)-(29) is introduced to master problem.

Our column generation algorithm was tested on allocation tasks based on transportation problem. We generate three sets of data cases, each consists in 20 tasks. Parameters of the problem sets are gathered in Table 1.

Table 1. Data sets characterization

Data sets	Number of producers	Number of customers	Number of arcs	Number of columns
A	10	10	50	1023
B	30	30	450	1.07e9
C	50	50	1250	1.13e15

We made an attempt to generate data sets with as many constraints active for some coalitions as it was possible. Network topology, prices and demands were generated randomly with uniform distribution, (prices from 0-100, demands from 1-200) assuring that each resulting network is connected. Temporary producer capacities where set to optimal values of the problem with unrestricted production capacities. Then we made multiple attempts to reduce production capacities for randomly chosen producer so as to hold feasibility of the problem. Number of attempts for a given producer were equal to doubled number of producers. Each attempt started with assuming half of temporary producer capacity and in case of infeasibility, we were gradually increasing capacity until feasibility recovery.

In all experiments, as a starting set of columns we assumed one column – grand coalition. Computational results are collected in Table 2.

Table 2. Computational results for TP-games

		Data sets		
		A	B	C
Number of generated columns	aver.	30.50	161.20	295.30
	min	21.00	65.00	79.00
	max	52.00	253.00	429.00
Percentage of generated columns [%]	aver.	2.97	1.5E-07	2.6E-11
	min	2.05	0.6E-07	0.7E-11
	max	5.08	2.4E-7	3.8E-11
Time [s]	aver.	8.24	13.66	1015.05
	min	1.58	5.91	154.94
	max	120.00	23.59	3448.00
Excess of allocated cost over joint cost [%]	aver.	34.92	35.30	32.45
	min	4.16	13.87	3.01
	max	82.47	101.62	69.10

Notice, that number of introduced columns is significantly smaller than number of constraints (17), in the first data set this is less than 6%. Due to exponential increase of total number of columns, the percentage of introduced columns is decreasing drastically for bigger problems. Thanks to that, relatively huge problems, for which all coalitions cannot be considered (e.g. Shapley values), are likely to be solved. Computation times are acceptable, however software we have implemented can be still optimized, e.g. eliminating messages output significantly influences on computational times. Last row in tabel 2 collects percentage excess of the sum of allocated values over total joint cost. The average values about 34% for all data sets confirms that generated data sets are not trivial. Additional cost for satisfying incremental cost test can even double total joint cost (maximal values in data set B).

5 Summary

Emptiness of the core of a cooperative game forces a relaxation of the efficiency feature for a subsidy-free allocation. We have shown that an α-efficient allocation, which can also be subsidy-free, can be determined by multicriteria model MASIT. Application of equitable rational preference relation assures that solution is unique, anonymous and fairly treats similar agents. Our computational experiments proved that MASIT_OWA model can be solved efficiently by column generation technique. We have formulated subproblem for transportation problem, however we expect similar results for other LP-games. Let us also mention, that all known for us cost allocation methods do not satisfy subsidy-free conditions in a considered class of problems. Also, in opposition to our approach, many of them cannot be applied to a large number of agents due to computational complexity.

References

1. Dreschel, J., Kimms, A.: Computing core cost allocations for cooperative procurement. Technical report (2007)
2. Feigenbaum, J., Krishnamurthy, A., Sami, R., Shenker, S.: Hardness results for multicast cost sharing. Theoretical Computer Science 304, 2003 (2003)
3. Gillies, D.B.: Solutions to general non-zero-sum games. Annals of Mathematics Studies 40, 47–85 (1959)
4. Jain, K., Vazirani, V.: Applications of approximation algorithms to cooperative games. In: STOC 2001: Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing, pp. 364–372. ACM, New York (2001)
5. Kaleta, M.: Subsize-free cost allocation method for infrastructure market game. In: Proceedings of the 13th IEEE IFAC International Conference on Methods and Models in Automation and Robotics, pp. 1171–1175 (2007)
6. Kostreva, M.M., Ogryczak, W.: Linear optimization with multiple equitable criteria. RAIRO Rech. Opér. 33, 275–297 (1999)

7. Rzadca, K., Trystram, D., Wierzbicki, A.: Fair game-theoretic resource management in dedicated grids. In: CCGRID 2007: Proceedings of the Seventh IEEE International Symposium on Cluster Computing and the Grid, Washington, DC, USA, pp. 343–350. IEEE Computer Society, Los Alamitos (2007)
8. Shapley, L.S., Shubik, M.: Quasi-cores in a monetary economy with nonconvex preferences. Econometrica 34(4), 805–827 (1966)
9. Wierzbicki, A.: Trust and Fairness in Open, Distributed Systems. Studies in Computational Intelligence, vol. 298. Springer, Heidelberg (2010)
10. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decisionmaking. IEEE Trans. Syst. Man Cybern. 18(1), 183–190 (1988)
11. Peyton Young, H., Russell Sage Foundation: Equity: in Theory and Practice / H. Peyton Young. Princeton University Press, Princeton (1994)

Recommendation Boosted Query Propagation in the Social Network

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

Institute of Informatics, Wrocław University of Technology
Wyb.Wyspiańskiego 27, 50-370 Wrocław, Poland
{grzegorz.kukla,kazienko,piotr.brodka,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

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 113–124, 2010.

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.

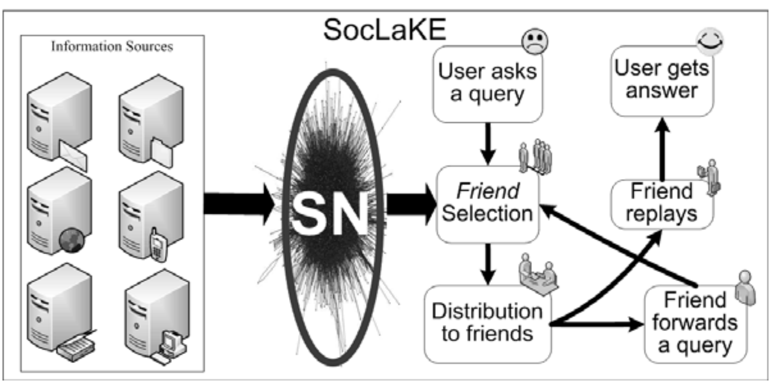


Fig. 1. Social Latent Knowledge Explorator

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.

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 Fig. 2.

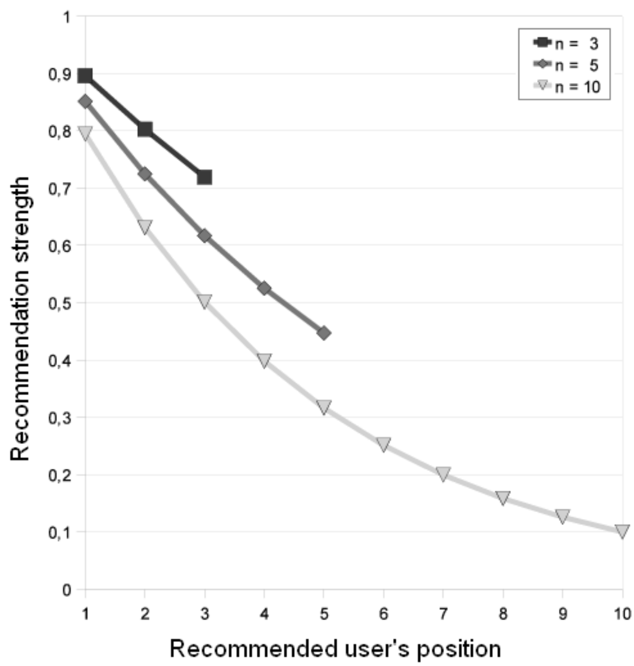


Fig. 2. The relation between recommendation strength and user's position on the recommendation list

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 $\pi_{ij}^{'}$ 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 $\pi_{ij}^{'}$ 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 $\pi_{ij}^{'}$ 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 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.

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

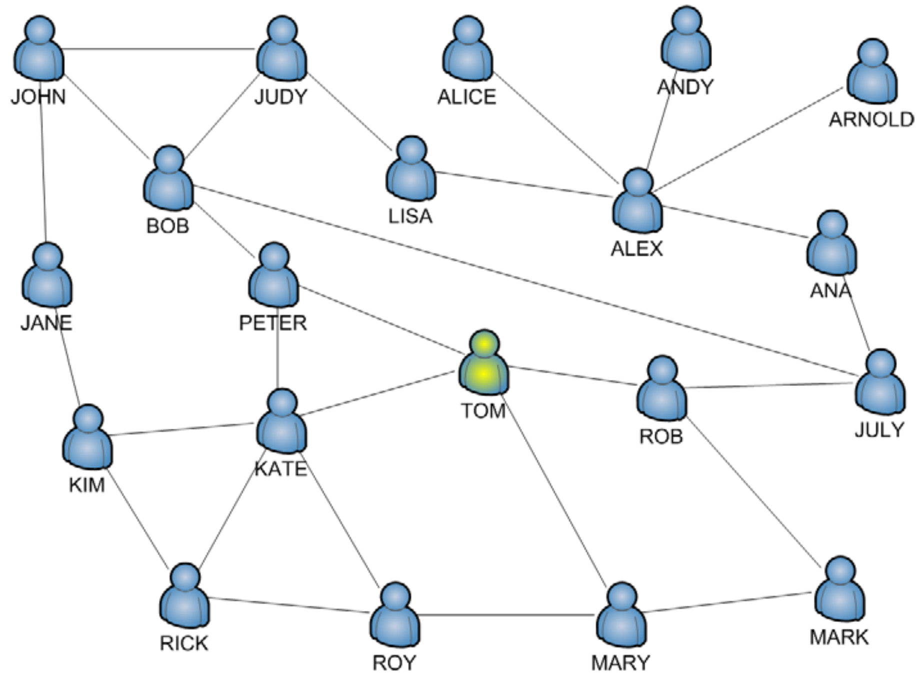


Fig. 3. An example social network

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 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).

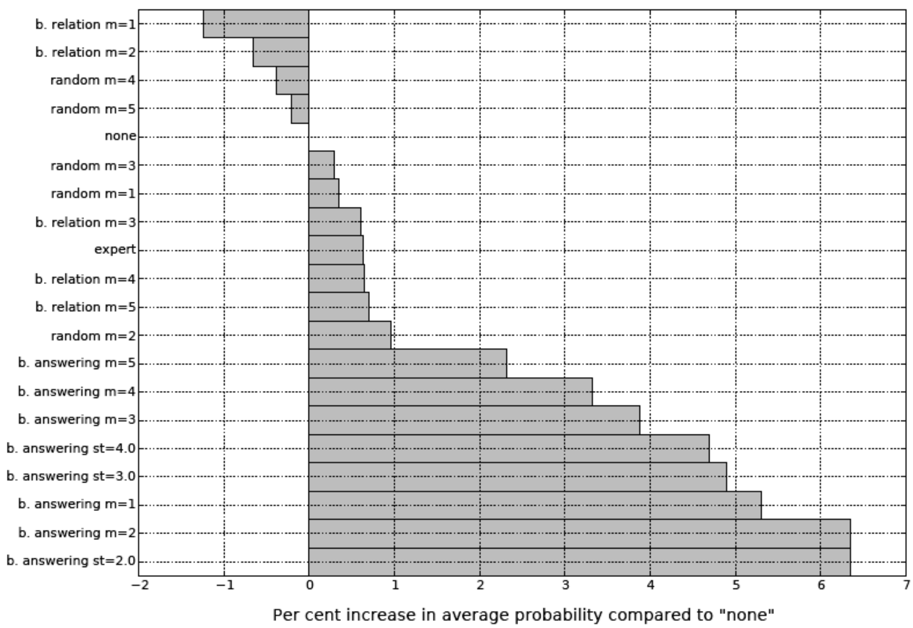


Fig. 4. Recommendation strategies efficiency comparison

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

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.

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 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

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

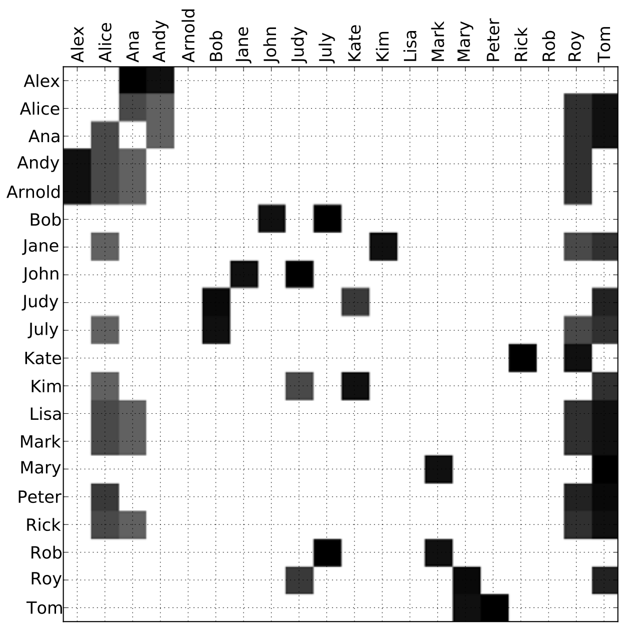


Fig. 5. Recommendation generated by "best answering st=2" strategy. The darker the field the stronger the recommendation of user in column to user in row.

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 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.

Acknowledgement

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

References

1. Balog, K., Azzopardi, L., de Rijke, M.: Formal models for expert finding in enterprise corpora. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 50. ACM, New York (2006)
2. Banerjee, A., Basu, S.: A social query model for decentralized search. In: Proceedings of the 2nd Workshop on Social Network Mining and Analysiss. ACM, New York (2000)

3. Banerjee, A., 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., Maglio, P.P., Cozzi, A., Dom, B.: Expertise identification using email communications. In: Information and Knowledge Management, pp. 528–531 (2003)
6. Carrington, P.J., Scott, J., Wasserman, S.: Models and Methods in Social Network Analysis. Cambridge University Press, Cambridge (2005)
7. Davitz, J., Yu, J., Basu, S., Gutelius, D., Harris, A.: iLink: search and routing in social networks. In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, p. 940 (2007)
8. de Oliveira da Silva, F.E., Motta, C.L.R., Santoro, F.M.: A Social Matching Approach to Support Team Configuration. In: Fonseca, B. (ed.) CRIWG 2009. LNCS, vol. 5784, pp. 49–64. Springer, Heidelberg (2009)
9. DiMicco, J.M., Yankelovich, N.: Constellation: Using Visualization to Find the Path to Experts. Poster Presentation at the ACM GROUP Conference, New York (2007)
10. Iwata, T., Saito, K., Yamada, T.: Modeling user behavior in recommender systems based on maximum entropy. In: Proceedings of the 16th international conference on World Wide Web, p. 1282. ACM, New York (2007)
11. Kazienko, P.: Web-Based Recommender Systems and User Needs - the Comprehensive View. Frontiers in Artificial Intelligence and Applications 181, 243–258 (2008)
12. Kazienko, P., Musiał, K., Kajdanowicz, T.: Multidimensional Social Network in the Social Recommender System. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans (accepted, in press, 2010)
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), 1009–1018 (2002)
15. Montaner, M., López, B., de la Rosa, J.L.: A Taxonomy of Recommender Agents on the Internet. Artificial Intelligence Review 19, 285–330 (2003)
16. Rosner, D., Grote, B., Hartman, K., Hofling, B., Guericke, O.: From natural language documents to sharable product knowledge: a knowledge engineering approach. The Journal of Universal Computer Science 3, 955–987 (1997)
17. Schafer, J.B., Konstan, J.A., Rield, J.: E-Commerce Recommendation Applications. Data Mining and Knowledge Discovery 5, 115–153 (2001)
18. Scott, J.: Social network analysis: A handbook, 2nd edn. Sage Publications Ltd., London (2000)
19. Shami, N.S., Yuan, Y.C., Cosley, D., Xia, L., Gay, G.: That's what friends are for: facilitating'who knows what'across group boundaries. In: Proceedings of the 2007 International ACM Conference on Supporting Group Work, pp. 379–382. ACM, New York (2007)
20. Stefanelli, M.: Knowledge management to support performance-based medicine. Methods Inf. Med. 41(1), 36–43 (2002)
21. Strogatz, S.H.: Exploring complex networks. Nature 410, 63–65 (1998)

Social Manipulation of Online Recommender Systems

Juan Lang, Matt Spear, and S. Felix Wu

University of California, Davis
jilang@ucdavis.edu, batman900@gmail.com, wu@cs.ucdavis.edu

Abstract. Online recommender systems are a common target of attack. Existing research has focused on automated manipulation of recommender systems through the creation of shill accounts, and either do not consider attacks by coalitions of real users, downplay the impact of such attacks, or state that such attacks are difficult to impossible to detect. In this study, we examine a recommender system that is part of an online social network, show that users successfully induced other users to manipulate their recommendations, that these manipulations were effective, and that most such manipulations are detectable even when performed by ordinary, non-automated users.

1 Introduction

Recommender systems are a common component of the online experience today, helping users find interesting content on sites like Slashdot and Digg, as well as guiding buyers to items on sites like Amazon, or to sellers on sites like eBay. Because a high rating confers advantages to the rated item or user, it's unsurprising that manipulation of recommender systems is a common problem. Most existing work has focused on detecting automated shill attacks. Detecting such attacks is certainly necessary, and in some settings, only wide-scale automated attacks are likely to be effective, e.g. when the rating an item receives is unbounded. For example, on eBay or Amazon, a purchaser may wish to choose not just the most highly rated seller or item, but the one with the most positive interactions. In such a setting, an attacker would need to create not only many positive ratings, but also to refresh them. Nonetheless, there are many settings in which ratings are bounded, e.g. the top rated items of the day, or the most popular item in a group. Alternatively, we could assume that existing shill detection and prevention techniques have removed automated attacks, and only social engineering attacks are possible. And in fact, documented attacks on eBay suggest that social engineering attacks have taken place[1], while on Amazon at least one publisher attempted to engage in such an attack[2]. Given that social engineering attacks have taken place, we wish to ask, are they effective? That is, do they result in any advantage to the attacker? And, are they detectable?

[1] http://www.auctionbytes.com/cab/abn/y03/m09/i17/s01
[2] http://www.insidehighered.com/news/2009/06/23/elsevier

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 125–139, 2010.

In this work, we present evidence of such social engineering attacks taking place within one online social network, Buzznet. The form of the attack is simple: the attackers simply asked other users to rate her[3] entry. We show that the attack was successful: in two separate contests, each of the top rated entrants sent hundreds to thousands of requests. Moreover, we show that in most cases, the attack is detectable, even without knowledge of the messages exchanged between the users.

The remainder of this paper is organized as follows: We define the problem more formally in Section 2. We describe related work in Section 3. We describe Buzznet and our data collection in Section 4. We describe attack detection in Section 5. We conclude in Section 6.

2 Background

An online recommender system is a system that combines user-provided ratings of items to provide an aggregated rating to other users. Not every user rates every item. For example, not every user may be interested in every item, or users may not have incentive to rate every item. The items being rated can be of any kind, the only requirement is that there is some variation in users' opinions of them. In some systems, users provide recommendations of a subset of available items, and the system recommends other items based on the recommendations given. For example, on Amazon, viewing an item also displays other items often purchased in conjunction with the item, and YouTube displays a selection videos "Recommended for You" based on videos the user has viewed previously. In other systems, ratings are not based on the viewing user's recommendations, but are global recommendations.

Another difference between recommender systems is the type of rating users can provide. In some systems, users provide a value within a limited scale, e.g. a value of 1 to 5, or a choice of "Like" or "Dislike". In others, including in many online social networks, a rating is only positive: users can vote for an item, and the best-ranked item is the one that receives the most votes. Examples of such systems include Facebook, Digg, and Buzznet. The focus of this work is on global, vote-based recommender systems.

Naturally, the users associated with the items being rated have an incentive to boost their own items' ratings, or to lower those of their competitors. We say an item's rating is *manipulated* if a user succeeds in raising or lowering the item's rating. We say an item is *socially manipulated* if a user induces other users of the system not under his or her direct control to rate the item such that the item's rating is manipulated. (We assume without loss of generality that a user cannot rate his or her own item.) This could be by rating an item higher or lower than they ordinarily would, or by rating an item they would not otherwise have rated.

Ultimately, the consumer of recommendations must take action on the recommendations, e.g. by purchasing an item, watching a video, etc. That is, there is a human judge who will choose among the top k items. We say a global top-k

[3] We arbitrarily use female pronouns for attackers.

recommender system is *r-fair* if at least r fraction of the top k items are not manipulated. The aim of this work is to ensure a $\frac{1}{2}$-fair system by detecting indirect evidence of social manipulation, i.e. without access to the requests asking users to rate an item, such that at least $k/2$ of the top k items are not manipulated. By doing so, we hope that the human judges will be able to separate the good entries from the manipulated ones, such that the outcome is also fair. We also aim to do so in a way that has a very low false positive rate: the harm from disqualifying legitimate, unmanipulated entries seems greater than the harm from allowing a manipulated entry to be judged side by side with an unmanipulated one.

3 Related Work

The first recommender system was Tapestry [5], and it spawned many variations. Resnick and Varian [16] give an early survey of recommender systems, and already begin to discuss the incentives for users to provide ratings to recommender systems, as well as the problem of bias in recommendations.

Lam and Riedl [8] and O'Mahony et al [13] separately introduce the shilling attack on recommender systems. Lam and Riedl describe two forms of automated attack: the RandomBot attack, and the AverageBot attack. In the RandomBot attack, a shill account provides random ratings for items other than the target, then a high rating (for a push attack) or a low rating (for a nuke attack) for the target. In the AverageBot attack, the shill account provides ratings equal to the average rating for all items other than the target, and high or low ratings for the target. O'Mahony et al describe a particular form of nuke attack, in which the attacker rates the two most popular items with high rating, and the target item with a low rating. Chirita et al [2] use statistical differences between the ratings provided by ordinary users and those provided by RandomBot attackers to discover attackers. They claim that the attacks will be generated using automated profiles, because large scale success could not be achieved using manual rating. Several works have improved their results by applying more advanced classifiers, and building more advanced automated attacks against which they test their work, e.g. [1], [11]. We present evidence that the attacks we consider are not carried out by automated profiles in Section 4.3.

Resnick and Sami [15] limit the amount of influence an attacker can have by including the reputation of each rater when computing the final score for an item. They note that one cannot distinguish between a rater who provides bad information on a single item from a rater who simply has an unusual opinion. This work avoids the problem by finding evidence that the item's rating is being manipulated, without concerning itself whether each rater is honest or malicious.

A different approach to combating recommender system manipulation is to include a trust value for each recommender [12], [10]. Such an approach may work if the users responding to requests to rate an item give generally poor ratings, but not if the raters give generally good ratings. Golbeck suggests combining trust with a social network in order to present different recommendations to different users [4]. Such an approach could improve ratings to individual users, but does not help choose globally recommended items. On the other hand, De Figueiredo and Barr [3] show that any global trust function is exploitable unless it is also trivial, i.e. based on direct experience. Our work takes a different viewpoint: if a global trust function is what's desired, and manipulation is unavoidable, may fraud be detected?

The Buzz system on Buzznet is similar to Digg, which has seen some analysis. Lerman and Galstyan [9] point out that the most popular items on Digg tend to receive early votes from beyond the submitter's friends, and therefore that knowing the social distance between the early Diggs an item receives and its submitter can predict the item's eventual popularity. Hsu et al [7] use Support Vector Regression to predict the popularity of Digg submissions based on a large selection of features, focusing on correctly predicting the item's popularity rank. Neither study considers the performance of their classifiers in an adversarial environment.

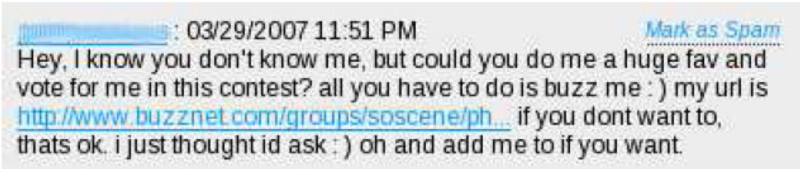
4 Buzznet

Buzznet is an online social network with the usual features: users create profiles, add friends, and post photos and other items. They can also vote for one another's posts by "Buzz"ing them. There are several incentives for manipulating an item's Buzz count: Buzznet has prominent links for the "Most Buzzed" items posted on any given day. Anyone wishing to promote themselves–e.g. a band or a celebrity–would want their posts to be visible as often and as prominently as possible. Another reason for manipulating the Buzz count is to increase one's chances of winning Buzznet contests. Periodically, Buzznet runs contests which users enter by submitting an item–a journal entry, a photo, or a video. The rules of each contest vary, but in some of them, the entrants who had accumulated the highest Buzz count by the end of the contest period were selected as finalists, with the winner(s) chosen by members of the Buzznet staff from among the finalists.

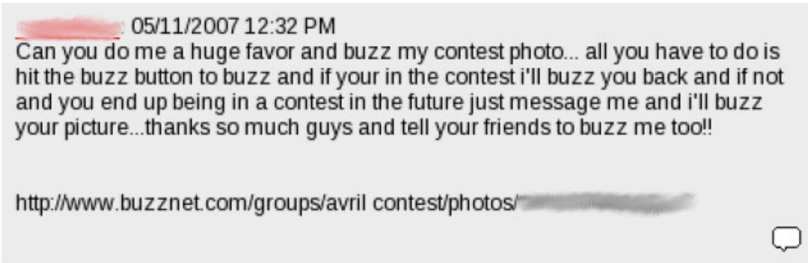
In order to look for evidence of manipulation of the contests, we performed a BFS-based crawl of the Buzznet social graph, until we had obtained the largest connected component, containing approximately 750,000 users and 9 million directed edges. For each of these users, we collected each of the public notes–posts from other users–each user had received, as well as all of the photos each user posted, and the comments, Buzz count, and number of views each photo received. In all, we retrieved approximately 5 million notes, 4 million photos, and 4 million photo comments.

4.1 Detecting Requests for Buzz

In our collected notes, we found many notes asking their recipients to vote for or Buzz an item. One such request is shown in Figure 1(a). Another way in which users asked other users to vote for them was to comment on a photo the targeted user posted, typically offering to trade Buzz. An example of a comment request is shown in Figure 1(b).



(a) Sample note



(b) Sample photo comment

Fig. 1. Sample "Buzz Me" requests

In order to find users who sent many such requests, we searched for notes and photo comments containing URLs whose target was a photo posted on Buzznet. We also searched for comments containing key phrases used in comment requests, including "Buzz me", "Buzz mine", and "Buzz for buzz?". Our identification method isn't perfect: it may have false positives, e.g. a user talking about his or her own photo with another user but not asking the user to Buzz it. In order to reduce false positives, we filtered the users into those who sent at least a quarter of all their messages containing a photo URL or containing the phrases frequently seen in "Buzz me" requests, and who sent at least 100 messages containing such a request. Collectively we refer to this set of users as Buzz Me spammers. Our identification method may also have false negatives, which we will revisit when we discuss our results in Section 5.

4.2 Impact on Contest Results

Some users sent hundreds or even thousands of requests, yet since Buzznet does not reveal who Buzzed which item, we can't know with certainty that the recipients of these requests Buzzed the requested item. Nevertheless, there are indications that the requests were successful. For example, the users may have responded positively, either in a comment to the photo or in a note to the user. In order to determine whether the requests resulted in a photo getting buzzed, we first investigated the responses the photos and their spammers received, then whether there was a correlation between the comments received and the photo's Buzz count.

For each identified request, we looked for a comment or a note from the requestee within one month after the request was made. In total, 14% of the requestees responded, and 79% of these responses contained the word "buzzed". We also examined the Buzz Me spammers' coverage of their commenters, i.e. the fraction of commenters to their photos that had been sent a note prior to commenting. Overall, 43% of the commenters had received a note from the Buzz Me spammer prior to commenting. That is, many of the recipients of requests responded affirmatively to a Buzz request, and a substantial portion of a photo's commenters had been asked to Buzz the photo.

We then looked at Pearson's correlation coefficient [6] between the number of comments a user received on any photo and the total Buzz each user received, both for the Buzz Me spammers and for the population as a whole. For the population as a whole, the correlation coefficient was 0.85, which is suggestive of a relationship between Buzz and comments, but not conclusive. On the other hand, for the Buzz Me spammers the correlation coefficient was 0.995. This very strong correlation seems to imply that many of the comments were associated with a Buzz. Thus, it seems very likely that the requests succeeded in increasing the Buzz of the targeted items. The next question is, did they impact the contests in which the spammers entered?

In order to answer that question, and to see whether requesting Buzz from other users appeared to violate any contest rules, we chose the two contests whose entrants sent the most requests. In each of the contests, the users were asked to submit a photo, and at the submission deadline, the entrants whose photos had received the most Buzz would be selected as finalists, from whom some humans (usually, Buzznet staff members) would select the winner or winners. The two contests were the "I'm So Scene" contest[4] and the "Designed by Avril" contest[5]. Hereafter, we refer to these as Contest 1 and Contest 2, respectively.

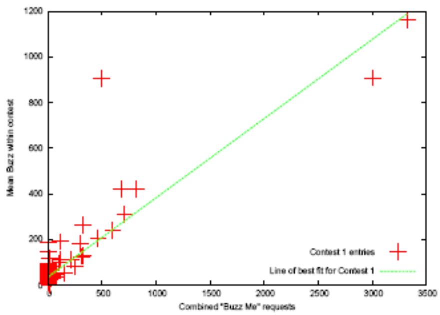
For each of these contests, we computed Pearson's correlation coefficient between the total number of Buzz requests sent, either as a note or as a comment, and the requester's mean Buzz within the contest. The correlation coefficient for Contest 1 was 0.891, and the correlation coefficient for Contest 2 was 0.869. Figure 2 shows a plot of the entrants' mean Buzz counts vs. the total requests sent for Contest 1, and Figure 2 shows the same plot for Contest 2. As the data show, with rare exception, only those sending large numbers of requests prospered in these contests. But were they cheaters?

Table 1. Note social distance (excluding infinite)

	Mean	Std. Dev.	Median
All contest entrants	1.30	0.52	1
Buzz Me spammers	1.98	0.47	2

[4] http://www.buzznet.com/groups/soscene/

[5] http://www.buzznet.com/groups/avrilcontest/



Contest 1 Contest 2

Fig. 2. Mean Buzz count vs. total "Buzz Me" requests

Contest 1. In this contest, one winner and two runners up were chosen from an unknown number of the top Buzzed entries. The contest organizers encourage entrants to "get your friends to BUZZ your entry," so requesting Buzz from users isn't expressly forbidden. On the other hand, entrants did not restrict themselves to asking their friends, as Table 1 shows: for all notes sent by any contest entrant, the median undirected social distance between sender and recipient was 1, whereas for the Buzz Me requests, the median distance was 2.

Of the top 10 most Buzzed entries, only one had sent no requests asking for Buzz. The remaining sent between 376 and 3750 requests apiece. The winner and runners up had each sent over 700 requests for Buzz. In this contest, only those who asked for Buzz had a chance of winning.

Contest 2. In this contest, the winner did not appear among the Buzz Me spammers. Yet a visit to the contest's forum shows this post from a Buzznet staff member:

> The Winners were not chosen by any of the staff here at Buzznet. There are a panel of judges that selected the top 10 models.

In reply, several people complained about an apparent change in the rules. One such reply, from a user who had sent nearly 700 requests, is:

> Then why did the rules at the beginning of the contest say the top 10 buzzed photos would win?

A visit to the Internet Archive[6] shows that, prior to the contest deadline, the rules stated that the 10 entries with the highest Buzz would be finalists. Yet the next stored copy in the Internet Archive, from July 2, 2007, shows a list of 10 finalists, only one of whom had sent Buzz requests in any significant volume (73 requests). None of the top 10 Buzzed photos, nor any of the top 10 requesters,

[6] http://www.archive.org/web/web.php

was in this list. It appears as though Buzznet became concerned about the impact that requests were having on their contest's rankings, and disqualified many of the entrants. That is, based on the change of the rules on the organizers' part, it appears as though this sort of manipulation was viewed as cheating, and disqualified.

4.3 Ruling Out Automated Manipulation

As we noted in Section 3, much of the existing work on identifying manipulation in recommender systems assumes that the manipulation will be done by automated accounts under the control of the attacker. We wish to show that the manipulation we see was done by ordinary users, not by automated profiles. One justification is that the contest organizers were concerned about cheating, and claim to have disqualified users who were cheating by verifying the IP addresses used for each account[7]. Presumably, accounts logging in from the same IP address were assumed to be under the control of a single individual, and disqualified. The disqualified users do not appear in the contest results we present here, that is, we may assume that certain forms of automated attack have already been removed from the data. Moreover, as we showed earlier in this section, the Buzz count an item received was correlated with the number of Buzz Me requests the item's poster sent. It seems unlikely that a user would send requests to profiles she controlled, so we believe that automated attacks are not likely to have played a role in the remaining entries to the contests.

In order to avoid relying on circular logic, however, we use two features as further justification that the Buzz the contest entries received was not due to automated attack: social distance, and photo comment entropy.

Mean Photo Comment Social Distance. Our intuition behind examining the mean social distance between a photo's poster and the photo's commenters is that an attacker creating automated profiles is unlikely to have left comments on photos from profiles at a similar distance distribution as non-attacker users' photos received. While we would prefer to use the social distance between the photo's poster and the users that Buzz it, the users who Buzz a photo are unavailable (only their number is available.) Because of the strong correlation between comments received and Buzz count, we use a photo's commenters as a proxy for the users who Buzzed the photo. Figure 3 shows the distribution of mean distances between posters and commenters for both the Buzz Me spammers and the non-spammers in the contests. (We use the log_2 of the distance in order to normalize the distance.) Visually, there appears to be a small difference between them. In order to test whether the difference is significant, we performed a Kolmogorov-Smirnoff (K-S) goodness-of-fit test [6] between the two distributions. The p-value from the K-S statistic is 1, that is, the null hypothesis that the two distributions are the same cannot be rejected at any significance level.

[7] http://www.buzznet.com/groups/soscene/forum/topics/10440/to-the-cheaterz/

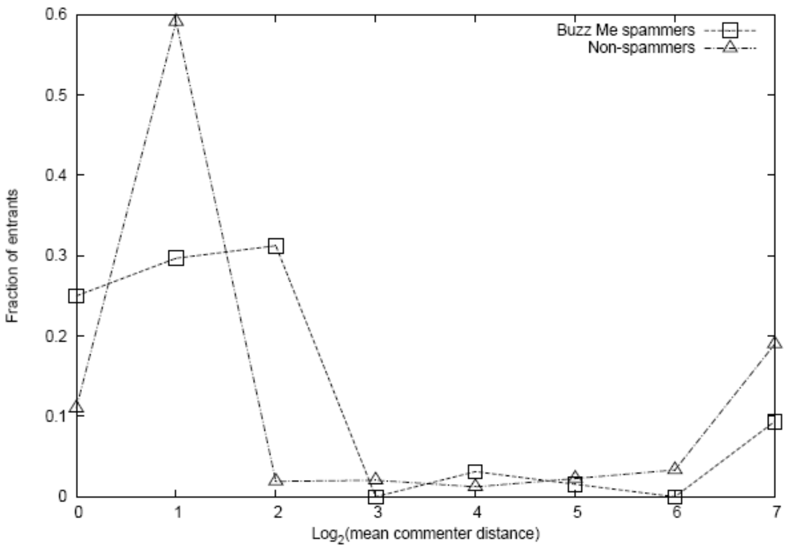


Fig. 3. Photo commenter social distance distribution

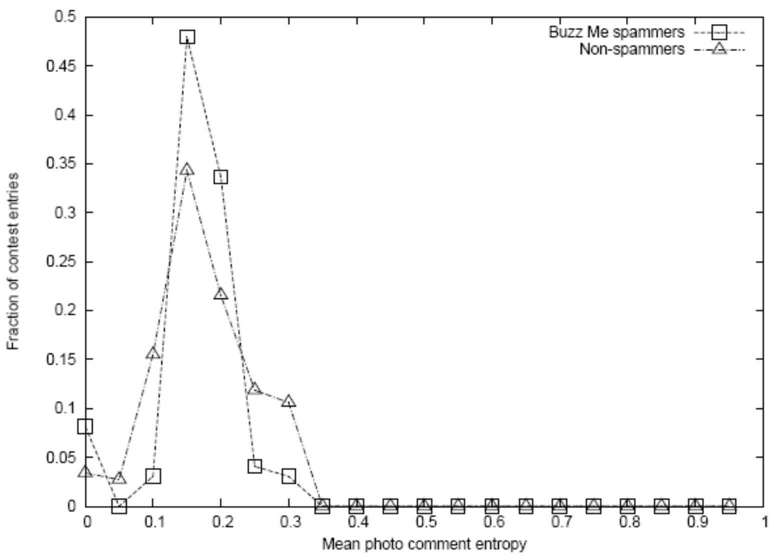


Fig. 4. Mean photo comment entropy distribution

Mean Photo Comment Entropy. We examine the mean photo comment entropy as further evidence that the accounts used to comment on, and Buzz, contest photos were not shill accounts. Our reasoning is that an attacker would be unlikely to post comments from automated profiles whose entropy is similar to posts made by ordinary users. The entropy of a comment c is defined as:

$$entropy(c) = \frac{1}{\lambda} \sum_{i=1}^{\lambda} p_i [log_{10}(\lambda) - log_{10}(p_i)] \tag{1}$$

where λ is the number of words in the comment, and p_i is the frequency with which word i appears in the comment. Figure 4 shows the distribution of mean photo comment entropy for Buzz Me spammers and for non-spammers. We again performed a K-S goodness-of-fit test, and the resulting p-value is 1, i.e. the null hypothesis that the two distributions are the same again cannot be rejected at any significance level.

As these examples show, individual users can attack a rating system directly by engaging in a form of social engineering. Given examples of the messages between users, it is easy to detect attempted manipulation of Buzz count for particular items. A straightforward modification of the attack would be to mask the requests, e.g. by sending them in another channel such as email. In the following section, we will discuss how the manipulation may detected without access to messages asking a user to rate an item.

5 Detecting Buzz Manipulation

5.1 Detecting Manipulated Contests

Before identifying users whom we suspect may be manipulating their contest entries, we wish to determine whether the contests themselves show significant signs of being manipulated. We expect that, absent manipulation, the Buzz counts of contest entries will be approximately power-law distributed: most entries will receive no Buzz, while a very small number of items will receive a much higher Buzz than most. Figure 5 shows the distribution of Buzz counts for all contest entries, as well as for Contests 1 and 2. As expected, overall the Buzz PDF was approximately Pareto distributed, with parameter $\alpha = 3.55053$. On the other hand, the distribution of Buzz counts for Contests 1 and 2 is significantly different. In order to test each contest, we used a K-S test against the Pareto distribution for each of the 24 contests in our crawl. Three of the contests had distributions that were radically different from the population distribution: Contests 1 and 2, and a third contest that did not have a significant portion of spam that we could identify, but whose Buzz counts were unusual. Thus it appears that a deviation from an expected distribution of Buzz counts at least highlights suspicious behavior within a contest.

5.2 Outlier Buzz Counts

The most straightforward way to determine whether an item's Buzz count has been manipulated is to test whether the count itself is very rare in the distribution of Buzz counts. Buzznet has two classes of photos, featured and non-featured. Featured photos are chosen by humans, and are displayed prominently on Buzznet's home page. Unsurprisingly, featured photos have a higher expected

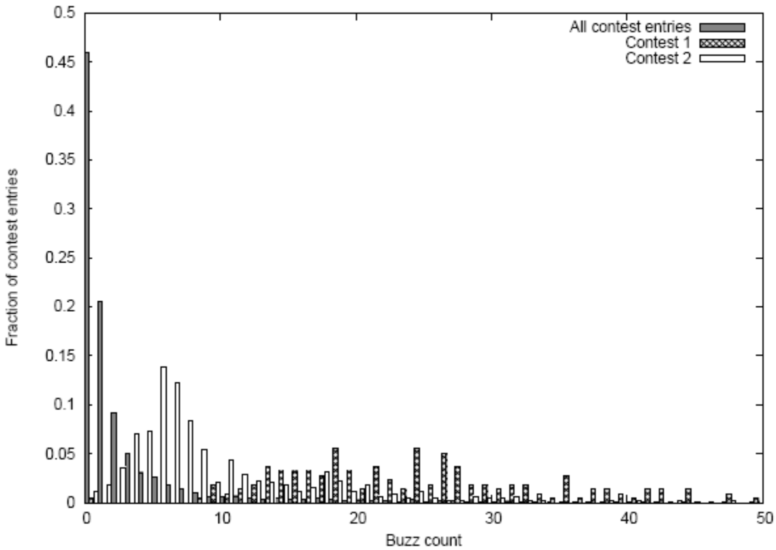


Fig. 5. Distribution of contest Buzz counts

Buzz count than non-featured ones. None of the contest entry photos was featured, so we compare the Buzz counts of contest entries with the non-featured photos.

To find a model for the expected distribution of Buzz counts, we fit a Pareto distribution to the Buzz counts of non-featured photos using maximum likelihood estimation. The resulting distribution had a shape parameter $\alpha = 2.05$. We then used a one-tailed test of the Buzz counts with the population model and a significance level of $\alpha = 0.0001$. There were 236 photos whose Buzz count exceeded the threshold. Roughly half of the most buzzed photos were posted by the most popular users, those for whom the $CCDF(in\ degree) \leq 0.0001$. The remaining photos were contest entries, which were either accepted or disqualified. The results for the top 20 most Buzzed entries in Contests 1 and 2 are shown in Table 2. In Contest 1, there was a previously unidentified Buzz Me spammer. This user sent 255 requests for Buzz, but her messages did not contain a photo URL, merely a description of how to find it. As we suggested in Section 4.1, our Buzz Me spam identification method may still contain false negatives, highlighting the importance of discovering manipulation of each item directly. The ambiguous entries had some requests sent by their posters, but fewer than 100.

To a certain extent, the photos caught with the one-tailed test shows that the detected users are victims of their own success: the Buzz counts they achieved are so much higher than expected, they clearly stand out. A more clever attacker might try to mask her activity more carefully by asking for Buzz from fewer

Table 2. Top 20 Buzzed photos in Contests 1 and 2

Contest	Buzz Me spammers	Disqualified	Ambiguous
1	12	5	3
2	13	4	3

users. The remaining tests show how manipulation may still be detected, even without an abnormally high Buzz count.

5.3 Unexpected Comment Dates

Unfortunately Buzznet does not allow us to see the Buzz count of an item over time, but as we described in Section 4.1, there is a very strong correlation between the number of comments a contest entry receives and its Buzz count. In general, we expect that items are most interesting to users shortly after they are posted, and that they will receive the most comments shortly after being posted. ([9] noted a similar phenomenon with the number of Diggs an item receives over time on Digg.) Figure 6 shows a histogram of the comments every photo received per day after being posted, stopping at 30 days from the posting date. As expected, the comments received per day follow a Pareto distribution. Hence a photo receiving the bulk of its comments after the posted date shows evidence of being manipulated: it takes time for the requestees to respond to requests for Buzz.

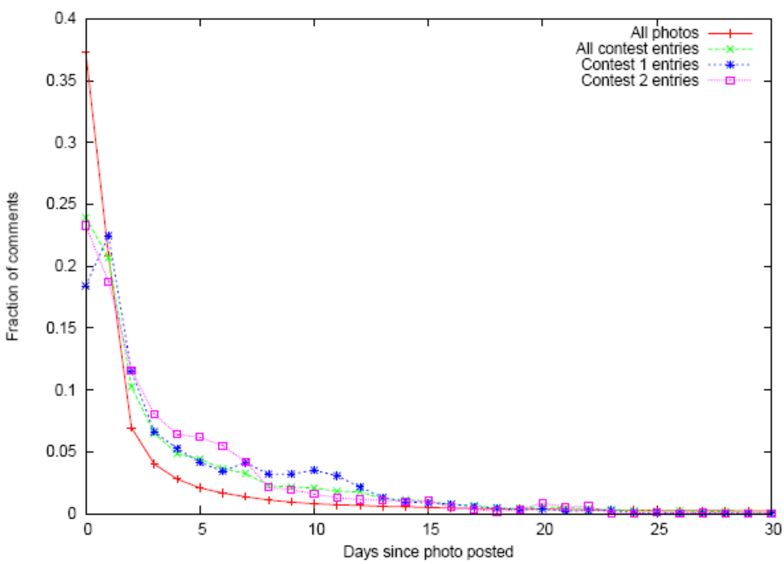


Fig. 6. Comments per day after photo post date

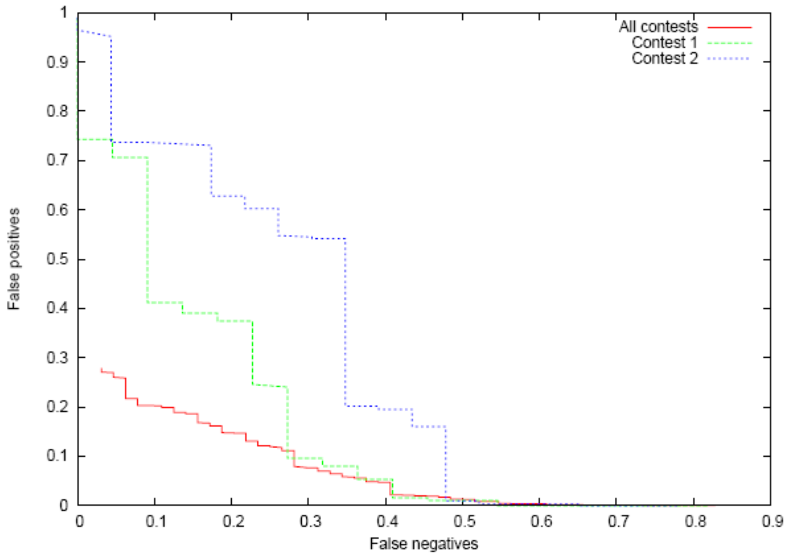


Fig. 7. ROC curve of photo comment date distribution

We used a K-S test on each photo's comment dates to see whether the comment dates followed the expected distribution. We cut off each entry's comments after 30 days to avoid influence from comments outside the contest period. We first used this test to check the results when the false positives were very low, fewer than 1%. By examining the resulting false positives by hand, we found several users who sent hundreds of requests for Buzz, but who were not found by our automated filter because the requests did not contain a URL. We added these users to our list of Buzz Me spammers, and computed ROC curves for the two contests, shown in Figure 7. As can be seen, the method allows us to find spammers with no false positives, but only when a significant fraction (approximately 2/3) of the spammers are not identified. We sought to improve our detection method by examining several other features, discussed next.

5.4 Buzz Count Mean and Variance

If a user's mean Buzz count is high, the user is exceptional: only 1% of users have a mean Buzz count of 3 or more. Without further information, the user can't be clearly classified: 1) Contest entries have generally higher Buzz counts than the overall population. 2) The entry's poster may be unusually popular. We can improve our classification by examining the users' Buzz count variance. Our intuition behind examining the users' Buzz count variance is that cheaters cannot manipulate their Buzz counts entirely at will: doing so takes some effort on their part. Thus, they tend to invest their energy in accumulating Buzz in few entries. This leads to two probable scenarios for cheaters: 1) They have posted a single entry, hence their variance is 0. Combined with a high mean, this is an indication of cheating: rarely will a user's only post become very popular. 2) They have posted more than one entry, but only few whose Buzz count they are manipulating, leading to a very large variance, much larger than a popular user's variance.

In order to test the combination of these features, we trained a C4.5 (J48) [14] decision tree classifier on the 4,239 unique contest entrants in our dataset. For brevity, we do not show the tree. Using this tree as a classifier, only two false positives were produced, but at the cost of a 41% false negative rate. In other words, combining features allowed us to retain a near-zero false positive rate, while reducing the false negative rate by a third compared to testing the photo comment date distribution alone. The tree also shows a number of features that match our intuition:

1. When the mean Buzz is low, the entrant is unlikely to have spammed other users asking for Buzz.
2. When the mean Buzz is relatively high (between 54 and 95), a high variance is indicative of the entrant having spammed, while a low variance is indicative of not having spammed: that is, the entrant is likely to be a popular user.
3. When the mean Buzz is very high (greater than 95), the situation is more complex. Surprisingly, users who are relatively unpopular (the CDF of their in degree is less than .8085) are non-spammers. There are only three such users in our dataset, so we expect them to be a quirk of the data we collected. For the remaining users, either a very high variance or a very low variance is indicative of having spammed, as we expected, and a moderate variance is indicative of not having spammed. The p-value of the K-S test applied to the comment date distribution also helps distinguish spammers from non-spammers.

6 Conclusion and Future Work

In this work, we have shown evidence of successful social manipulation of a recommender system. We have also shown that it is possible to detect most such manipulation indirectly, i.e. without catching users in the act of asking others to rate their items. By catching most manipulators, we can ensure that a contest is at least $\frac{1}{2}$-fair. For future work, we intend to look for evidence of similar manipulation in other vote-based recommender systems. We also intend to revisit our assumption that false positives are worse than false negatives by applying cost-sensitive classifiers to our data.

References

1. Burke, R., Mobasher, B., Williams, C., Bhaumik, R.: Classification features for attack detection in collaborative recommender systems. In: KDD 2006: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 542–547. ACM, New York (2006)

2. Chirita, P.A., Nejdl, W., Zamfir, C.: Preventing shilling attacks in online recommender systems. In: WIDM 2005: Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management, pp. 67–74. ACM, New York (2005)
3. DeFigueiredo, D., Barr, E., Wu, S.F.: Trust is in the eye of the beholder. In: CSE 2009: Proceedings of the 2009 International Conference on Computational Science and Engineering, pp. 100–108. IEEE Computer Society, Washington (2009)
4. Golbeck, J.: Generating predictive movie recommendations from trust in social networks. In: Stølen, K., Winsborough, W.H., Martinelli, F., Massacci, F. (eds.) iTrust 2006. LNCS, vol. 3986, pp. 93–104. Springer, Heidelberg (2006)
5. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. ACM Commun. 35(12), 61–70 (1992)
6. Hogg, R.V., Tanis, E.A.: Probability and Statistical Inference. Prentice-Hall, Englewood Cliffs (2009)
7. Hsu, C.F., Khabiri, E., Caverlee, J.: Ranking comments on the social web. In: CSE 2009: Proceedings of the 2009 International Conference on Computational Science and Engineering, pp. 90–97. IEEE Computer Society, Washington (2009)
8. Lam, S.K., Riedl, J.: Shilling recommender systems for fun and profit. In: WWW 2004: Proceedings of the 13th International Conference on World Wide Web, pp. 393–402. ACM, New York (2004)
9. Lerman, K., Galstyan, A.: Analysis of social voting patterns on digg. In: WOSP 2008: Proceedings of the First Workshop on Online Social Networks, pp. 7–12. ACM, New York (2008)
10. Massa, P., Avesani, P.: Trust-aware recommender systems. In: RecSys 2007: Proceedings of the 2007 ACM Conference on Recommender Systems, pp. 17–24. ACM, New York (2007)
11. Mehta, B., Hofmann, T., Fankhauser, P.: Lies and propaganda: detecting spam users in collaborative filtering. In: IUI 2007: Proceedings of the 12th International Conference on Intelligent User Interfaces, pp. 14–21. ACM, New York (2007)
12. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: IUI 2005: Proceedings of the 10th International Conference on Intelligent User Interfaces, pp. 167–174. ACM, New York (2005)
13. O'Mahony, M., Hurley, N., Kushmerick, N., Silvestre, G.: Collaborative recommendation: A robustness analysis. ACM Trans. Internet Technol. 4(4), 344–377 (2004)
14. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
15. Resnick, P., Sami, R.: The influence limiter: provably manipulation-resistant recommender systems. In: RecSys 2007: Proceedings of the 2007 ACM conference on Recommender systems, pp. 25–32. ACM, New York (2007)
16. Resnick, P., Varian, H.R.: Recommender systems. ACM Commun. 40(3), 56–58 (1997)

Bicriteria Models for Fair and Efficient Resource Allocation*

Włodzimierz Ogryczak

Warsaw University of Technology, Institute of Control & Computation Engineering
Nowowiejska 15/19, 00-665 Warsaw, Poland
wogrycza@ia.pw.edu.pl

Abstract. Resource allocation problems are concerned with the allocation of limited resources among competing agents so as to achieve the best system performances. In systems which serve many users, like in networking, there is a need to respect some fairness rules while looking for the overall efficiency. The so-called Max-Min Fairness is widely used to meet these goals. However, allocating the resource to optimize the worst performance may cause a dramatic worsening of the overall system efficiency. Therefore, several other fair allocation schemes are searched and analyzed. In this paper we show how the scalar inequality measures can be consistently used in bicriteria models to search for fair and efficient allocations while taking into account importance weighting of the agents.

1 Introduction

Resource allocation problems are concerned with the allocation of limited resources among competing activities [12]. In this paper, we focus on approaches that, while allocating resources to maximize the system efficiency, they also attempt to provide a fair treatment of all the competing agents (activities) [19,29]. The problems of efficient and fair resource allocation arise in various systems which serve many users, like in telecommunication systems among others. In networking a central issue is how to allocate bandwidth to flows efficiently and fairly [3,6,9,15,32,31,34]. In location analysis of public services, the decisions often concern the placement of a service center or another facility in a position so that the users are treated fairly in an equitable way, relative to certain criteria [25].

The generic resource allocation problem may be stated as follows. Each activity is measured by an individual performance function that depends on the corresponding resource level assigned to that activity. A larger function value is considered better, like the performance measured in terms of quality level, capacity, service amount available, etc. Hence, it may be viewed as a multiagent optimization problem. It covers various complex resource allocation problem like network dimensioning as well as general many to many multiagent assignment

* This work was partially supported by the Polish Ministry of Science and Higher Education under the grants N N516 4307 33 and 69/N-SINGAPUR/2007/0.

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 140–159, 2010.

problems with the special case of the Santa Claus problem [2] representing fair allocation of indivisible goods. Such problems occur in many contexts like paper assignment problem [11,18], social meeting on the web or transportation problems [24], among others.

Models with an (aggregated) objective function that maximizes the mean (or simply the sum) of individual performances are widely used to formulate resource allocation problems, thus defining the so-called mean solution concept. This solution concept is primarily concerned with the overall system efficiency. As based on averaging, it often provides solution where some smaller services are discriminated in terms of allocated resources. An alternative approach depends on the so-called Max-Min solution concept, where the worst performance is maximized. The Max-Min approach is consistent with Rawlsian [36] theory of justice, especially when additionally regularized with the lexicographic order. The latter is called the Max-Min Fairness (MMF) and commonly used in networking [34]. Allocating the resources to optimize the worst performances may cause, however, a large worsening of the overall (mean) performances. Moreover, the MMF approach does not allow us to reflect any importance weighting of agents. Therefore, there is a need to seek a compromise between the two extreme approaches discussed above.

Fairness is, essentially, an abstract socio-political concept that implies impartiality, justice and equity [35,41], Nevertheless, fairness was frequently quantified with the so-called inequality measures to be minimized [1,37,39]. Unfortunately, direct minimization of typical inequality measures contradicts the maximization of individual outcomes and it may lead to inferior decisions. The concept of fairness has been studied in various areas beginning from political economics problems of fair allocation of consumption bundles [8,33,35] to abstract mathematical formulation [40]. In order to ensure fairness in a system, all system entities have to be equally well provided with the system's services. This leads to concepts of fairness expressed by the equitable efficiency [16,31]. The concept of equitably efficient solution is a specific refinement of the Pareto-optimality taking into account the inequality minimization according to the Pigou-Dalton approach. In this paper the use of scalar inequality measures in bicriteria models to search for fair and efficient allocations is analyzed. There is shown that properties of convexity and positive homogeneity together with some boundedness condition are sufficient for a typical inequality measure to guarantee that it can be used consistently with the equitable optimization rules.

The paper is organized as follows. In the next section we introduce the fairness notion based on the equitable optimization with the preference structure that complies with both the efficiency (Pareto-optimality) principle and with the Pigou-Dalton principle of transfers. It is additionally extended in Section 3 to the problems with importance weighted agents. In Section 4 the basic inequality measures are discussed and the fair consistency concepts based on the underachievement criteria are introduced. Further, in Section 5, the equitable consistency of the underachievement criteria is analyzed and sufficient conditions for the inequality measures to keep this consistency property are introduced. We verify the properties for the basic inequality measures.

2 Equity and Fairness

The generic resource allocation problem may be stated as follows. There is a system dealing with a set I of m services. There is given a measure of services realization within a system. In applications we consider, the measure usually expresses the service quality. In general, outcomes can be measured (modeled) as service time, service costs, service delays as well as in a more subjective way. There is also given a set Q of allocation patterns (allocation decisions). For each service $i \in I$ a function $f_i(\mathbf{x})$ of the allocation pattern $\mathbf{x} \in Q$ has been defined. This function, called the individual objective function, measures the outcome (effect) $y_i = f_i(\mathbf{x})$ of allocation $\mathbf{x}$ pattern for service i. In typical formulations a larger value of the outcome means a better effect (higher service quality or client satisfaction). Otherwise, the outcomes can be replaced with their complements to some large number. Therefore, without loss of generality, we can assume that each individual outcome y_i is to be maximized which allows us to view the generic resource allocation problem as a vector maximization model:

$$\max \{\mathbf{f}(\mathbf{x}) \ : \ \mathbf{x} \in Q\} \tag{1}$$

where $\mathbf{f}(\mathbf{x})$ is a vector-function that maps the decision space $X = R^n$ into the criterion space $Y = R^m$, and $Q \subset X$ denotes the feasible set. We do not assume any special form of the feasible set Q while analyzing properties of the solution concepts. We rather allow the feasible set to be a general, possibly discrete (nonconvex), set. Hence, the problem cover various complex resource allocation problem like network dimensioning as well as general many to many multiagent assignment problems with the special case of the Santa Claus problem [2] representing fair allocation of indivisible goods. Although we allow the feasible set to contain more complex relations than the basic assignment constraints, like in problems of network resource allocation [34].

Model (1) only specifies that we are interested in maximization of all objective functions f_i for $i \in I = \{1, 2, \ldots, m\}$. In order to make it operational, one needs to assume some solution concept specifying what it means to maximize multiple objective functions. The solution concepts may be defined by properties of the corresponding preference model. The preference model is completely characterized by the relation of weak preference, denoted hereafter with $\succeq$. The corresponding relations of strict preference $\succ$ and indifference $\cong$ are defined then by the following formulas:

$$\mathbf{y}' \succ \mathbf{y}'' \Leftrightarrow (\mathbf{y}' \succeq \mathbf{y}'' \quad \text{and} \quad \mathbf{y}'' \not\succeq \mathbf{y}'),$$
$$\mathbf{y}' \cong \mathbf{y}'' \Leftrightarrow (\mathbf{y}' \succeq \mathbf{y}'' \quad \text{and} \quad \mathbf{y}'' \succeq \mathbf{y}').$$

The standard preference model related to the Pareto-optimal (efficient) solution concept assumes that the preference relation $\succeq$ is *reflexive*:

$$\mathbf{y} \succeq \mathbf{y}, \tag{2}$$

transitive:

$$(\mathbf{y}' \succeq \mathbf{y}'' \quad \text{and} \quad \mathbf{y}'' \succeq \mathbf{y}''') \quad \Rightarrow \quad \mathbf{y}' \succeq \mathbf{y}''', \tag{3}$$

and *strictly monotonic*:

$$\mathbf{y} + \varepsilon \mathbf{e}_i \succ \mathbf{y} \quad \text{for } \varepsilon > 0;\ i = 1, \dots, m, \tag{4}$$

where $\mathbf{e}_i$ denotes the i–th unit vector in the criterion space. The last assumption expresses that for each individual objective function more is better (maximization). The preference relations satisfying axioms (2)–(4) are called hereafter *rational preference relations.* The rational preference relations allow us to formalize the Pareto-optimality (efficiency) concept with the following definitions. We say that outcome vector $\mathbf{y}'$ rationally dominates $\mathbf{y}''$ ($\mathbf{y}' \succ_r \mathbf{y}''$), iff $\mathbf{y}' \succ \mathbf{y}''$ for all rational preference relations $\succeq$. We say that feasible solution $\mathbf{x} \in Q$ is a *Pareto-optimal (efficient)* solution of the multiple criteria problem (1), iff $\mathbf{y} = \mathbf{f}(\mathbf{x})$ is rationally nondominated.

Simple solution concepts for multiple criteria problems are defined by aggregation (or utility) functions $g : Y \to R$ to be maximized. Thus the multiple criteria problem (1) is replaced with the maximization problem

$$\max \{g(\mathbf{f}(\mathbf{x})) \ : \ \mathbf{x} \in Q\} \tag{5}$$

In order to guarantee the consistency of the aggregated problem (5) with the maximization of all individual objective functions in the original multiple criteria problem (or Pareto-optimality of the solution), the aggregation function must be strictly increasing with respect to every coordinate.

The simplest aggregation functions commonly used for the multiple criteria problem (1) are defined as the mean (average) outcome

$$\mu(\mathbf{y}) = \frac{1}{m} \sum_{i=1}^{m} y_i \tag{6}$$

or the worst outcome

$$M(\mathbf{y}) = \min_{i=1,\dots,m} y_i. \tag{7}$$

The mean (6) is a strictly increasing function while the minimum (7) is only nondecreasing. Therefore, the aggregation (5) using the sum of outcomes always generates a Pareto-optimal solution while the maximization of the worst outcome may need some additional refinement. The mean outcome maximization is primarily concerned with the overall system efficiency. As based on averaging, it often provides a solution where some services are discriminated in terms of performances. On the other hand, the worst outcome maximization, ie, the so-called Max-Min solution concept is regarded as maintaining equity. Indeed, in the case of a simplified resource allocation problem with the knapsack constraints, the Max-Min solution meets the perfect equity requirement. In the general case, with possibly more complex feasible set structure, this property is not fulfilled. Nevertheless, if the perfectly equilibrated outcome vector $\bar{y}_1 = \bar{y}_2 = \ldots = \bar{y}_m$

is nondominated, then it is the unique optimal solution of the corresponding Max-Min optimization problem [26]. In other words, the perfectly equilibrated outcome vector is a unique optimal solution of the Max-Min problem if one cannot find any (possibly not equilibrated) vector with improved at least one individual outcome without worsening any others. Unfortunately, it is not a common case and, in general, the optimal set to the Max-Min aggregation may contain numerous alternative solutions including dominated ones. The Max-Min solution may be then regularized according to the Rawlsian principle of justice [36] which leads us to the lexicographic Max-Min concepts or the so-called Max-Min Fairness [13,20,30,4]. Although they are possible alternative refinements of the Max-Min ordering [10].

In order to ensure fairness in a system, all system entities have to be equally well provided with the system's services. This leads to concepts of fairness expressed by the equitable rational preferences [16]. First of all, the fairness requires impartiality of evaluation, thus focusing on the distribution of outcome values while ignoring their ordering. That means, in the multiple criteria problem (1) we are interested in a set of outcome values without taking into account which outcome is taking a specific value. Hence, we assume that the preference model is impartial (anonymous, symmetric). In terms of the preference relation it may be written as the following axiom

$$(y_{\pi(1)}, y_{\pi(2)}, \ldots, y_{\pi(m)}) \cong (y_1, y_2, \ldots, y_m) \quad \text{for any permutation } \pi \text{ of } I \tag{8}$$

which means that any permuted outcome vector is indifferent in terms of the preference relation. Further, fairness requires equitability of outcomes which causes that the preference model should satisfy the (Pigou–Dalton) principle of transfers. The principle of transfers states that a transfer of any small amount from an outcome to any other relatively worse–off outcome results in a more preferred outcome vector. As a property of the preference relation, the principle of transfers takes the form of the following axiom

$$y_{i'} > y_{i''} \quad \Rightarrow \quad \mathbf{y} - \varepsilon \mathbf{e}_{i'} + \varepsilon \mathbf{e}_{i''} \succ \mathbf{y} \quad \text{for } 0 < \varepsilon \leq (y_{i'} - y_{i''})/2 \tag{9}$$

The rational preference relations satisfying additionally axioms (8) and (9) are called hereafter *fair (equitable) rational preference relations*. We say that outcome vector $\mathbf{y}'$ *fairly (equitably) dominates* $\mathbf{y}''$ ($\mathbf{y}' \succ_e \mathbf{y}''$), iff $\mathbf{y}' \succ \mathbf{y}''$ for all fair rational preference relations $\succeq$. In other words, $\mathbf{y}'$ fairly dominates $\mathbf{y}''$, if there exists a finite sequence of vectors $\mathbf{y}^j$ ($j = 1, 2, \ldots, s$) such that $\mathbf{y}^1 = \mathbf{y}''$, $\mathbf{y}^s = \mathbf{y}'$ and $\mathbf{y}^j$ is constructed from $\mathbf{y}^{j-1}$ by application of either permutation of coordinates, equitable transfer, or increase of a coordinate. An allocation pattern $\mathbf{x} \in Q$ is called *fairly (equitably) efficient* or simply *fair* if $\mathbf{y} = \mathbf{f}(\mathbf{x})$ is fairly nondominated. Note that each fairly efficient solution is also Pareto-optimal, but not vice verse.

In order to guarantee fairness of the solution concept (5), additional requirements on the class of aggregation (utility) functions must be introduced. In particular, the aggregation function must be additionally symmetric (impartial), i.e. for any permutation π of I,

$$g(y_{\pi(1)}, y_{\pi(2)}, \ldots, y_{\pi(m)}) = g(y_1, y_2, \ldots, y_m) \tag{10}$$

as well as be equitable (to satisfy the principle of transfers)

$$g(y_1, \ldots, y_{i'} - \varepsilon, \ldots, y_{i''} + \varepsilon, \ldots, y_m) > g(y_1, y_2, \ldots, y_m) \tag{11}$$

for any $0 < \varepsilon \leq (y_{i'} - y_{i''})/2$. In the case of a strictly increasing function satisfying both the requirements (10) and (11), we call the corresponding problem (5) a *fair (equitable) aggregation* of problem (1). Every optimal solution to the fair aggregation (5) of a multiple criteria problem (1) defines some fair (equitable) solution.

Note that both the simplest aggregation functions, the sum (6) and the minimum (7), are symmetric although they do not satisfy the equitability requirement (11). To guarantee the fairness of solutions, some enforcement of concave properties is required. For any strictly concave, increasing utility function $u : R \rightarrow R$, the function $g(\mathbf{y}) = \sum_{i=1}^{m} u(y_i)$ is a strictly monotonic and equitable thus defining a family of the fair aggregations. Various concave utility functions u can be used to define such fair solution concepts. In the case of the outcomes restricted to positive values, one may use logarithmic function thus resulting in the *Proportional Fairness* (PF) solution concept [14]. Actually, it corresponds to the so-called Nash criterion [23] which maximizes the product of additional utilities compared to the status quo. For a common case of upper bounded outcomes $y_i \leq y^*$ one may maximize power functions $-\sum_{i=1}^{m} (y^* - y_i)^p$ for $1 < p < \infty$ which corresponds to the minimization of the corresponding p-norm distances from the common upper bound y^* [17].

Fig. 1 presents the structure of fair dominance for two-dimensional outcome vectors. For any outcome vector $\bar{\mathbf{y}}$, the fair dominance relation distinguishes set $D(\bar{\mathbf{y}})$ of dominated outcomes (obviously worse for all fair rational preferences) and set $S(\bar{\mathbf{y}})$ of dominating outcomes (obviously better for all fair rational preferences). However, some outcome vectors are left (in white areas) and they can be differently classified by various specific fair rational preferences. The MMF fairness assigns the entire interior of the inner white triangle to the set of preferred outcomes while classifying the interior of the external open triangles as worse outcomes. Isolines of various utility functions $u(\mathbf{y}) = u(\bar{\mathbf{y}})$ may split the white areas in different ways. For instance, there is no fair dominance between vectors $(0.01, 1)$ and $(0.02, 0.02)$ and the MMF considers the latter as better while the proportional fairness points out the former. On the other hand, vector $(0.02, 0.99)$ fairly dominates $(0.01, 1)$ and all fairness models (including MMF and PF) prefers the former. One may notice that the set $D(\bar{\mathbf{y}})$ of directions leading to outcome vectors being dominated by a given $\bar{\mathbf{y}}$ is, in general, not a cone and it is not convex. Although, when we consider the set $S(\bar{\mathbf{y}})$ of directions leading to outcome vectors dominating given $\bar{\mathbf{y}}$ we get a convex set.

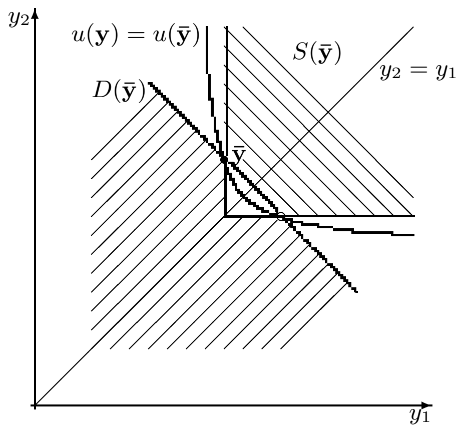


Fig. 1. Structure of the fair dominance: $D(\bar{\mathbf{y}})$ – the set fairly dominated by $\bar{\mathbf{y}}$, $S(\bar{\mathbf{y}})$ – the set of outcomes fairly dominating $\bar{\mathbf{y}}$

3 Fairness with Importance Weighted Agents

Frequently, one may be interested in putting into allocation models some additional agent weights $v_i > 0$. Typically the model of distribution weights is introduced to represent the agent importance thus defining distribution of outcomes $y_i = f_i(\mathbf{x})$ according to measures defined by the weights v_i for $i = 1, \dots, m$. Note that such distribution weights allow us for a clear interpretation of weights as the agent repetitions [7]. Splitting an agent into two agents does not cause any change of the final distribution of outcomes. For theoretical considerations one may assume that the problem is transformed (disaggregated) to the unweighted one (that means all the agent weights are equal to 1). Note that such a disaggregation is possible for integer as well as rational agent weights, but it usually dramatically increases the problem size. Therefore, we are interested in solution concepts which can be applied directly to the weighted problem.

As mentioned, for some theoretical considerations it might be convenient to disaggregate the weighted problems into the unweighted one. Therefore, to simplify the analysis we will assume integer weights v_i, although while discussing solution concepts we will use the normalized agent weights $\bar{v}_i = v_i / \sum_{i=1}^m v_i$ for $i = 1, \dots, m$, rather than the original quantities v_i. Note that, in the case of unweighted problem (all $v_i = 1$), all the normalized weights are given as $\bar{v}_i = 1/m$. Furthermore, to avoid possible misunderstandings between the weighted outcomes and the corresponding unweighted form of outcomes we will use the following notation. Index set I will always denote unweighted agents (with possible repetitions if originally weighted) and vector $\mathbf{y} = (y_i)_{i \in I} = (y_1, y_2, \dots, y_m)$ will denote the unweighted outcomes. While directly dealing with the weighted problem (without its disaggregation to the unweighted one) we will use I_v to denote the set of agents and the corresponding outcomes will be represented by vector $\mathbf{y} = (y_{v_i})_{i \in I_v}$. We illustrate this with the following small example.

Let us consider a weighted resource allocation problem with two agents A1 and A2 having assigned demand weights $v_1 = 1$ and $v_2 = 9$, respectively. Their outcomes relate to two potential allocation patterns P1 and P2 are given as follows:

	A1	A2
P1	10	0
P2	0	0

Hence, $I_v = \{1, 2\}$ and the potential resource allocations generate two outcome vectors $\mathbf{y}' = (10_1, 0_9)$ and $\mathbf{y}'' = (0_1, 0_9)$, respectively. The demand weights are understood as agents repetitions. Thus, the problem is understood as equivalent to the unweighted problem with 10 agents ($I = \{1, 2, \ldots, 10\}$) where the first one corresponds to A1 and the further nine unweighted agents correspond to single agent A2. In this disaggregated form, the outcome vectors generated by allocation patterns P1 and P2 are given as $\mathbf{y}' = (10, 0, 0, 0, 0, 0, 0, 0, 0, 0)$ and $\mathbf{y}'' = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$, respectively. Note that outcome vector $\mathbf{y}''$ with all the coordinates 0 is obviously worse than unequal vector $\mathbf{y}'$ with one distance reduced to 0. Actually, $\mathbf{y}'$ Pareto dominates $\mathbf{y}''$.

The classical solution concepts of mean and Max-Min are well defined for aggregated models using importance weights $v_i > 0$. Exactly, the Max-Min solution concept is defined by maximization of the minimum outcome

$$M(\mathbf{y}) = \max_{i \in I} y_i = \max_{i \in I_v} y_{v_i}, \tag{12}$$

thus not affected by the importance weights at all. The same applies to its lexicographic regularization expressed as the MMF concept.

The solution concept of the mean outcome (6) can easily accommodate the importance weights as

$$\mu(\mathbf{y}) = \frac{1}{m} \sum_{i \in I} y_i = \sum_{i \in I_v} \bar{v}_i y_{v_i}. \tag{13}$$

Similarly, for any utility function $u : R \to R$ we get

$$\mu(u(\mathbf{y})) = \frac{1}{m} \sum_{i \in I} u(y_i) = \sum_{i \in I_v} \bar{v}_i u(y_{v_i}). \tag{14}$$

The fair dominance for general weighted problems can be derived by their disaggregation to the unweighted ones. It can be mathematically formalized as follows. First, we introduce the right-continuous cumulative distribution function (cdf):

$$F_{\mathbf{y}}(d) = \sum_{i \in I_v} \bar{v}_i \delta_i(d), \quad \delta_i(d) = \begin{cases} 1 & \text{if } y_{v_i} \le d \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

which for any real (outcome) value d provides the measure of outcomes smaller or equal to d. Next, we introduce the quantile function $F_{\mathbf{y}}^{(-1)}$ as the left-continuous inverse of the cumulative distribution function $F_{\mathbf{y}}$:

$$F_{\mathbf{y}}^{(-1)}(\beta) = \inf \{\eta : F_{\mathbf{y}}(\eta) \geq \beta\} \quad \text{for } 0 < \beta \leq 1.$$

By integrating $F_{\mathbf{y}}^{(-1)}$ one gets $F_{\mathbf{y}}^{(-2)}(0) = 0$ and

$$F_{\mathbf{y}}^{(-2)}(\beta) = \int_0^\beta F_{\mathbf{y}}^{(-1)}(\alpha) d\alpha \quad \forall\ 0 < \beta \leq 1, \tag{16}$$

where $F_{\mathbf{y}}^{(-2)}(1) = \mu(\mathbf{y})$. The graph of function $F_{\mathbf{y}}^{(-2)}(\beta)$ (with respect to β) take the form of concave curves. It is called Absolute Lorenz Curve (ALC) [28], due to its relation to the classical Lorenz curve used in income economics as a cumulative population versus income curve to compare equity of income distributions. Indeed, the Lorenz curve may be viewed as function $LC(\xi) = \frac{1}{\mu(\mathbf{y})} \int_0^\xi F_{\mathbf{y}}^{(-1)}(\alpha) d\alpha$ thus equivalent to function $F_{\mathbf{y}}^{(-2)}(\beta)$ normalized by the distribution average. Therefore, the classical Lorenz model is focused on equity while ignoring the average result and any perfectly equal distribution of income has the diagonal line as the Lorenz curve (the same independently from the income value). Within the ALC model both equity and values of outcomes are represented. The ALC defines the relation (partial order) equivalent to the equitable dominance. Exactly, outcome vector $\mathbf{y}'$ equitably dominates $\mathbf{y}''$, if and only if $F_{\mathbf{y}'}^{(-2)}(\beta) \geq F_{\mathbf{y}''}^{(-2)}(\beta)$ for all $\beta \in (0, 1]$ where at least one strict inequality holds. Note that for the expanded form to the unweighted outcomes, the ALC is completely defined by the values of the (cumulated) ordered outcomes. Hence, $\bar{\theta}_i(\mathbf{y}) = mF_{\mathbf{y}}^{(-2)}(i/m)$ for $i = 1, \ldots, m$, and pointwise comparison of cumulated ordered outcomes is enough to justify equitable dominance.

Finally, the impartiality of the allocation process (8) is considered in terms that two allocation schemes leading to the same distribution (cdf) of outcomes are indifferent

$$F_{\mathbf{y}'} = F_{\mathbf{y}''} \quad \Rightarrow \quad \mathbf{y}' \cong \mathbf{y}''. \tag{17}$$

The principle of transfers (9) is considered for single units of demand. Although it can can be applied directly to the outcomes of importance weighted agents in the following form [27]:

$$y_{v_{i'}} > y_{v_{i''}} \quad \Rightarrow \quad \mathbf{y} - \varepsilon \bar{v}_{i''} \mathbf{e}_{v_{i'}} + \varepsilon \bar{v}_{i'} \mathbf{e}_{v_{i''}} \succ \mathbf{y} \quad \text{for } 0 < \varepsilon \leq \frac{y_{i'} - y_{i''}}{\bar{v}_{i'} + \bar{v}_{i''}}. \tag{18}$$

Alternatively, the fair dominance can be expressed on the cumulative distribution functions. Having introduced the right-continuous cumulative distribution function one may further integrate the cdf (15) to get the second order cumulative distribution function $F_{\mathbf{y}}^{(2)}(\eta) = \int_{-\infty}^\eta F_{\mathbf{y}}(\xi) d\xi$ for $\eta \in R$, representing average shortage to any real target η. By the theory of convex conjugate functions, the pointwise comparison of the second order cumulative distribution functions provides an alternative characterization of the equitable dominance relation [28].

Exactly, $\mathbf{y}'$ fairly dominates $\mathbf{y}''$, if and only if $F^{(2)}_{\mathbf{y}'}(\eta) \leq F^{(2)}_{\mathbf{y}''}(\eta)$ for all η where at least one strict inequality holds.

Furthermore, the classical results of majorization theory [21] allow us to refer the equitable dominance to the mean utility. For any convex, increasing utility function $u : R \rightarrow R$, if outcome vector $\mathbf{y}'$ fairly dominates $\mathbf{y}''$, then

$$\sum_{i=1}^{m} \frac{u(y'_i)}{m} = \sum_{i \in I_v} \bar{v}_i u(y'_{v_i}) \geq \sum_{i=1}^{m} \frac{u(y''_i)}{m} = \sum_{i \in I_v} \bar{v}_i u(y''_{v_i}).$$

Finally, there are three alternative analytical characterizations of the relation of fair dominance as specified in the following theorem. Note that according to condition (iii), the fair dominance is actually the so-called increasing convex order which is more commonly known as the second degree stochastic dominance (SSD) [28].

Theorem 1. *For any outcome vectors* $\mathbf{y}', \mathbf{y}'' \in A$ *each of the three following conditions is equivalent to the (weak) equitable dominance* $\mathbf{y}' \succeq_e \mathbf{y}''$*:*

(i) $F^{(-2)}_{\mathbf{y}'}(\beta) \geq F^{(-2)}_{\mathbf{y}''}(\beta)$ *for all* $\beta \in (0, 1]$*;*

(ii) $F^{(2)}_{\mathbf{y}'}(\eta) \leq F^{(2)}_{\mathbf{y}''}(\eta)$ *for all real* η*;*

(iii) $\sum_{i \in I_v} \bar{v}_i u(y'_i) \geq \sum_{i \in I_v} \bar{v}_i u(y''_i)$ *for any concave, increasing function* u*.*

Following Theorem 1, the importance weighted fair preference models are mathematically equivalent to the risk averse preference models for the decisions under risk, where the scenarios correspond to the agents and the importance weights define their probabilities while the agent outcomes represent realizations of a return under various scenarios.

4 Inequality Measures and Fair Consistency

Inequality measures were primarily studied in economics [39] while recently they become very popular tools in Operations Research. Typical inequality measures are some deviation type dispersion characteristics. They are *translation invariant* in the sense that $\varrho(\mathbf{y} + a\mathbf{e}) = \varrho(\mathbf{y})$ for any outcome vector $\mathbf{y}$ and real number a (where $\mathbf{e}$ vector of units $(1, \ldots, 1)$), thus being not affected by any shift of the outcome scale.

Moreover, the inequality measures are also *inequality relevant* which means that they are equal to 0 in the case of perfectly equal outcomes while taking positive values for unequal ones.

The simplest inequality measures are based on the absolute measurement of the spread of outcomes, like the *maximum (absolute) difference*

$$d(\mathbf{y}) = \max_{i,j \in I} |y_i - y_j| = \max_{i,j \in I_v} |y_{v_i} - y_{v_j}| \tag{19}$$

or the *mean absolute difference* also called the Gini's mean difference

$$\Gamma(\mathbf{y}) = \frac{1}{2m^2} \sum_{i \in I} \sum_{j \in I} |y_i - y_j| = \frac{1}{2} \sum_{i \in I_v} \sum_{j \in I_v} |y_{v_i} - y_{v_j}| \bar{v}_i \bar{v}_j. \tag{20}$$

In most application frameworks better intuitive appeal may have inequality measures related to deviations from the mean outcome like the *maximum (absolute) deviation*

$$R(\mathbf{y}) = \max_{i\in I} |y_i - \mu(\mathbf{y})| = \max_{i\in I_v} |y_{v_i} - \mu(\mathbf{y})| \tag{21}$$

or the *mean (absolute) deviation*

$$\delta(\mathbf{y}) = \frac{1}{m}\sum_{i\in I} |y_i - \mu(\mathbf{y})| = \sum_{i\in I_v} |y_{v_i} - \mu(\mathbf{y})|\bar{v}_i. \tag{22}$$

Note that the *standard deviation* σ (or the *variance* σ^2) represents both the deviations and the spread measurement as

$$\begin{aligned} \sigma(\mathbf{y}) &= \sqrt{\frac{1}{m}\sum_{i\in I}(y_i - \mu(\mathbf{y}))^2} = \sqrt{\frac{1}{2m^2}\sum_{i\in I}\sum_{j\in I}(y_i - y_j)^2} \\ &= \sqrt{\sum_{i\in I_v}(y_{v_i} - \mu(\mathbf{y}))^2\bar{v}_i} = \sqrt{\frac{1}{2}\sum_{i\in I_v}\sum_{j\in I_v}(y_{v_i} - y_{v_j})^2\bar{v}_i\bar{v}_j}. \end{aligned} \tag{23}$$

Deviational measures may be focused on the downside semideviations as related to worsening of outcome while ignoring downside semideviations related to improvement of outcome. One may define the *maximum (downside) semideviation*

$$\Delta(\mathbf{y}) = \max_{i\in I}(\mu(\mathbf{y}) - y_i) = \max_{i\in I_v}(\mu(\mathbf{y}) - y_{v_i}), \tag{24}$$

and the *mean (downside) semideviation*

$$\bar{\delta}(\mathbf{y}) = \frac{1}{m}\sum_{i\in I}(\mu(\mathbf{y}) - y_i)_+ = \sum_{i\in I_v}(\mu(\mathbf{y}) - y_{v_i})_+\bar{v}_i, \tag{25}$$

where $(.)_+$ denotes the nonnegative part of a number. Similarly, the *standard (downside) semideviation* is given as

$$\bar{\sigma}(\mathbf{y}) = \sqrt{\frac{1}{m}\sum_{i\in I}(\mu(\mathbf{y}) - y_i)_+^2} = \sqrt{\sum_{i\in I_v}(\mu(\mathbf{y}) - y_{v_i})_+^2\bar{v}_i}. \tag{26}$$

Due to the mean definition, the mean absolute semideviation is always equal to half of the mean absolute deviation ($\bar{\delta}(\mathbf{y}) = \frac{1}{2}\delta(\mathbf{y})$) but similar symmetry property in general does not apply to the maximum semideviation or the standard semideviation.

One can easily notice that direct minimization of typical inequality measures may contradict the optimization of individual outcomes resulting in equal but very low outcomes. As some resolution one may consider a bicriteria mean-equity model:

$$\max\ \{(\mu(\mathbf{f}(\mathbf{x})), -\varrho(\mathbf{f}(\mathbf{x})))\ :\ \mathbf{x} \in Q\} \tag{27}$$

which takes into account both the efficiency with optimization of the mean outcome $\mu(\mathbf{y})$ and the equity with minimization of an inequality measure $\varrho(\mathbf{y})$.

For typical inequality measures bicriteria model (27) is computationally very attractive since both the criteria are concave and LP implementable for many measures. Unfortunately, for any dispersion type inequality measures the bicriteria mean-equity model is not consistent with the outcomes maximization, and therefore is not consistent with the fair dominance. When considering a simple discrete problem with two allocation patterns P1 and P2 generating outcome vectors $\mathbf{y}' = (0, 0)$ and $\mathbf{y}'' = (2, 8)$, respectively, for any dispersion type inequality measure one gets $\varrho(\mathbf{y}'') > 0 = \varrho(\mathbf{y}')$ while $\mu(\mathbf{y}'') = 5 > 0 = \mu(\mathbf{y}')$. Hence, $\mathbf{y}''$ is not bicriteria dominated by $\mathbf{y}'$ and vice versa. Thus for any dispersion type inequality measure ϱ, allocation P1 with obviously worse outcome vector than that for allocation P2 is a Pareto-optimal solution in the corresponding bicriteria mean-equity model (27).

Note that the lack of consistency of the mean-equity model (27) with the outcomes maximization applies also to the case of the maximum semideviation $\Delta(\mathbf{y})$ (24) used as an inequality measure whereas subtracting this measure from the mean $\mu(\mathbf{y}) - \Delta(\mathbf{y}) = M(\mathbf{y})$ results in the worst outcome and thereby the first criterion of the MMF model. In other words, although a direct use of the maximum semideviation in the mean-equity model may contradict the outcome maximization, the measure can be used complementary to the mean leading us to the worst outcome criterion which does not contradict the outcome maximization. This construction can be generalized for various (dispersion type) inequality measures. For any inequality measure ϱ we introduce the corresponding underachievement function defined as the difference between the mean outcome and the inequality measure itself, i.e.

$$M_\varrho(\mathbf{y}) = \mu(\mathbf{y}) - \varrho(\mathbf{y}). \tag{28}$$

In the case of maximum semideviation the corresponding underachievement $M_\Delta(\mathbf{y})$ function represents the worst outcome $M(\mathbf{y})$. Similarly, in the case of mean semideviation one gets the underachievement function

$$M_{\bar{\delta}}(\mathbf{y}) = \mu(\mathbf{y}) - \bar{\delta}(\mathbf{y}) = \frac{1}{m}\sum_{i\in I} \min\{y_i, \mu(\mathbf{y})\} = \sum_{i\in I_v} \bar{v}_i \min\{y_{v_i}, \mu(\mathbf{y})\}$$

representing the mean underachievement. Further, due to $|y_i - y_j| = y_i + y_j - 2\min\{y_i, y_j\}$, one gets an alternative formula for the mean absolute difference

$$\Gamma(\mathbf{y}) = \mu(\mathbf{y}) - \frac{1}{m^2}\sum_{i\in I}\sum_{j\in I} \min\{y_i, y_j\} = \mu(\mathbf{y}) - \sum_{i\in I_v}\sum_{j\in I_v} \bar{v}_i\bar{v}_j \min\{y_{v_i}, y_{v_j}\} \tag{29}$$

and the corresponding underachievement function

$$M_\Gamma(\mathbf{y}) = \mu(\mathbf{y}) - \Gamma(\mathbf{y}) = \frac{1}{m^2}\sum_{i\in I}\sum_{j\in I} \min\{y_i, y_j\} = \sum_{i\in I_v}\sum_{j\in I_v} \bar{v}_i\bar{v}_j \min\{y_{v_i}, y_{v_j}\}$$

representing the mean pairwise worse outcome.

Note that one could consider a scaled $\varrho_\alpha(\mathbf{y}) = \alpha\varrho(\mathbf{y})$ as a different inequality measure. Therefore, in order to avoid creation of such redundant new inequality measures we allow the measures to be scaled with any positive factor $\alpha > 0$. For any inequality measure ϱ we introduce the corresponding underachievement function defined as the difference of the mean outcome and the (scaled) inequality measure itself, i.e.

$$M_{\alpha\varrho}(\mathbf{y}) = \mu(\mathbf{y}) - \alpha\varrho(\mathbf{y}). \tag{30}$$

This allows us to replace the original mean-equity bicriteria optimization (27) with the following bicriteria problem:

$$\max\{(\mu(\mathbf{f}(\mathbf{x})), \mu(\mathbf{f}(\mathbf{x})) - \alpha\varrho(\mathbf{f}(\mathbf{x}))) \;:\; \mathbf{x} \in Q\} \tag{31}$$

where the second objective represents the corresponding underachievement measure $M_{\alpha\varrho}(\mathbf{y})$ (30). Note that for any inequality measure $\varrho(\mathbf{y}) \geq 0$ one gets $M_{\alpha\varrho}(\mathbf{y}) \leq \mu(\mathbf{y})$ thus really expressing underachievements (comparing to mean) from the perspective of outcomes being maximized.

We will say that an inequality measure ϱ is *fairly α-consistent* if

$$\mathbf{y}' \succeq_e \mathbf{y}'' \quad \Rightarrow \quad \mu(\mathbf{y}') - \alpha\varrho(\mathbf{y}') \geq \mu(\mathbf{y}'') - \alpha\varrho(\mathbf{y}'') \tag{32}$$

The relation of fair α-consistency will be called *strong* if, in addition to (32), the following holds

$$\mathbf{y}' \succ_e \mathbf{y}'' \quad \Rightarrow \quad \mu(\mathbf{y}') - \alpha\varrho(\mathbf{y}') > \mu(\mathbf{y}'') - \alpha\varrho(\mathbf{y}''). \tag{33}$$

Theorem 2. *If the inequality measure $\varrho(\mathbf{y})$ is fairly α-consistent (32), then except for outcomes with identical values of $\mu(\mathbf{y})$ and $\varrho(\mathbf{y})$, every efficient solution of the bicriteria problem (31) is a fairly efficient allocation pattern. In the case of strong consistency (33), every allocation pattern $\mathbf{x} \in Q$ efficient to (31) is, unconditionally, fairly efficient.*

Proof. Let $\mathbf{x}^0 \in Q$ be an efficient solution of (31). Suppose that $\mathbf{x}^0$ is not fairly efficient. This means, there exists $\mathbf{x} \in Q$ such that $\mathbf{y} = \mathbf{f}(\mathbf{x}) \succ_e \mathbf{y}^0 = \mathbf{f}(\mathbf{x}^0)$. Then, it follows $\mu(\mathbf{y}) \geq \mu(\mathbf{y}^0)$, and simultaneously $\mu(\mathbf{y}) - \alpha\varrho(\mathbf{y}) \geq \mu(\mathbf{y}^0) - \alpha\varrho(\mathbf{y}^0)$, by virtue of the fair α-consistency (32). Since $\mathbf{x}^0$ is efficient to (31) no inequality can be strict, which implies $\mu(\mathbf{y}) = \mu(\mathbf{y}^0)$ and and $\varrho(\mathbf{y}) = \varrho(\mathbf{y}^0)$.

In the case of the strong fair α-consistency (33), the supposition $\mathbf{y} = \mathbf{f}(\mathbf{x}) \succ_e \mathbf{y}^0 = \mathbf{f}(\mathbf{x}^0)$ implies $\mu(\mathbf{y}) \geq \mu(\mathbf{y}^0)$ and $\mu(\mathbf{y}) - \alpha\varrho(\mathbf{y}) > \mu(\mathbf{y}^0) - \alpha\varrho(\mathbf{y}^0)$ which contradicts the efficiency of $\mathbf{x}^0$ with respect to (31). Hence, the allocation pattern $\mathbf{x}^0$ is fairly efficient.

5 Fair Consistency Conditions

Typical dispersion type inequality measures are directly defined for the weighted distributions of outcomes without any need of disaggregation. Actually, they depend only distribution of outcomes. Hence, they are impartial in the sense that

$$F_{\mathbf{y}'} = F_{\mathbf{y}''} \quad \Rightarrow \quad \varrho(\mathbf{y}') \cong \varrho(\mathbf{y}''). \tag{34}$$

as well as *clustering invariant* in the sense that any split of equal outcomes does not affect the measure. Moreover, typical inequality measures are convex, i.e. $\varrho(\lambda\mathbf{y}' + (1-\lambda)\mathbf{y}'') \leq \lambda\varrho(\mathbf{y}') + (1-\lambda)\varrho(\mathbf{y}'')$ for any $\mathbf{y}', \mathbf{y}''$ and $0 \leq \lambda \leq 1$. Certainly, the underachievement function $M_{\alpha\varrho}(\mathbf{y})$ must be also monotonic for the fair consistency which enforces more restrictions on the inequality measures. We will show further that convexity together with positive homogeneity and some boundedness of an inequality measure is sufficient to guarantee monotonicity of the corresponding underachievement measure and thereby to guarantee the fair α-consistency of inequality measure itself.

We say that (dispersion type) inequality measure $\varrho(\mathbf{y}) \geq 0$ is *Δ-bounded* if it is upper bounded by the maximum downside deviation, i.e.,

$$\varrho(\mathbf{y}) \leq \Delta(\mathbf{y}) \quad \forall\mathbf{y}. \tag{35}$$

Moreover, we say that $\varrho(\mathbf{y}) \geq 0$ is strictly Δ-bounded if inequality (35) is a strict bound, except from the case of perfectly equal outcomes, i.e., $\varrho(\mathbf{y}) < \Delta(\mathbf{y})$ for any $\mathbf{y}$ such that $\Delta(\mathbf{y}) > 0$.

Theorem 3. *Let $\varrho(\mathbf{y}) \geq 0$ be a convex, positively homogeneous, clustering invariant and translation invariant (dispersion type) inequality measure. If $\alpha\varrho(\mathbf{y})$ is Δ-bounded, then $\varrho(\mathbf{y})$ is fairly α-consistent in the sense of (32).*

Proof. The relation of fair dominance $\mathbf{y}' \succeq_e \mathbf{y}''$ denotes that there exists a finite sequence of vectors $\mathbf{y}^0 = \mathbf{y}'', \mathbf{y}^1, \ldots, \mathbf{y}^t$ such that $\mathbf{y}^k = \mathbf{y}^{k-1} - \varepsilon_k\mathbf{e}_{i'} + \varepsilon_k\mathbf{e}_{i''}$, $0 \leq \varepsilon_k \leq y_{i'}^{k-1} - y_{i''}^{k-1}$ for $k = 1, 2, \ldots, t$ and there exists a permutation π such that $y'_{\pi(i)} \geq y_i^t$ for all $i \in I$. Note that the underachievement function $M_{\alpha\varrho}(\mathbf{y})$, similar as $\varrho(\mathbf{y})$ depends only on the distribution of outcomes. Further, if $\mathbf{y}' \geq \mathbf{y}''$, then $\mathbf{y}' = \mathbf{y}'' + (\mathbf{y}' - \mathbf{y}'')$ and $\mathbf{y}' - \mathbf{y}'' \geq 0$. Hence, due to concavity and positive homogeneity, $M_{\alpha\varrho}(\mathbf{y}') \geq M_{\alpha\varrho}(\mathbf{y}'') + M_{\alpha\varrho}(\mathbf{y}' - \mathbf{y}'')$. Moreover, due to the bound (35), $M_{\alpha\varrho}(\mathbf{y}' - \mathbf{y}'') \geq \mu(\mathbf{y}' - \mathbf{y}'') - \Delta(\mathbf{y}' - \mathbf{y}'') \geq \mu(\mathbf{y}' - \mathbf{y}'') - \mu(\mathbf{y}' - \mathbf{y}'') = 0$. Thus, $M_{\alpha\varrho}(\mathbf{y})$ satisfies also the requirement of monotonicity. Hence, $M_{\alpha\varrho}(\mathbf{y}') \geq M_{\alpha\varrho}(\mathbf{y}^t)$. Further, let us notice that $\mathbf{y}^k = \lambda\bar{\mathbf{y}}^{k-1} + (1-\lambda)\mathbf{y}^{k-1}$ where $\bar{\mathbf{y}}^{k-1} = \mathbf{y}^{k-1} - (y_{i'} - y_{i''})\mathbf{e}_{i'} + (y_{i'} - y_{i''})\mathbf{e}_{i''}$ and $\lambda = \varepsilon/(y_{i'} - y_{i''})$. Vector $\bar{\mathbf{y}}^{k-1}$ has the same distribution of coefficients as $\mathbf{y}^{k-1}$ (actually it represents results of swapping $y_{i'}$ and $y_{i''}$). Hence, due to concavity of $M_{\alpha\varrho}(\mathbf{y})$, one gets $M_{\alpha\varrho}(\mathbf{y}^k) \geq \lambda M_{\alpha\varrho}(\bar{\mathbf{y}}^{k-1}) + (1-\lambda)M_{\alpha\varrho}(\mathbf{y}^{k-1}) = M_{\alpha\varrho}(\mathbf{y}^{k-1})$. Thus, $M_{\alpha\varrho}(\mathbf{y}') \geq M_{\alpha\varrho}(\mathbf{y}'')$ which justifies the fair α-consistency of $\varrho(\mathbf{y})$.

For strong fair α-consistency some strict monotonicity and concavity properties of the underachievement function are needed. Obviously, there does not exist any inequality measure which is positively homogeneous and simultaneously strictly convex. However, one may notice from the proof of Theorem 3 that only convexity properties on equally distributed outcome vectors are important for monotonous underachievement functions.

We say that inequality measure $\varrho(\mathbf{y}) \geq 0$ is *strictly convex on equally distributed outcome vectors*, if

$$\varrho(\lambda\mathbf{y}' + (1-\lambda)\mathbf{y}'') < \lambda\varrho(\mathbf{y}') + (1-\lambda)\varrho(\mathbf{y}'')$$

for $0 < \lambda < 1$ and any two vectors $\mathbf{y}' \neq \mathbf{y}''$ representing the same outcomes distribution as some $\mathbf{y}$, i.e., $\mathbf{y}' = (y_{\pi'(1)}, \ldots, y_{\pi'(m)})$ π' and $\mathbf{y}'' = (y_{\pi''(1)}, \ldots, y_{\pi''(m)})$ for some permutations π' and π'', respectively.

Theorem 4. *Let $\varrho(\mathbf{y}) \geq 0$ be a convex, positively homogeneous, clustering invariant and translation invariant (dispersion type) inequality measure. If $\varrho(\mathbf{y})$ is also strictly convex on equally distributed outcomes and $\alpha\varrho(\mathbf{y})$ is strictly Δ-bounded, then the measure $\varrho(\mathbf{y})$ is fairly strongly α-consistent in the sense of (33).*

Proof. Due to the clustering invariance we may consider only the unweighted case. The relation of weak fair dominance $\mathbf{y}' \succeq_e \mathbf{y}''$ denotes that there exists a finite sequence of vectors $\mathbf{y}^0 = \mathbf{y}'', \mathbf{y}^1, \ldots, \mathbf{y}^t$ such that $\mathbf{y}^k = \mathbf{y}^{k-1} - \varepsilon_k \mathbf{e}_{i'} + \varepsilon_k \mathbf{e}_{i''}$, $0 \leq \varepsilon_k \leq y_{i'}^{k-1} - y_{i''}^{k-1}$ for $k = 1, 2, \ldots, t$ and there exists a permutation π such that $y'_{\pi(i)} \geq y_i^t$ for all $i \in I$. The strict fair dominance $\mathbf{y}' \succ_e \mathbf{y}''$ means that $y'_{\pi(i)} > y_i^t$ for some $i \in I$ or at least one ε_k is strictly positive. Note that the underachievement function $M_{\alpha\varrho}(\mathbf{y})$ is strictly monotonous and strictly convex on equally distributed outcome vectors. Hence, $M_{\alpha\varrho}(\mathbf{y}') > M_{\alpha\varrho}(\mathbf{y}'')$ which justifies the fair strong α-consistency of the measure $\varrho(\mathbf{y})$.

The specific case of fair 1-consistency is also called *the mean-complementary fair consistency.* Note that the fair $\bar{\alpha}$-consistency of measure $\varrho(\mathbf{y})$ actually guarantees the mean-complementary fair consistency of measure $\alpha\varrho(\mathbf{y})$ for all $0 < \alpha \leq \bar{\alpha}$, and the same remain valid for the strong consistency properties. It follows from a possible expression of $\mu(\mathbf{y}) - \alpha\varrho(\mathbf{y})$ as the convex combination of $\mu(\mathbf{y}) - \bar{\alpha}\varrho(\mathbf{y})$ and $\mu(\mathbf{y})$. Hence, for any $\mathbf{y}' \succeq_e \mathbf{y}''$, due to $\mu(\mathbf{y}') \geq \mu(\mathbf{y}'')$ one gets $\mu(\mathbf{y}') - \alpha\varrho(\mathbf{y}') \geq \mu(\mathbf{y}'') - \alpha\varrho(\mathbf{y}'')$ in the case of the fair $\bar{\alpha}$-consistency of measure $\varrho(\mathbf{y})$ (or respective strict inequality in the case of strong consistency). Therefore, while analyzing specific inequality measures we seek the largest values α guaranteeing the corresponding fair consistency.

As mentioned, typical inequality measures are convex and many of them are positively homogeneous. Moreover, the measures such as the mean absolute (downside) semideviation $\bar{\delta}(\mathbf{y})$ (25), the standard downside semideviation $\bar{\sigma}(\mathbf{y})$ (26), and the mean absolute difference $\Gamma(\mathbf{y})$ (20) are Δ-bounded. Indeed, one may easily notice that $\mu(\mathbf{y}) - y_i \leq \Delta(\mathbf{y})$ and therefore

$$\begin{aligned}
\bar{\delta}(\mathbf{y}) &\leq \sum_{i \in I_v} \Delta(\mathbf{y})\bar{v}_i = \Delta(\mathbf{y}) \\
\bar{\sigma}(\mathbf{y}) &\leq \sqrt{\sum_{i \in I_v} \Delta(\mathbf{y})^2 \bar{v}_i} = \Delta(\mathbf{y}) \\
\Gamma(\mathbf{y}) &= \sum_{i \in I_v} \sum_{j \in I_v} (\mu(\mathbf{y}) - \min\{y_i, y_j\})\bar{v}_i \bar{v}_j \leq \sum_{i \in I_v} \sum_{j \in I_v} \Delta(\mathbf{y})\bar{v}_i \bar{v}_j = \Delta(\mathbf{y})
\end{aligned}$$

where the last formula is due to (29). Actually, all those inequality measures are strictly Δ-bounded since for any unequal outcome vector at least one outcome must be below the mean thus leading to strict inequalities in the above bounds.

Obviously, Δ-bounded (but not strictly) is also the maximum absolute downside deviation $\Delta(\mathbf{y})$ itself. This allows us to justify the maximum downside deviation $\Delta(\mathbf{y})$ (24), the mean absolute (downside) semideviation $\bar{\delta}(\mathbf{y})$ (25), the standard downside semideviation $\bar{\sigma}(\mathbf{y})$ (26) and the mean absolute difference $\Gamma(\mathbf{y})$ (20) as fairly 1-consistent (mean-complementary fairly consistent) in the sense of (32). Recall that the mean absolute semideviation is twice the mean absolute (downside) semideviation which means that $\alpha\delta(\mathbf{y})$ is Δ-bounded for any $0 < \alpha \leq 0.5$.

We emphasize that, despite the standard semideviation is a fairly 1-consistent inequality measure, the consistency is not valid for variance, semivariance and even for the standard deviation. These measures, in general, do not satisfy the all assumptions of Theorem 3. Certainly, we have enumerated only the simplest inequality measures studied in the resource allocation context which satisfy the assumptions of Theorem 3 and thereby they are fairly 1-consistent. Theorem 3 allows one to show this property for many other measures. In particular, one may easily find out that any convex combination of fairly α-consistent inequality measures remains also fairly α-consistent. On the other hand, among typical inequality measures the mean absolute difference seems to be the only one meeting the stronger assumptions of Theorem 4 and thereby maintaining the strong consistency.

Table 1. Fair consistency results for the basic inequality measures

Measure			α–consistency	
Mean absolute semideviation	$\bar{\delta}(\mathbf{y})$	(25)	1	
Mean absolute deviation	$\delta(\mathbf{y})$	(22)	0.5	
Maximum semideviation	$\Delta(\mathbf{y})$	(24)	1	
Mean absolute difference	$\Gamma(\mathbf{y})$	(20)	1	strong
Standard semideviation	$\bar{\sigma}(\mathbf{y})$	(26)	1	

The fair consistency results for basic dispersion type inequality measures considered in resource allocation problems are summarized in Table 1 where α values for unweighted as well as weighted problems are given and the strong consistency is indicated. Table 1 points out how the inequality measures can be used in resource allocation models to guarantee their harmony both with outcome maximization (Pareto-optimality) and with inequalities minimization (Pigou-Dalton equity theory). Exactly, for each inequality measure applied with the corresponding value α from Table 1 (or smaller positive value), every efficient solution of the bicriteria problem (31), ie. $\max\{(\mu(\mathbf{f}(\mathbf{x})), \mu(\mathbf{f}(\mathbf{x})) - \alpha\varrho(\mathbf{f}(\mathbf{x}))) : \ \mathbf{x} \in Q\}$, is a fairly efficient allocation pattern, except for outcomes with identical values of $\mu(\mathbf{y})$ and $\varrho(\mathbf{y})$. In the case of strong consistency (as for mean absolute difference), every solution $\mathbf{x} \in Q$ efficient to (31) is, unconditionally, fairly efficient.

As mentioned, the mean absolute semideviation is twice the mean absolute semideviation which means that $\alpha\delta(\mathbf{y})$ is Δ-bounded for any $0 < \alpha \leq 0.5$. The symmetry of mean absolute semideviations

Table 2. Marginal fair consistency results for symmetric inequality measures

Measure		α–consistency		
Maximum absolute deviation	$R(\mathbf{y})$ (21)	$\dfrac{\min_{i\in I_v} \bar{v}_i}{(1-\min_{i\in I_v} \bar{v}_i)}$	$\dfrac{1}{m-1}$	
Maximum absolute difference	$d(\mathbf{y})$ (19)	$\min_{i\in I_v} \bar{v}_i$	$\dfrac{1}{m}$	
Standard deviation	$\sigma(\mathbf{y})$ (23)	$\dfrac{\min_{i\in I_v} \bar{v}_i}{1-\min_{i\in I_v} \bar{v}_i}$	$\dfrac{1}{\sqrt{m-1}}$	strong

$$\bar{\delta}(\mathbf{y}) = \sum_{i\in I_v} (y_{v_i} - \mu(\mathbf{y}))_+ \bar{v}_i = \sum_{i\in I_v} (\mu(\mathbf{y}) - y_{v_i})_+ \bar{v}_i$$

can be also used to derive some marginal Δ-boundedness relations for other inequality measures. In particular, one may find out that any downside semideviation from the mean cannot be larger than $\kappa = (1-\min_{i\in I_v} \bar{v}_i)/\min_{i\in I_v} \bar{v}_i$ downside semideviations, where $\kappa = m-1$ for the case of m unweighted agents. Hence, the maximum absolute deviation satisfies the inequality $\min_{i\in I_v} \bar{v}_i/(1-\min_{i\in I_v} \bar{v}_i) R(\mathbf{y}) \le \Delta(\mathbf{y})$, while the maximum absolute difference fulfills $\min_{i\in I_v} \bar{v}_i d(\mathbf{y}) \le \Delta(\mathbf{y})$. In the case of unweighted agents these bounds take the forms $\frac{1}{m-1} R(\mathbf{y}) \le \Delta(\mathbf{y})$ and $\frac{1}{m} d(\mathbf{y}) \le \Delta(\mathbf{y})$, respectively. Similarly, for the standard deviation one gets

$$\sqrt{\frac{\min_{i\in I_v} \bar{v}_i}{1-\min_{i\in I_v} \bar{v}_i}}\, \sigma(\mathbf{y}) \le \Delta(\mathbf{y}) \quad \text{or} \quad \frac{1}{\sqrt{m-1}} \sigma(\mathbf{y}) \le \Delta(\mathbf{y})$$

for weighted or unweighted problems, respectively. Hence, $\alpha\sigma(\mathbf{y})$ is strictly Δ-bounded for any $\sqrt{\frac{\min_{i\in I_v} \bar{v}_i}{1-\min_{i\in I_v} \bar{v}_i}}$, since for any unequal outcome vector at least one outcome must be below the mean thus leading to strict inequalities in the above bounds. These allow us to justify the maximum absolute deviation with $0 < \alpha \le \min_{i\in I_v} \bar{v}_i/(1-\min_{i\in I_v} \bar{v}_i)$, the maximum absolute difference with $0 < \alpha \le \min_{i\in I_v} \bar{v}_i$ and the standard deviation with $0 < \alpha \le \frac{\min_{i\in I_v} \bar{v}_i}{1-\min_{i\in I_v} \bar{v}_i}$ as fairly α-consistent within the specified intervals of α. Moreover, the α-consistency of the standard deviation is strong. These marginal consistency results are summarized in Table 2 for weighted and unweighted agents, respectively.

6 Conclusions

The problems of efficient and fair resource allocation arise in various systems which serve many users. Fairness is, essentially, an abstract socio-political concept that implies impartiality, justice and equity. Nevertheless, in operations research it was quantified with various solution concepts. The equitable optimization with the preference structure that complies with both the efficiency (Pareto-optimality) and with the Pigou-Dalton principle of transfers may be used to formalize the fair solution concepts. Multiple criteria models equivalent to equitable optimization allows to generate a variety of fair and efficient resource allocation patterns [31].

In this paper we have analyzed how scalar inequality measures can be used to guarantee the fair consistency. It turns out that several inequality measures can be combined with the mean itself into the optimization criteria generalizing the concept of the worst outcome and generating fairly consistent underachievement measures. We have shown that properties of convexity and positive homogeneity together with being bounded by the maximum downside semideviation are sufficient for a typical inequality measure to guarantee the corresponding fair consistency. It allows us to identify various inequality measures which can be effectively used to incorporate fairness factors into various resource allocation problems while preserving the consistency with outcomes maximization. Among others the mean semideviation turns out to be such a consistent inequality measure while the mean absolute difference is strongly consistent. In multiagent allocation problems another way of defining a fair allocation as a bicriteria decision problem would be to take as second criterion a measure of envy-freeness [5]. This could lead to another class of significant questions on consistency for further research.

The considered fairness model is primarily well suited for the centralized resource allocation problems, like the bandwidth allocation problem [3,9,34]. Nevertheless, the classical unweighted fairness models are used as the basis for some distributed systems managements (c.f., [38]). The analyzed bicriteria fairness models may be considered as introduction of a compensation term into the utilitarian model. Hence, they may help to develop fair distributed mechanisms. It seems to be a promising direction for further research on possible implementations for specific environments.

Our analysis is related to the properties of solutions to resource allocation models. It has been shown how inequality measures can be included into the models avoiding contradiction to the maximization of outcomes. We do not analyze algorithmic issues for the specific resource allocation problems. Generally, the requirement of the measures convexity necessary for the fair consistency, guarantees that the corresponding optimization criteria belong to the class of convex optimization, not complicating the original resource allocation model with any additional discrete structure. Most of the inequality measures, we analyzed, can be implemented with auxiliary linear programming constraints thus offering reasonable optimization models for continuous as well as discrete problems [18]. Actually, among the measures of Table 1 only the standard semideviation leads to nonlinear optimization while the maximum semideviation, the mean absolute semideviation as well as the mean absolute difference are LP implementable. Nevertheless, further research on efficient computational algorithms for solving the specific models is necessary.

References

1. Atkinson, A.B.: On the measurement of inequality. J. Economic Theory 2, 244–263 (1970)
2. Bansal, N., Sviridenko, M.: The Santa Claus problem. In: Proc. of STOC 2006, Seattle, pp. 31–41 (May 2006)
3. Bertsekas, D., Gallager, R.: Data Networks. Prentice-Hall, Englewood Cliffs (1987)
4. Bouveret, S., Lemaitre, M.: Computing leximin-optimal solutions in constraints networks. Artificial Intell. 173, 343–364 (2009)
5. Bouveret, S., Lang, J.: Efficiency and envy-freeness in fair division of indivisible goods: logical representation and complexity. In: Proc. IJCAI 2005, Edinburgh, pp. 935–940 (August 2005)
6. Bonald, T., Massoulie, L.: Impact of fairness on Internet performance. In: Proc. of ACM Sigmetrics, pp. 82–91 (June 2001)
7. Brams, S.J., Taylor, A.D.: Fair Division: From Cake Cutting to Dispute Resolution. Cambridge Univ. Press, New York (1996)
8. Dalton, H.: The measurement of the inequality of income. Economic J. 30, 348–361 (1920)
9. Denda, R., Banchs, A., Effelsberg, W.: The fairness challenge in computer networks. In: Crowcroft, J., Roberts, J., Smirnov, M.I. (eds.) QofIS 2000. LNCS, vol. 1922, pp. 208–220. Springer, Heidelberg (2000)
10. Golden, B., Perny, P.: Infinite order Lorenz dominance for fair multiagent optimization. In: Proc. of AAMAS 2010, Toronto, pp. 383–390 (May 2010)
11. Goldsmith, J., Sloan, R.: The AI conference paper assignment problem. In: Proc. AAAI Workshop on Preference Handling for Artificial Intelligence, Vancouver, pp. 53–57 (July 2007)
12. Ibaraki, T., Katoh, N.: Resource Allocation Problems, Algorithmic Approaches. MIT Press, Cambridge (1988)
13. Jaffe, J.: Bottleneck flow control. IEEE Trans. Communications 7, 207–237 (1980)
14. la Kelly, F., Mauloo, A., Tan, D.: Rate control for communication networks: shadow prices, proportional fairness and stability. J. Opnl. Res. Soc. 49, 206–217 (1997)
15. Kleinberg, J., Rabani, Y., Tardos, E.: Fairness in routing and load balancing. J. Comput. Syst. Sci. 63, 2–21 (2001)
16. Kostreva, M.M., Ogryczak, W.: Linear optimization with multiple equitable criteria. RAIRO Oper. Res. 33, 275–297 (1999)
17. Kostreva, M.M., Ogryczak, W., Wierzbicki, A.: Equitable aggregations and multiple criteria analysis. Eur. J. Opnl. Res. 158, 362–367 (2004)
18. Lesca, J., Perny, P.: LP Solvable Models for Multiagent Fair Allocation problems. In: Proc. of ECAI 2010, Lizbon (August 2010)
19. Luss, H.: On equitable resource allocation problems: A lexicographic minimax approach. Oper. Res. 47, 361–378 (1999)
20. Marchi, E., Oviedo, J.A.: Lexicographic optimality in the multiple objective linear programming: the nucleolar solution. Eur. J. Opnl. Res. 57, 355–359 (1992)

21. Marshall, A.W., Olkin, I.: Inequalities: Theory of Majorization and Its Applications. Academic Press, New York (1979)
22. Moulin, H.: Axioms of Cooperative Decision Making. Cambridge Univ. Press, New York (1988)
23. Nash, J.F.: The bargaining problem. Econometrica 18, 155–162 (1950)
24. Ogryczak, W.: On the Distribution Approach to Location Problems. Comp. Indust. Engg. 37, 595–612 (1999)
25. Ogryczak, W.: Inequality measures and equitable approaches to location problems. Eur. J. Opnl. Res. 122, 374–391 (2000)
26. Ogryczak, W.: Comments on properties of the minimax solutions in goal programming. Eur. J. Opnl. Res. 132, 17–21 (2001)
27. Ogryczak, W.: On principles of fair resource allocation for importance weighted agents. In: Proc. of SOCINFO 2009, Warsaw, pp. 57–62 (2009)
28. Ogryczak, W., Ruszczyński, A.: Dual stochastic dominance and related mean-risk models. SIAM J. Optimization 13, 60–78 (2002)
29. Ogryczak, W., Śliwiński, T.: On equitable approaches to resource allocation problems: the conditional minimax solution. J. Telecom. Info. Tech. 3(02), 40–48 (2002)
30. Ogryczak, W., Śliwiński, T.: On direct methods for lexicographic min-max optimization. In: Gavrilova, M.L., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Laganá, A., Mun, Y., Choo, H. (eds.) ICCSA 2006. LNCS, vol. 3982, pp. 774–783. Springer, Heidelberg (2006)
31. Ogryczak, W., Wierzbicki, A.: On multi-criteria approaches to bandwidth allocation. Control and Cybernetics 33, 427–448 (2004)
32. Ogryczak, W., Wierzbicki, A., Milewski, M.: A multi-criteria approach to fair and efficient bandwidth allocation. OMEGA 36, 451–463 (2008)
33. Pigou, A.C.: Wealth and Welfare. Macmillan, London (1912)
34. Pióro, M., Medhi, D.: Routing, Flow and Capacity Design in Communication and Computer Networks. Morgan Kaufmann, San Francisco (2004)
35. Rawls, J.: Justice as fairness. Philosophical Review LXVII, 164–194 (1958)
36. Rawls, J.: The Theory of Justice. Harvard University Press, Cambridge (1971)
37. Rothschild, M., Stiglitz, J.E.: Some further results in the measurement of inequality. J. Econ. Theory 6, 188–204 (1973)
38. Rzadca, K., Trystram, D., Wierzbicki, A.: Fair Game-Theoretic Resource Management in Dedicated Grids. In: CCGRID 2007, pp. 343–350 (2007)
39. Sen, A.: On Economic Inequality. Clarendon Press, Oxford (1973)
40. Steinhaus, H.: Sur la division pragmatique. Econometrica 17, 315–319 (1949)
41. Young, H.P.: Equity in Theory and Practice. Princeton Univ. Press, Princeton (1994)

Dynamic Context-Sensitive PageRank for Expertise Mining*

Daniel Schall and Schahram Dustdar

Distributed Systems Group, Vienna University of Technology
Argentinierstr. 8/184-1, A-1040 Vienna, Austria
{schall,dustdar}@infosys.tuwien.ac.at

Abstract. Online tools for collaboration and social platforms have become omnipresent in Web-based environments. Interests and skills of people evolve over time depending in performed activities and joint collaborations. We believe that ranking models for recommending experts or collaboration partners should not only rely on profiles or skill information that need to be manually maintained and updated by the user. In this work we address the problem of expertise mining based on performed interactions between people. We argue that an expertise mining algorithm must consider a person's interest and activity level in a certain collaboration context. Our approach is based on the PageRank algorithm enhanced by techniques to incorporate contextual link information. An approach comprising two steps is presented. First, offline analysis of human interactions considering tagged interaction links and second composition of ranking scores based on preferences. We evaluate our approach using an email interaction network.

1 Introduction

The collaboration landscape has changed dramatically over the last years by enabling users to shape the Web and availability of information. While in the past collaborations were bounded to intra-organizational collaborations using a companies specific platform, and also limited to messaging tools such as email, it is nowadays possible to utilize the knowledge of an immense number of people participating in collaborations on the Web. The shift toward the Web 2.0 allows people to write blogs about their activities, share knowledge in forums, write Wiki pages, and utilize social platforms to stay in touch with other people.

Task-based platforms for human computation (i.e., *crowdsourcing*) including Amazon Mechanical Turk[1] enable access to the manpower of thousands of people on demand by creating human-tasks that are processed by the crowd. Human computation techniques applied in service-oriented environments involving Human-Provided Services (HPS) [1] enable the dynamic discovery and composition of experts. In such open and dynamic environments it becomes essential to manage expertise profiles and reputation of people in an automated manner.

* This work is supported by the EU FP7 project COIN (No. ICT-2008-216256).

[1] Amazon Mechanical Turk: http://www.mturk.com/

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 160–175, 2010.

Somebody seeking help or advice on a specific problem or businesses issuing task requests using, for example, before mentioned platforms need to be able to find the right person who can assist with offering his/her expertise. Recent work in expert finding (e.g., [2]) has been addressing the search for persons with the right skill level by using ontologies and by combining diverse semantic information from user/skill profiles. Since Web-scale collaborations involving a large amount of people not only demand for scalable algorithms and ranking solutions, but in many cases it is also desirable to consider the global properties of a human interaction network to determine the importance of users. Algorithms and models designed for ranking Web documents in the Web graph such as *PageRank* [3] or the *Hubs and Authorities* model (Hyperlink-Induced Topic Search [4] abbreviated as HITS) have been applied in online communities (e.g., Java discussion forums) [5,6] for expertise mining.

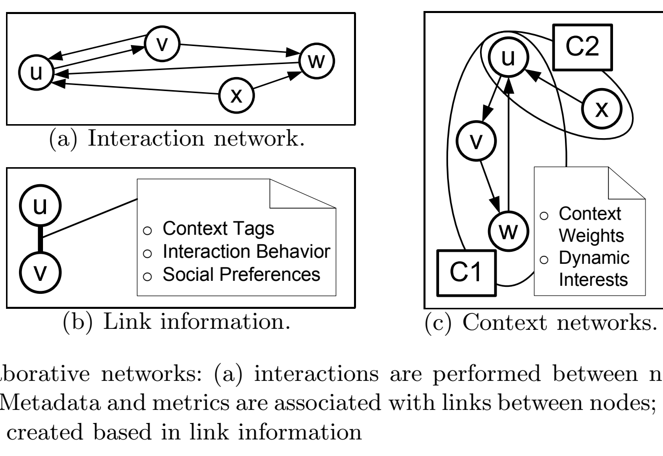
In this paper we present the following key contributions:

- We propose link analysis techniques derived from the popular PageRank model. Our approach is called *Dynamic Skill and Activity-based PageRank* (DSARank) accounting for activity and contextual link information.
- We present metrics to capture dynamic properties in human collaborations.
- Evaluation and discussion of our approach using a real email interaction dataset. Our experiments confirm that our proposed ranking approach is well suited for expertise mining.

2 Techniques for Expertise Ranking

2.1 Context-Sensitive Interaction Mining

Our approach is based on mining of interactions between users. Therefore, an interaction network (see Fig. 1(a)) is modeled as a graph $G = (N, E)$ composed of the set of nodes N and the set of edges E. Note, in this work the terms edge and link have the same meaning.



(a) Interaction network. (b) Link information. (c) Context networks.

Fig. 1. Collaborative networks: (a) interactions are performed between nodes in the network; (b) Metadata and metrics are associated with links between nodes; (c) Context networks are created based in link information

We argue that context information is essential for expertise mining. The context of an interaction can be captured by, for example, extracting relevant keywords from messages exchanged between users or by tags applied to various collaboration artifacts. In this work, we focus on *tags* (Fig. 1(b)) serving as input for contextual link information. Interaction metrics such as weights depicting the interest and focus of a user to collaborate with other peers in a specific context are automatically calculated through mining. Fig. 1(c) shows networks for context $C1$ and $C2$. Each context network may have one or more tags associated with it.

Existing work in the area of expertise mining (e.g., [5,6]) typically focuses on a graph representation as depicted by Fig. 1(a). In contrast, we present an approach and algorithm that is suitable for scenarios as shown in Fig. 1(c). Before presenting the detailed mechanisms, we overview existing link analysis techniques.

2.2 Link Analysis Approaches

We focus on two well-established techniques (i.e., PageRank and HITS) that have attracted extensive research (see [7]) in the information retrieval and theoretical computer science community for over a decade. Here we outline the basic principles of both approaches and discuss in which cases one technique should be favored over the other in expertise mining applications. Using the well-known PageRank [3], the importance of a node u is defined as:

$$PR(u) = \overbrace{\frac{(1-\alpha)}{|N|}}^{personalization} + \overbrace{\alpha \cdot \sum_{(v,u)\in E} \frac{PR(v)}{\texttt{outdegree}(v)}}^{\text{importance propagation}} \tag{1}$$

The equation consists of two components. The first part (*personalization*) can be used to assign preferences to certain nodes. Without any preferences or bias towards a particular node, a personalization vector $p(u) = 1/|N|$ is assumed. The second part of Eq. 1 propagates importance scores between connected nodes. Page and Brin [3] originally proposed a 'teleportation' factor $\alpha = 0.85$ based on the *random surfer model*. The damping factor $d = (1-\alpha)$ is based on the assumption that a user on the Web typically follows six links ($d = 1/6 = 0.15$).

A quite intuitive justification for applying PageRank in expertise ranking applications is the expected 'informedness' of users. Each link between people can be interpreted as a channel to propagate information in the network. The strength of a link limits the flow of information between two nodes v and u.

Table 1. PageRank and related symbols

Symbol	Description
α	The PageRank 'teleportation' factor (a value between 0.8 and 0.9).
$\|N\|$	The number of nodes in G.
$\boldsymbol{PR}$	The PageRank vector $\boldsymbol{PR}$ holding importance scores for each node.
$\texttt{outdegree}(v)$	The number of links originating from v.

For example, v may notify one of its neighbors u about some news or forward information to a newly chosen node.

The second approach discussed in this work for expertise ranking is HITS [4]. The HITS model has been proposed for expertise ranking in, for example, question and answer communities (Q/A) [6]. In such environments, hubs are characterized by users posting questions attracting answers from other knowledgeable users, representing authorities in the Q/A community.

$$
\begin{aligned}
H(u) &= \sum_{(u,v)\in E} A(v) \\
A(v) &= \sum_{(u,v)\in E} H(u)
\end{aligned}
\tag{2}
$$

The benefit of the HITS approach is the availability of two ranking scores for each node (see Table 2). Using the concept of hubs and authorities is beneficial in task-based *online help and support* environments to support the coordination of delegation flows [8].

Table 2. HITS and related symbols

Symbol	Description
$H(u)$	Hub score of u characterized by the authority of nodes u points to.
$A(v)$	Authority score of v influenced by hubs pointing to v.

To briefly outline this idea. A member of the online help and support community (modeled as HPS [1]) may receive a request for support (RFS) and delegate work to some other peer in the network (characterizing hubs in the network). Receivers of the delegated work, however, expect RFSs fitting their skills and expertise (i.e., being an authority in the given domain). Careless delegations of work will overload these peers resulting in degraded processing time due to missing expertise. Within the online help and support community, authorities give feedback using rating mechanism (e.g., a number on the scale from 1 to 5) to indicate their satisfaction - whether a particular hub distributes work according to their skills and interest. A 'good hub' is thereby characterized by a neighborhood of peers that are satisfied with received RFSs. On the other hand, delegation of work is strongly influenced by trust, for example, whether the initial receiver of the RFS (hub within the community) expects that the peer will process work in a reliable and timely manner. Thus, RFS receivers need to be trusted by influential hubs that are highly rated in order to be recognized as authoritative peers in the community.

2.3 PageRank vs. HITS

Both PageRank and HITS follow similar principles by making use of the link structure in networks to estimate the importance of nodes. The benefit of

PageRank is that it takes the global network into account. The PageRank model is based on the interpretation of a graph G as a full stochastic transition matrix achieved by assigning teleportation probabilities to each node (uniform by using a personalization vector $p(u) = 1/|N|$). This fact helps bootstrapping the introduction of new nodes by using, for example, profile similarity-based trust prediction [9] to address the newcomer problem.

HITS operates on a small portion of the graph, which is typically query dependent. Query dependent subgraphs permit online computation of hub and authority scores. Our previous work [8] addressed online computation of HITS using context information in collaborative environments. Furthermore, hub and authority scores are available for each node. This not only helps in finding authoritative nodes, but also *coordinators* (i.e., hubs) distributing work in online help and support communities.

3 Dynamic Skill and Activity-Aware PageRank

Based on the previous discussions on techniques for expertise mining, we propose *DSARank* to account for ranking of experts in context networks. Such networks carry context information through tagged interaction links expressing the importance of a link for a specific collaboration context.

3.1 Definition

The main idea of our approach is to use *preferences* based on a set of interaction metrics to calculate personalized PageRank vectors. We define *DSARank* as:

$$DSARank(u) = \overbrace{(1-\alpha)\cdot\sum_{w_m \in W_{M'}} w_m p_m(u)}^{\text{context-sensitive personalization}} + \overbrace{\alpha \cdot \sum_{(v,u)\in E} W(v,u) DSARank(v)}^{\text{importance propagation}} \tag{3}$$

The approach as shown in Eq. 3 is straightforward. For simplicity assume – without considering more advanced interaction metrics – that preferences should be assigned for finding experts in context $C1$:

$$p(u) \equiv \begin{cases} 1 & \text{, if } u \in g(C1) \\ 0 & \text{, otherwise} \end{cases} \tag{4}$$

Table 3. Metrics and weights in interaction networks

Symbol	Description
m	A metric (measured value) in interaction networks. The set $M' = \{m_1, m_2, \ldots, m_n\}$ defines interaction metrics with $M' \subseteq M$.
w_m	Denotes the weight of a metric to assign preferences for specific metrics. If not explicitly specified, equally important metrics are assumed with $W_{M'} = \{w \mid w_m = 1/\lvert M'\rvert, \forall m \in M'\}$. The sum of metric weights must be equal to 1.
$W(v,u)$	The weight of the edge $(v,u) \in E$ calculated as $W(v,u) = 1/\texttt{outdegree}(v)$.

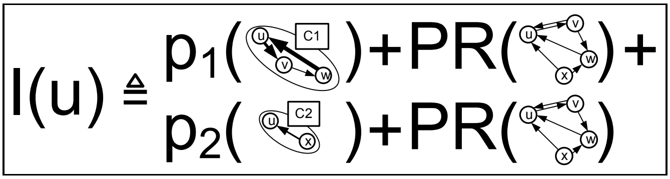


Fig. 2. Expertise ranking using DSARank: $I(u)$ is the importance of u given context networks $C1$ and $C2$. Context-sensitive ranking scores are composed as the linear sum of personalized PageRank vectors.

Preferences are thereby given to those users who have interacted in a particular context $C1$ with other users depicted as the subgraph $g(C1)$. For example, users have performed tasks in $C1$ or have been involved in discussions related to $C1$. The problem with this naïve approach becomes evident when computing Eq. 3 for different preferences – say $g(C1)$ and $g(C2)$ or some other combinations of context networks $C1 \ldots Cn$. The demanded set of skills and expertise of users in expert finder applications is typically only known at runtime (the *query* specified by the expert seeker). The online computation of Eq. 3 is, however, not feasible for large-scale interaction networks requiring up to several hours to calculate the importance of users.

We propose the PageRank linearity theorem to solve the problem of context-sensitive expertise mining. The linearity theorem [10,11] is defined as:

Theorem 1 (Linearity). *For any personalization vectors* $\boldsymbol{p}_1, \boldsymbol{p}_2$ *and weights* w_1, w_2 *with* $w_1 + w_2 = 1$, *the following equality holds:*

$$\boldsymbol{PPV}(w_1\boldsymbol{p}_1 + w_2\boldsymbol{p}_2) = w_1\boldsymbol{PPV}(\boldsymbol{p}_1) + w_2\boldsymbol{PPV}(\boldsymbol{p}_2) \tag{5}$$

The above equality states that personalized PageRank vectors $\boldsymbol{PPV}$ can be composed as the weighted sum of PageRank vectors. Fig. 2 illustrates this concept where $I(u)$ denotes the importance of u in the context networks $C1$ and $C2$. The linearity theorem is a very important concept for creating context-sensitive expertise scores because it enables the *online* combination of PageRank vectors based on the expert seeker's query preferences. Based on Eq. 5, Eq. 3 can be rewritten as:

$$\boldsymbol{DSARank}^Q = \sum_{C \in Q} w_C \overbrace{\boldsymbol{DSARank}(w_{m1}\boldsymbol{p}_{m1} + w_{m2}\boldsymbol{p}_{m2} \ldots + w_{mn}\boldsymbol{p}_{mn})}^{\text{context-sensitive importance}} \tag{6}$$

with $\sum_m w_m = 1$ (see also Table 3) and $\forall m : \sum_m p_m = 1$.

3.2 Ranking Algorithm

Algorithm 1 shows the steps to compute context-sensitive importance scores as defined in Eq. 6. The steps 2-7 are used to assign metric values to the personalization vector $p(u)$ by invoking the function `UpdatePrefVector`. The detailed definition of interaction-based metrics follows in the next section. Steps 8-16 show the iterative computation of importance scores. The temp vector $s(u)$ (see line 11) holds scores that are updated to $DSARank$ scores in line 14 using the PageRank teleportation parameter α.

```
   input : Graph G and metrics M'
   output: Expertise scores DSARank
1  /* initialize DSARank_1 */
2  for Node u in G do
3      for Metric m in M' do
4          UpdatePrefVector(p(u), m)
5      end
6      DSARank_1(u) ← p(u)
7  end
8  while not converged do
9      for Node u in G do
10         for Node v with (v, u) ∈ E do
11             s(u) ← DSARank_t(u) + W(v, u)DSARank_t(v)
12         end
13         /* update expertise scores */
14         DSARank_{t+1}(u) ← (1 − α) · p(u) + α · s(u)
15     end
16     /* Normalize rankings and test for convergence */
17 end
18 return DSARank
```

Algorithm 1. Calculating context-sensitive importance scores

Convergence is reached once the change in ranking scores does not exceed a predefined threshold ϵ:

$$||\boldsymbol{DSARank}_{t+1} - \boldsymbol{DSARank}_t|| \leq \epsilon \tag{7}$$

While Algorithm 1 shows the offline mining procedure, the actual context-dependent ranking scores ($\boldsymbol{DSARank}^Q$) are aggregated based on the expert seeker's query preferences.

Algorithm 2 shows the composition of expertise ranking scores based on a set of context parameters (e.g., keywords, context identifiers, or context topic descriptions) formulated as an expert query Q. The weight w_C (line 3) is the preference for a specific context parameter. For example, the preference of the expert seeker that a person (expert) has stronger expertise in a specific topic. Again, the weights must satisfy $\sum_C w_C = 1$. This is due to the fact that the personalization vector $\boldsymbol{p}$ has to represent a valid probability distribution.

input : Query Q specifying context parameters
output: Composite expertise scores
for *Node* $u \in G$ **do**
  for *Context parameter* C *in* Q **do**
    $DSARank^{Q}(u) \leftarrow DSARank^{Q}(u) + w_C DSARank^{C}(u)$
  end
end
return $DSARank^{Q}$

Algorithm 2. Calculating composite expertise scores

Algorithm 1 is a generic approach addressing the requirements of context-sensitive expertise mining. The key point is to define a set of metrics to capture dynamic user interests and context-dependent expertise. We have defined a broad range of context-sensitive link metrics in [12]. In the following section, we focus on a subset of metrics and show their impact in the evaluation section.

4 Interaction Metrics

The first metric defines the context (topic) sensitive interest (or focus) of a user. This can be measured as the *intensity* of a user interacting in a specific context compared to the user's focus on other contexts. Fig. 3 illustrates the basic idea of this approach. The interest and skills of people typically evolve over time. A collaboration partner may change his/her interaction behavior by focusing on, for example, a new topic of interest. The presented metric captures interaction intensities of all nodes in a given context. As a first step we create context dependent subgraphs as illustrated by Fig. 3. We assume that users tag their interactions (e.g., exchanged emails, delegated tasks, etc.) using a predefined set $T = \{t_1, t_2, \ldots, t_n\}$ of tags. Each tag appears with a certain frequency $f(t_n)$. Based on a set of accumulated tags between v and u we calculate a weighted edge $(v, u) \in E$:

$$W_C(v,u) \propto \frac{f(t_C) + \gamma}{\sum_{i=1..n}(f(t_{Ci}) + \gamma)} \tag{8}$$

The edge weight is essentially based on the number of tags associated by v when interacting with u. We apply a simple yet effective smoothing approach

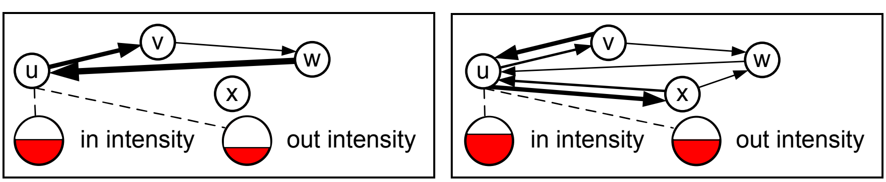


Fig. 3. Weighted context network measuring interaction intensity of nodes. Nodes interact in different contexts with different intensities.

called *Lidstone smoothing* with $0 < \gamma < 1$, which is a well known technique in information retrieval to account for missing or misplaced tags. Using Eq. 8, various exchanged (tagged) resources influence the interaction weight between users. We now turn to the definition of a metric to measure the *intensity* of interactions of a specific user. For example (see Fig. 3), u's incoming/outgoing links within certain subgraphs g are associated with different weights. Let us define the set of all links $L(u) = \texttt{outlinks}(u) \cup \texttt{inlinks}(u)$ originating from u ($\texttt{outlinks}(u)$) and pointing to u ($\texttt{inlinks}(u)$).

Definition 1 (Interaction Intensity). *Let us define the interaction intensity of u over a link l as:*

$$i_C(l;u) = \left[\frac{w(l;C)}{\sum_{l' \in L(u)} w(l';C)} \right] \times |l| \tag{9}$$

The weight $w(l;C)$ is based on the smoothened tag-based context weight using Eq. 8. The term in square brackets in Eq. 9 is the relative weight of the link l among all links attached to u. This is multiplied times $|l|$, the number of interactions associated with l (e.g., count of delegated tasks). Put in simple words, Eq. 9 combines three factors: *relevance of link for a given context* $w(l;C)$, *importance of link* '[...]' $\times$ *strength of link* $|l|$.

Next, we define two further metrics based on the definition of $i_C(l;u)$.

Definition 2 (Interaction Intensity Level). *We define the interaction intensity level IIL_C of u as:*

$$IIL_C(u) = \left[\beta^2 \left(\sum_{l \in \texttt{outlinks}(u)} i_C^{out}(l;u) \right)^2 + (2-\beta)^2 \left(\sum_{l \in \texttt{inlinks}(u)} i_C^{in}(l;u) \right)^2 \right]^{(1/2)} \tag{10}$$

Eq. 10 calculates the overall interaction intensity of u in a given context C. IIL_C basically depicts the user's interest and *activity level* in a specific context C. The factor $\beta \in [0,2]$ allows IIL_C to be biased towards $i_C^{out}(l;u)$ or $i_C^{in}(l;u)$, where 1 means no bias, i.e., equal importance for in-/out intensities. Biasing IIL_C is only done for *all* nodes in the network to have valid probabilities for $\boldsymbol{p}$.

Definition 3 (IIL Imbalance). *Let us define $imb_C(IIL) \in [-1,1]$ as:*

$$imb_C(IIL) = \frac{\sum_{l \in \texttt{inlinks}(u)} i_C^{in}(l;u) - \sum_{l \in \texttt{outlinks}(u)} i_C^{out}(l;u)}{\sum_{l \in L(u)} i_C(l;u)} \tag{11}$$

IIL imbalance $imb(IIL)$ is a useful measure to determine two extreme values: *Passive* involvement of users in interactions such as 'observers' or 'monitors' resulting in $imb(IIL)$ being equal to 1. *Active* involvement of users with the extreme case of all user interactions being outgoing (none of the interactions is reciprocated by others), thus $imb(IIL) = -1$. This metric is mainly used to constraint and filter relevant users, for example, to rank only those users within a range $0.1 < imb(IIL) < 0.9$.

We defined the previous (intensity) metrics to model a user's interest to interact with its neighbors in a particular context. These intensity metrics do not account for importance propagation within a contextual (sub)graph $g(C_n)$. For this purpose we define a metric to model interaction behavior and importance propagation in context networks. A user v may regard some interactions with u as important in a specific context. As mentioned before, the relevance of a link in a context C_n is proportional to the frequency of applied tags (cf. Eq. 8).

Definition 4 (Skill and Expertise Fingerprints). *Let us define the expertise of u in contextual subgraphs as:*

$$SE_C(u) = \sum_{(v,u)\in E} \left(\frac{W_C(v,u)}{\sum_{(v,w)\in E} W_C(v,w)} \right) SE_C(v) \tag{12}$$

We term the propagation of importance in contextual subgraphs (i.e., context networks) as Skill and Expertise $SE_C(u)$ *fingerprints*. $SE_C(u)$ can be interpreted as the *expected informedness* of a user u in context C to denote the chance of u receiving an item of information or that v will contact u in a specific context C.

5 Evaluation

To evaluate our ranking approach, we use a real email dataset because at the time of writing this paper, we did not have access to data generated by HPS-based crowdsourcing applications. To test the effectiveness of *DSARank*, we establish a set of ranking metrics to compare *DSARank* with the standard PageRank (as defined by Eq. 1) algorithm to obtain baseline results. In addition we select concrete examples of ranking results, for example a concrete collaboration context, and analyze how well these results reflect the actual expertise (or knowledge) of people. For evaluation, we use a set of well established ranking metrics.

5.1 Evaluation Metrics and Ranking Parameters

Kendall's τ. We use the definition of Kendall's τ by [13]. Consider the pairs of vertices v, w. The pair is *concordant* if two rankings agree on the order, *disconcordant* if both rankings disagree on the order, *exact-tie* if the exact ranking does not order the pair, and *approximate-tie* if the approximate ranking does not order the pair. Denote the number of these pairs by c, d, e, and a, respectively. Given the total number of pairs as $m_p = \frac{n(n-1)}{2}, n = |N|$, then Kendall's $\tau \in [-1, 1]$ is defined as: $\tau = (c-d)/\sqrt{(m_p - e)(m_p - a)}$. Kendall's τ helps to understand whether two algorithms are rank-similar. In other words, if τ equals 1, there are no cases where the pair v, w is ranked in a different order.

Top-k Set Metrics. The overlap similarity $OSim(T_{k1}, T_{k2})$ of the top-k sets T_{k1} and T_{k2} ranked by *DSARank* and PageRank is defined as $OSim(T_{k1}, T_{k2}) = \frac{T_{k1} \cap T_{k2}}{k}$ [10].

Ranking Parameters. We set the IIL parameter β to 1.2; thereby assigning a bias to out-intensities. We use a filter of $-0.9 < imb(IIL) < 0.9$. If $imb(IIL)$ of a user is not within this range, we 'downgrade' the user's $IIL_C(u)$ to 0 because links between users might be irrelevant if IIL is strongly imbalanced.

Teleportation Parameter α. There is some controversy whether the same parameter value as applied in the Web graph is suitable for different kind of networks (e.g., see [14] for a discussion on the damping factor in citation networks). In this work, we experimented with different values for α. Low α (e.g., $\alpha < 0.45$) results in unstable ranking results so that the proposed algorithm delivers quite different results compared to the baseline algorithm. In all experiments we use a factor of $\alpha = 0.85$ yielding good results as discussed in the following.

5.2 Dataset Properties

We use the Enron email interaction network[2], which is to our knowledge the only substantial collection of real email messages publicly available. The entire email graph comprises about 255635 messages and nearly 1647224 users (i.e., recipients of messages).

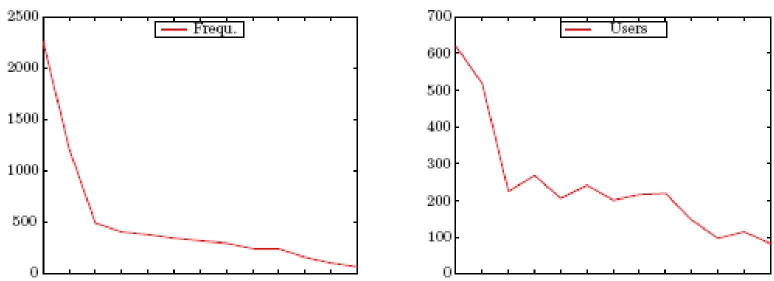
Tagged Message Corpus. A subset of messages of the entire message corpus was labeled by UC Berkeley's Natural Language Processing group. These tags were applied to about 1700 messages. The tagged message corpus serves well to test our context-based ranking approach. Different categories of tags were applied to interaction links comprising messages between people with the focus on business-related emails. 13 categories or *contexts* are available (Table 4). Tag statistics are depicted by Fig. 4.

Table 4. Primary categories in labeled interaction graph. The index establishes the correspondence to the tag-statistics in Fig. 4 (horizontal axis).

ID	Index	Description	ID	Index	Description
3.1	2	Regulations and regulators (includes price caps)	3.8	9	Internal company operations
3.2	5	Internal projects – progress and strategy	3.9	11	Alliances / partnerships
3.3	3	Company image – current	3.10	4	Legal advice
3.4	6	Company image – changing / influencing	3.11	13	Talking points
3.5	10	Political influence / contributions / contacts	3.12	8	Meeting minutes
3.6	1	California energy crisis / California politics	3.13	12	Trip reports
3.7	7	Internal company policy			

[2] Email Dataset: `http://bailando.sims.berkeley.edu/enron_email.html`

Applied Expansion and Filtering. We expand the subset of labeled messages by searching the entire email-message corpus for related messages. For example, given a labeled message, we search for messages which are most likely related to the labeled message; for example, in `reply-to` or `forwarded` messages. Thereby, we expand the existing labeled message corpus by adding 5248 related messages. However, some messages are simply 'broadcast' messages (e.g., announcements or periodic auto-generated messages from a person), which we filter out because these messages might distort ranking results. In addition, sender and recipient of messages must be real people (e.g., we filter messages *from* and *to* distribution lists) because we focus on importance rankings of people.



(a) Tag frequency given as the total number of occurrences.

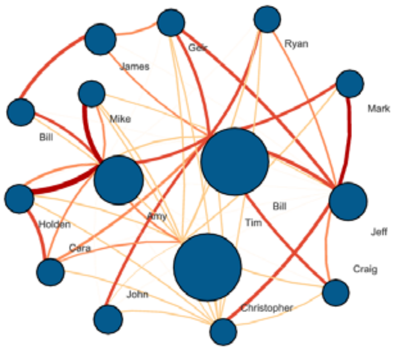
(b) Distinct number of users in an interaction context.

Fig. 4. Tag statistics in email interactions

5.3 Basic Results

As a first example, we select the subgraph for ID 3.6 (i.e., the specific interaction context) and rank all users. The results are detailed in Fig. 5 by a graph visualization and a detailed description in the table. The goal of testing *DSARank* in context networks is to verify whether the obtained ranking results contain the most knowledgeable people in a specific context. Using *DSARank* in category 3.6, we verified that the top-ranked users were indeed key people in the given context. Not only the number of inbound interactions with important users plays a key role, but also *interaction intensity* in a given context dependent subgraph. It is well possible that users get in some cases substantially promoted (i.e., higher rank) because they interact with important users in a given context with high intensity. *DSARank* provides accurate results since users are not ranked in a single context. The primary goal of this work was not to understand in detail the roles and whereabouts of the email dataset. Therefore, we do not provide the details in this paper. For validation of our results, we provide details[3] about the user identities as well as descriptions of specialism (where available).

[3] Email Mining: `http://www.infosys.tuwien.ac.at/staff/dschall/email/`



ID	scoreA	rankA	scoreB	rankB	IIL	imb
37	0.109475	1	0.004121	21	7.31	-0.81
8	0.102375	2	0.020758	1	5.13	0.11
90	0.043071	3	0.008326	9	1.10	0.08
253	0.029936	4	0.001733	170	2.07	-0.85
347	0.020443	5	0.001665	282	1.39	-0.87
92	0.016677	6	0.003912	23	0.39	0.82
152	0.016375	7	0.013148	2	1.16	1.00
47	0.014248	8	0.003593	27	0.66	0.41
29	0.014195	9	0.005415	16	1.14	1.00
14	0.014084	10	0.010911	4	2.27	1.00

User ID (ordered by rank)	37	8	90	253	347	92	152	47	29	14
User Name (see graph)	Tim	Bill	Amy	Jeff	James	Holden	John	Cara	Bill	Mark

Fig. 5. Filtered interaction graph of selected collaboration context and results: columns `scoreA/rankA` show results by *DSARank* and `scoreB/rankB` by unbiased PageRank

5.4 Comparison of Composite Contexts

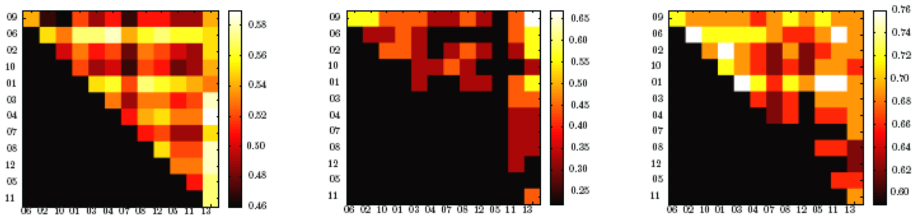
Here we analyze the impact of composite context-sensitive ranking results. We rank in different subgraphs and combine results using Eq. 6 to create composite *DSARank* scores. Each context-dependent result vector $\boldsymbol{DSARank}^{C1}$ and $\boldsymbol{DSARank}^{C2}$ is combined with $w_{C1} = w_{C2}$. To create PageRank scores, we use the entire interaction graph G to create the vector $\boldsymbol{PR}$. Furthermore, let us define a metric to compare the 'dominance' of context-dependent subgraphs given the set of contexts $C_N = \{C1, C2, \ldots, Cn\}$.

We calculate the intensity of all users $U(C)$ in $g(C)$ as subgraph intensity:

$$i(g(C)) = \frac{1}{|U(C)||C_N|} \sum_{i=1..n} \sum_{u \in U(Ci)} IIL_{Ci}(u) \tag{13}$$

Each subfigure in Fig. 6(a) shows the comparison of *DSARank* and PageRank ranking results using previously introduced ranking metrics. We start with the first Fig. 6(a) depicting Kendall's τ.

Kendall's τ. Fig. 6(a) visualizes the results to indicate whether there is a strong disagreement (low value for Kendall's τ) when combining different contexts. On the other hand, if Kendall's τ is equal to 1, then ranking considering context information does not change the ranking order (i.e., using context for expertise mining does not bring any additional benefit). Contextual graphs that share links with other context graphs (i.e., links with a shared set of tags) yield stronger agreements between *DSARank* and PageRank. This is the case in, for example, categories 3.6 and 3.1 (see IDs in Table 4), which share a high degree of links (tags). Thus, 3.6 and 3.1 become more dominant when combined with other contexts. It is therefore less likely that the order of rankings change.



(a) Kendall's τ. (b) $OSim_{k=10}$. (c) $OSim_{k=30}$.

Fig. 6. Comparison of composite contexts using Kendall's τ and $OSim$ metrics

On the other hand, if a context, for example category 3.13 (the corresponding subgraph) has few shared links with other contexts; and also low subgraph intensity (3.13 has the lowest subgraph intensity), then we observe also stronger agreements in rankings. This can be explained as the limited impact of low intensity contexts on changing the position of users within ranking results.

Overlap Similarities $OSim$**.** We compared $DSARank$ and PageRank in terms of overlap similarities. In Figure 6(b) and 6(c) we show the visualizations of the results. By comparing the top-10 segment of ranked users (Figure 6(b)), we see higher overlap similarities between high-intensity contexts, for example, the context pairs (3.9, 3.6), (3.9, 3.2). Low intensity contexts such as 3.13 combined with, for example (3.13, 3.9), yields also high similarities. The top-30 segment (Figure 6(c)) shows stronger similarities in 3.1 as well as 3.6; both contexts have many shared links with other contexts.

To conclude our experiments, PageRank provides a valuable method for expertise ranking. This has also been confirmed by other research (e.g., see [5]). However, 'context' is still a key issue in expertise ranking as the level of user expertise varies depending on context. Based on discussions of Kendall's τ and $OSim$ to compare ranking approaches, parameterization (i.e., personalization) through interaction metrics is important to model user interest, activity level, and expertise in context networks.

6 Related Work

Task-based platforms on the Web allow users to share their expertise [15]. Also users offer their expertise by helping other users in forums or Q/A communities [6,16]. By analyzing email conversations [17], the authors studied graph-based algorithms such as HITS and PageRank to estimate the expertise of users. The authors in [18] followed a graph-entropy model to measure the importance of users. In [19], an email analysis in enterprises, defining information flow metrics in the social interaction graph was presented. The work by [5] followed a graph-based approach and applied HITS as well as PageRank in online communities (i.e., a Java question and answer forum). The authors in [20] studied minimum spanning tree algorithms in social networks.

While the above cited works attempted to model the importance of users based on interactions; they did not consider that interactions typically take place in different *contexts*. Approaches for calculating personalized PageRank scores [10,11] were introduced to enable topic-sensitive search on the Web. The novelty of our approach is that expertise mining is performed by considering contextual link information.

7 Conclusion

In this work we have motivated the need for context-sensitive expertise mining in open collaboration environments. As a first step, we compared popular and well established methods including PageRank and HITS. Most existing approaches utilizing these algorithms, however, do not consider context information for expertise mining. Here we systematically derived a context-sensitive expertise mining technique and algorithm called $DSARank$. Furthermore, we defined a set of metrics to measure collaboration characteristics to capture user-interest and contextual skills.

We are currently working on HPS testbeds to test different interaction models in crowdsourcing applications. Interactions in the testbed are based on Web service technology such as SOAP-based messages exchanged between HPSs and dynamically routed by means of delegations in HPS networks. This will help to perform more experiments in realistic service-oriented crowdsourcing scenarios.

References

1. Schall, D., Truong, H.L., Dustdar, S.: Unifying Human and Software Services in Web-Scale Collaborations. IEEE Internet Computing 12(3), 62–68 (2008)
2. Becerra-Fernandez, I.: Searching for experts on the Web: A review of contemporary expertise locator systems. ACM Trans. Inter. Tech. 6(4), 333–355 (2006)
3. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. Technical report (1998)
4. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM 46(5), 604–632 (1999)
5. Zhang, J., Ackerman, M.S., Adamic, L.: Expertise networks in online communities: structure and algorithms. In: WWW, pp. 221–230. ACM, New York (2007)
6. Jurczyk, P., Agichtein, E.: Discovering authorities in question answer communities by using link analysis. In: CIKM 2007, pp. 919–922. ACM, New York (2007)
7. Berkhin, P.: A survey on pagerank computing. Internet Mathematics 2, 73–120 (2005)
8. Schall, D., Skopik, F., Dustdar, S.: Trust-based discovery and interactions in mixed service-oriented systems. Technical Report TUV-1841-2010-01 (2010)
9. Skopik, F., Schall, D., Dustdar, S.: Start trusting strangers? bootstrapping and prediction of trust. In: Vossen, G., Long, D.D.E., Yu, J.X. (eds.) WISE 2009. LNCS, vol. 5802, pp. 275–289. Springer, Heidelberg (2009)
10. Haveliwala, T.H.: Topic-sensitive pagerank. In: WWW, pp. 517–526 (2002)
11. Jeh, G., Widom, J.: Scaling personalized web search. In: WWW, pp. 271–279. ACM, New York (2003)

12. Schall, D.: Human Interactions in Mixed Systems - Architecture, Protocols, and Algorithms. PhD thesis, Vienna University of Technology (2009)
13. Fogaras, D., Csalogany, K., Racz, B., Sarlos, T.: Towards scaling fully personalized pagerank: Algorithms, lower bounds, and experiments. Internet Mathematics 2(3), 333–358 (2005)
14. Maslov, S., Redner, S.: Promise and pitfalls of extending google's pagerank algorithm to citation networks. J. Neurosci. 28(44), 11103–11105 (2008)
15. Yang, J., Adamic, L., Ackerman, M.: Competing to share expertise: the taskcn knowledge sharing community. In: Conference on Weblogs and Social Media (2008)
16. Agichtein, E., Castillo, C., Donato, D., Gionis, A., Mishne, G.: Finding high-quality content in social media. In: WSDM 2008, pp. 183–194. ACM, New York (2008)
17. Dom, B., Eiron, I., Cozzi, A., Zhang, Y.: Graph-based ranking algorithms for e-mail expertise analysis. In: DMKD 2003, pp. 42–48. ACM, New York (2003)
18. Shetty, J., Adibi, J.: Discovering important nodes through graph entropy the case of enron email database. In: LinkKDD 2005, pp. 74–81. ACM, New York (2005)
19. Karagiannis, T., Vojnovic, M.: Email Information Flow in Large-Scale Enterprises. Technical report, Microsoft Research (2008)
20. Lappas, T., Liu, K., Terzi, E.: Finding a team of experts in social networks. In: KDD 2009, pp. 467–476. ACM, New York (2009)

Trust-Based Peer Assessment for Virtual Learning Systems

Milorad Tosic and Valentina Nejkovic

Intelligent Information Systems Lab, Department of Computer Science
Faculty of Electronic Engineering, University of Nis, Serbia
mbtosic@acm.org, valentina@elfak.ni.ac.rs

Abstract. In order to deal with new requirements imposed by emerging learning environments following social computing paradigm, we address the problem of assessment of individual student skills, contributions, and activities. Rather then clicking links to launch tools or to view content, such learning environments encourage more of a monitoring mode of operation that is very difficult to sense and record by the software alone. In this paper we propose adoption of peer-assessment approach in order to overcome the obstacle as well as to make the overall solution scalable. We propose a novel method for students peer-assessment based on trust concept. The overall approach is presented and practical experiments are conducted using developed web service. The grade scores determined by the learning peers/students are statistically proven as highly correlated with those marked by the teachers, indicating that approach proposed in this paper may be adopted as a legitimate assessment method.

Keywords: wiki, education, teaching and learning, peer-assessment, trust.

1 Introduction

The advent of social computing and Web 2.0 technologies in particular enabled education to grow towards student centered personalized learning environments [1]. Social computing imply collaboration, knowledge sharing and customisation, which provide educationalists with significant opportunities for creating socially engaging tasks that require active student participation and knowledge building [2], [3], [4]. However, social computing is not a ready-made of-the-shelf solution to be taken and adopted in educational institutions. Rather, we characterize social computing as a flexible toolbox that facilitates creation of innovative learning practices [5], [4], [6]. Many educational institutions have addressed wiki as appropriate social computing tool to be included into teaching and learning scenarios, and we witness its use in many different ways, from simple course deliver materials to collaborative content and projects developments, students progress tracking etc. [7], [8], [9]. Evident benefits of adopting wikis in education are improvements in efficiency and scalability of the teaching process [9]. The traditional way of teaching (simple delivering of teaching materials) is replaced with

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 176–191, 2010.

a new wave of learning processes that are presented as growing, accumulating, adopting and sharing knowledge among students.

For the purpose of this research, we adopt the concept of blended learning [10] theory that further includes mix of active learning, computer-supported collaborative learning and problem-based learning techniques with wiki system as support in teaching and learning activities. Such wiki-based teaching and learning follows cognitive constructivism [11] in a sense that students learn by actively constructing new knowledge, not by passive digest of the served information. It is important that the student manage his/her own activities [2]. Teachers as well as other group members continuously monitor students activity. Students have insight into each others work, thoughts, and contributions [2]. In their individual work, students work on problem-based projects within information systems domain. The work is divided into several units where each unit requires finishing a given task and writing corresponding reports. The work that each student invests into preparing the report is reflected by the quantitative activity indicators, such as number of created wiki pages, structure of the created pages, quality and quantity of published content, discussions at traditional classrooms as well as forum discussions. These indicators are next aggregated and used for assessment of students learning outcomes [12], but ensuring objective grading scores remains a challenging task since used learning methods consist of heterogeneous activities. Note that the indicators are limited to active participation indicators only. For example, read without responding can not be used as the indicator in spite of the correlation between the number of messages read against student performance.

Student assessment is even more challenging in emerging Personal Virtual Learning Environments (PVLE) based on the latest Social Web and Semantic Web technologies such as Widgets and Mashups. In this case Widgets (small embedded Web applications) update in real-time without any page refresh effecting end user behaviour. Rather then clicking links to launch tools of to view content, the widgets encourage more of a monitoring mode of operation with users navigating to a course page, then leaving it open in the background occasionally binging it into focus to see if any new conversations were happening within the collection of present widgets [13]. As a consequence, using quantitative activity indicators, such as page views histogram for example, for students learning outcomes assessment becomes inappropriate.

In this paper, we adopt principles of peer-assessment in approaching the problem of assessment of students work when social computing tools with activity based learning are used. Peer-assessment is used in teaching practice since 18th century [14]. Many studies have explored its impacts on improvements of students learning efficiency as well as assessment process [15], [16], [17], [18], [19], [20], [21], [22]. The most common peer-assessment method is grading score method, where students/peers assess other peers by giving grades [23].

The influence of trust of peers within peer-assessment methods has not been addressed in empirical studies on peer-assessment [24]. In this paper, we take trust concept [25] (meaning trust of peers) in consideration. We aim to explore how factor of trust of peers as assessors influences the reliability of grading scores method and does grading score method can be improved including trust-based decision. We design a trust-propagation based mechanism for generating grading scores based on self-assessment in social network based learning environment.

The paper is organized as follows. The Section 2 gives background on social computing, wiki concept, peer-assessment and trust as well as research experience and problem identification. In Section 3, peer assessment as trust propagation is discussed in more details. Section 4 presents our approach to grading scores with trust-based decision details, and gives results of case study concerning impact of the approach on students assessments. Section 5 concludes the paper.

2 Background

2.1 Social Computing

The recent advent of Web 2.0 and related technologies brings a new paradigm in ways we communicate, learn, and educate, known as Social Computing. Social computing seeks to improve enterprise and interpersonal information flow and recorded memory, improve how people create, manage, categorize, seek, obtain, evaluate and use information, break down barriers, and extend the use of information and information technologies.

Developments in Social Computing have created a disruptive impact on diverse areas of human life, such as industry, citizenship, identity, social inclusion, education, health and public governance [3]. Nowadays we address one or combination of several of the following applications, usually used by hundreds of millions of users worldwide [5], as social computing, such as: blogs, podcasts, forums, mashups, virtual worlds, social bookmarking, social news, collaborative platforms for content creation and sharing (such as wikis), social networking websites, search engines, auction websites, games, collaborative filtering of content, file, photo and video sharing, tagging and annotation, Voice over IP and peer-to-peer services. In addition to the application aspect, technological infrastructure and theoretical foundation represent orthogonal core aspects of social computing. From the structural view, the social computing may be comprehended as a community intelligent system modeled as a knowledge network that is comprised of triple interwoven networks of the media network, the human network, and the knowledge network [26].

Due to its strong multidisciplinary character, social computing may be defined in different ways depending on the particular application domain. We prefer the computation theory oriented definition given in [27], where social computing is defined as *"computational facilitation of social studies and human social dynamics as well as the design and use of ICT technologies that consider social context"*.

Social platforms are generating new ways how we acquire, access, manipulate, process, retrieve, present, and visualize information in the teaching and learning space. The social media for education has become dynamic, ubiquitous, distributed, real-time, collaborative, bottom-up, many-to-many, value-based, and personalized, and brought the new wave of changes in education. It provides resources, connections and new tools for creativity and collaboration, which empower all actors in the educational environments in new ways, in structured as well as unstructured learning settings.

2.2 Wiki in Education

The first documented use of a wiki in educational purposes was at the college level. In late 1997, researchers at Georgia Institute of Technology built the initial version of CoWeb, a variation on Ward Cunninghams original wiki WikiWikiWeb [8]. Researchers and instructors at other institutions also began experimenting with CoWeb and other versions of wiki. Experience reports and personal observations of wiki use in the classroom have also proliferated as teachers begin experimenting and sharing their practices [8]. As wiki use in education has become more visible, it has begun being applied at secondary school levels all around the world in subjects ranging from computer science to language arts, social studies and physics.

Recent literature reports growing number of new case studies where wiki is used and tested. In [9], Mader presents ten case studies on using Wiki in education, collected from different teachers. However, not all case studies give positive results. There are also case studies with negative results such as experiment given in [28] where the author of paper presents failed experiment. However, most case studies report positive results on using Wiki in education. Those studies include different processes of using Wiki in teaching including preparation of new Wiki based course as well as using Wikis for assessment of results at the end of course. In general, publications on Wikis in education range from descriptive efforts to characterize Wiki learning activities and cultures, prescriptive efforts to establish guidelines for implementing Wiki learning activities, and a few design reports that document technological innovations to support classroom use [8].

2.3 Peer-Assessment

Peer assessment has been used in many different fields, such as writing composition, business, science, electrical engineering, medicine, information and social science [29]. In [30], Topping states that peer-assessment represents a reliable and valid method for assessment and teaching. Students learning is positively influenced by assessment [31], since they are informed about their weaknesses and strengths in this way. Active involvement of students in the assessment process supports students learning. Peer assessment has been implemented for many years in curricula in numerous ways [32]. The most cited definition of peer-assessment is given in [30]: "Peer assessment is an educational arrangement where students judge a peer's performance quantitatively, by providing a peer with scores or grades, and/or qualitatively, by providing the peer with written or oral feedback". Peer-assessment helps students to develop certain skills in communication, self-evaluation, observation and self-criticism [31].

Topping specifies typology of peer assessment between students in higher education [30], where peer-assessment types are identified as: 1) grading a peer's research report, 2) providing qualitative feedback on a students from the same group presentation, and 3) evaluating a student trainee's professional task performance [33]. Traditionally, peer-assessment is used in small classes and was conducted using pen-and-paper methods [34]. Nowadays, web-based tools for peer-assessment are under development and their popularity is increasing, particularly in education domain (web-based peer-assessment). Luxton-Reilly gives review and analyzes of tools that support peer-assessment, revealing the common features and significant differences, as well as suggestions on future directions for research on peer assessment tools [34].

2.4 Concept of Trust

Trust is an integral component in many kinds of human interaction, allowing people to act under uncertainty and in the presence of the risk of negative consequences. It is a widely used term whose definition differs among researchers and application areas. Trust plays a role across many areas, including sociology, psychology, economics, political science, history, philosophy, and computer science. Researchers from different areas addressed the problem of the trust concept definition [35]. Common agreement is that the most promising approach assumes collaboration between Social Sciences and Computer Science. Social societies are strongly based on the trust between people. On the other side, computing and communications systems demand translation of particular aspects of sociological thinking to take advantage of concepts such as reputation and trust to generate systems offering trustworthy and secure information services and networking applications. In communications systems, reputation can act as control access mechanism providing certain level of trust. [35].

In computer science, trust is not a new research topic. Concept of trust exists in diverse fields such as security and access control in computer networks, reliability in distributed systems, game theory and agent systems, and policies for decision making under uncertainty [36]. Trust judgment in such systems are made by humans based on their prior knowledge about sources perceived reputation, or past personal experience about its quality relative to other alternative sources they may consider. Trust related research fields can be organized in the four major areas [36]: 1) policy-based trust, 2) reputation-based trust, 3) general models of trust, and 4) trust in information resources. For the purposes of this research, we focus on trust in web-based social networks where trust of an online user is mapped to a student/peer.

2.5 Research Experience and Problem Identification

This section gives brief overview on our research since 2003, when we started to include social computing approach in teaching and learning process. Then, we developed wiki into a social computing tool and involved it to support novel way of teaching and learning that we adopted specifically to answer practical needs.

The first wiki we used was intended as a tool for teaching and learning process support, and represent web site of the course where students were offered to actively participate in content creating and modifications on a collaborative way. Students started to actively contribute to the web site and use it as a learning support tool. The first experience resulted in positive results in wiki adoption among students and we continued using it until nowadays for small groups and for individual coursework development such as home works (some topics on which students work outside classrooms and labs work). The experience included other positive effects also: improved student-teacher communication, students were more active on the projects, management of the course was more natural and students liked this approach more. In last couple of years, we noted increasing number of users as well as increasing number of wiki pages, since some users started to actively contribute to wiki. We maintained tight reflection loop by following students needs and responding with new mechanisms and social indicators that got included into the learning wiki system.

One of the main challenges that we face in our University teaching practice is how to adopt interactive course materials development imposed by continuous change within Information Systems knowledge domain. At the beginning, we involve students for preparing lecture materials under tight continuous teachers supervision. Then, wiki starts to be used as a learning repository and collaborative space. As a next step, we started to apply active learning methodology in combination with wiki system as teaching and learning support. Finally, students became very active in using wiki. As a consequence, the problem of objective students assessment, that would include all types of learning activities, appeared.

3 Trust-Based Peer Assessment

In this paper we consider peer assessment as being generally about students assigning each other marks. A mark may take its value from a finite discrete set of values. Without loss of generality, we adopt the set of mark values as the *mark values vector* $MV = [5, 6, 7, 8, 9, 10]$[1]. Students are instructed to anonymously assign marks to other students of their group such that assigned mark represents the students best guess about level of knowledge that his/her colleague possess relevant to the body of knowledge covered by the course. Mark 5 means failing, mark 6 means "pass mark", while mark 10 is the highest possible mark. Students may assign marks to arbitrary number of peers.

We model students peer assessment as a three-stage decision making process[2]. In the first stage, each individual student makes decision about set of peers she would take into consideration for the assessment. Then, she decides about

[1] The values actually correspond to marks students traditionally get on final exams at our University. The general assumption is that it is a finite discrete set of values.

[2] Note that the stages in the process are not necessarily strictly separated. Instead, it is possible for decisions from different stages to be shuffled depending on the particu-lar algorithm.

assignment of specific mark to each of the selected peers. Finally, social decision is made by propagating the assignments at the system level. We model factors that influence students decision making process as shown in Figure 1, where S represents the level of students assessment skills, PKWs is previous knowledge of student S, NKWs is new knowledge that student S acquired within course, E is experience in assessment, and UNK represents unknown variable[3]. We introduce the unknown variable to represent outside factors such as: some kind of bargaining (for example, if student A assess student B with better mark, then student A would expect better assessment from student B in return) and personal preference/sympathy (acquaintance) of student that is being assessed.

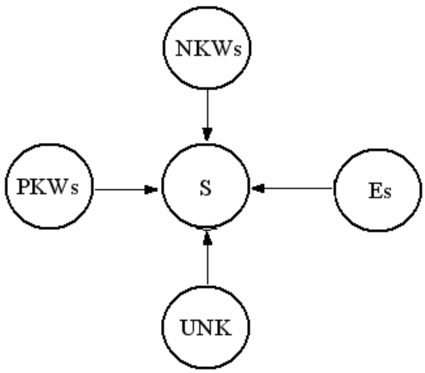


Fig. 1. Factors that influence the students peer-assessment decision making process

Based on the identified factors that influence students assessments, we can now assume that different categories of the students decision making capacity for assessments exist. Then we interpret the assessment capacity as a trust level. Thus, we address a trust network S appropriate for our further representation. In general, a trust network represents a graph where nodes represent agents and edges represent trust relations [37], [38]. A trust relation from agent u to agent v indicates how much trust u places in v. Thus, an edge in a trust network is associated with a trust value as its weight. A trust value can be a single scalar, a Beta distribution, or can follow another representation, which depends on the adopted trust model [39]. Here we will only consider a trust value as a single scalar. In our case, we will consider trust network as assessment network, an agent as a student and trust relations as assessment relations.

Let as introduce the peer assessment network as a directed weighted graph whose nodes represent agents (or peers), edges represent assessment relations, and weights represent assessment values or marks.

[3] Note that the model is not instructional in any sense. It is used to help us reason about the problem. More detailed discussion about different aspects of the model is out of scope of the paper.

Definition 1. *Peer Assessment Network (PAN) is a directed weighted graph* PAN(V,E)*, where V is a finite set of agents $V = \{v_1, \ldots, v_n\}$, and E is a set of edges representing assessment relations $\{e_{ij}|e_{ij} = (v_i, v_j),\ \ v_i, v_j \in V, |E| \leq N^2$.*

Figure 2 shows example of an assessment network, where an edge from node s to node t with assessed value w means s assess t to the extent of w.

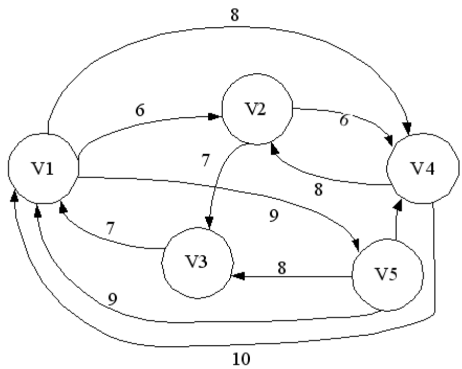


Fig. 2. Example of peer assessment network where edge weights represent assessment values

Now we consider the general case where each of the edges in PAN is associated a weight representing assessment value. Instead of using a binary adjacency matrix, we define the adjacency matrix with assessment. The entries in the adjacency matrix with assessment represent the assessment values associated to the corresponding edges.

Definition 2. *Assessment adjacency matrix A of a peer assessment network* PAN(V,E) *of n students is an $n \times n$ matrix where the entry a_{ij} equals value that v_i assessed v_j with and 0 if v_i didnt assess v_j.*

Assessment adjacency matrix for PAN given in Figure 2, is:

$$A = \begin{bmatrix} 0 & 6 & 0 & 8 & 9 \\ 0 & 0 & 7 & 6 & 0 \\ 7 & 0 & 0 & 0 & 0 \\ 10 & 8 & 0 & 0 & 0 \\ 9 & 0 & 8 & 6 & 0 \end{bmatrix} \tag{1}$$

Now, we introduce binary trust levels: good and bad. The idea here is that we want to trust only good nodes, and take their assessments in consideration for calculations, while assessments of bad nodes are discarded. The model of accepted assignments is represented in Figure 3.

Bad nodes are the nodes that have no incident edges coming from good nodes. The states of the nodes can be calculated from adjacency matrix of assessment network, where adjacency matrix is defined as follows.

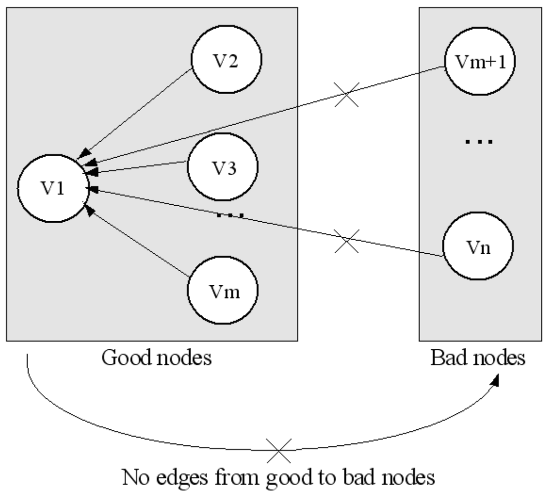


Fig. 3. Model of accepted assignments based on binary trust levels

Definition 3. *Adjacency matrix B of a peer assessment network* PAN(V,E), *with the node set $V = \{v_1, \ldots, v_N\}$ and edge set E, of n students is an $n \times n$ matrix where entry b_{ij} is given as:*

$$b_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Thus, our bad node will be node v_j associated with column j whose all entries are equal to 0. All nodes identified as bad nodes are associated with the same trust value equal to 0.

Now, we approximate trust of an individual agent, based on the trust model given in Figure 1, by using the trust vector T. The idea is to provide means to refine trust level of a good node into more levels.

Definition 4. *Trust vector T of a peer assessment network PAN(V,E) of n students is an n vector where the entry t_i is given as*

$$b_{ij} = \begin{cases} tw_0, & \text{bad node or } \ av_i < ll_1 \\ tw_1, & ll_1 \leq av_i < ul_1 \\ tw_2, & ll_2 \leq av_i < ul_2 \\ tw_3, & ll_3 \leq av_i < ul_3 \\ tw_4, & ll_4 \leq av_i < ul_4 \\ tw_5, & ll_5 \leq av_i \leq ul_5 \end{cases} \tag{3}$$

where $ll_j(j = 1, \ldots, |MV| - 1)$ is lower limit discretization vector $ul_j(j = 1, \ldots, |MV - 1|)$, is upper limit discretization vector $tw_k(k = 0, \ldots, |MV| - 1)$, is trust weight vector, and $av_i(i = 0, \ldots, n)$ represents average value of values from i-th column of assessment adjacent matrix A of PAN(V,E).

The lower limit discretization vector that we use in our case is $LL = [6, 6.5, 7.5, 8.5, 9.5]$, upper limit discretization vector $UL = [6.5, 7.5, 8.5, 9.5, 10]$, and trust weight vector $TW = [0, 1.0, 1.2, 1.5, 2.0, 2.5]$. Note that values in these vectors are application specific and may be selected appropriately without any effect to the rest of the discussion.

Definition 5. *Final grades vector FG of a* PAN(V,E) *of n students is a n vector where the entry fg_i is calculated as:*

$$fg_i = \frac{\sum_{j=1}^{N} a_{ij} * t_j}{\sum_{j=1}^{N} t_J}, \quad i = 1, \ldots, n \tag{4}$$

where a_{ij} is entry of assessment adjacency matrix A while t_j is entry of the trust vector T for the given PAN(V,E) of n students.

Now, we are able to formalize our trust-based peer assessment grading approach as follows: Given a peer assessment network PAN(V,E), calculate adjacency matrix with assessment A. Then classify all nodes in the network as bad or good based on the corresponding adjacency matrix B. Only the nodes that are classified as good are considered for further calculations. Next, define trust vector T of the assessment network, and finally calculate vector FG by using the equation (4).

4 Experiment Setup and Results

In the following, we describe the experiment setup and further discuss obtained results of used service for students self-assessment. We start from the following hypothesis:

> *Hypothesis H1. Traditional grading scores method applied by teachers and trust-based peer-assessment method provide grading scores with no significant statistical difference.*

4.1 Software Architecture of the Web Service for Peer-Assessment

This section gives the experimental setup software service architecture, as shown in Figure 4. The service has three main modules:

- *User management module:* This module handles authorization and authority levels of the systems users. There are three roles: administrator, teacher and student role.
- *Assessments management module:* Provides peer-assessment of students, where each student can assess the other students and put notes. When student is logged in, the table of only students that logged student has assess to is shown. Logged student can change previous given marks any time until peer-assessment procedure is closed.

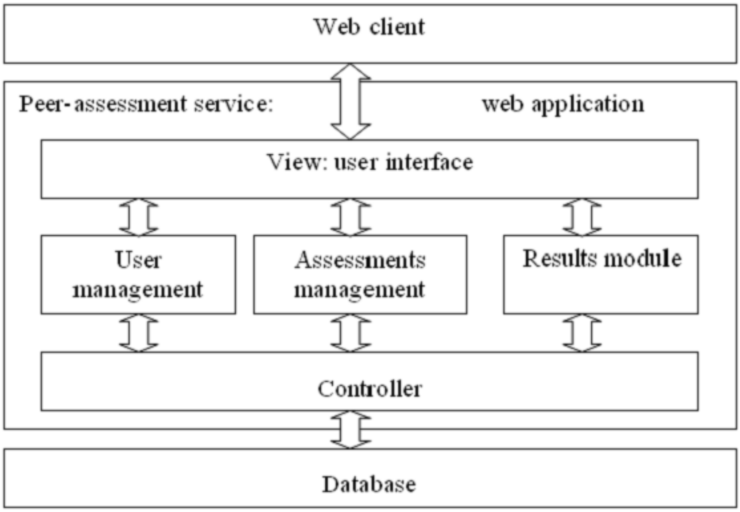


Fig. 4. Architecture of trust-based peer-assessment service

- *Results module:* Provides peer-assessment results and current table with calculated marks. This module computes the final marks based on peer-assessment approach presented in previous section. Only teachers can preview current state of peer-assessment.

Main modules are connected with user interface component and controller that make further connections to database.

4.2 Experiment Setup

Participants in this experiment are students from the Faculty of Electronic Engineering, University of Nis, from the group of five Information Systems related courses. Average number of students attending a course from this group of courses was about 150 per semester per course in the last three years. For this case study, we engaged students attending Information Systems course. Students were organized in three groups, each group with approximately 30 students. Course is divided on traditional classroom and lab classes. Teaching process is staged in lecture weeks and each week students should present what they have been learned.

We stimulate students to create wiki pages as they learn with no additional limitations. All existing content of student lab exercises, accumulated by former generations, can be used by any current student as a literature. All students are welcome to comment each others work. In this way, students have insight into each others work, so they can competently make peer assessment.

At the beginning of the experiment, a short introduction was given to students, including overall teaching philosophy and importance of their participation in the wiki learning system. Then teachers created groups of students. Students from the same group attended classes together. Teachers then published individual projects for students. Also, there were optional additional exercises for each of the students, where they can earn extra credit points. Students within same group worked collaboratively and presented weekly reports. Reports are available online and each student from the group can read and revise what other students have done. Teaching assistant selected few reports as candidates for the best report each week, and published list of the reports for voting. At the end of semester, the authors of the best reports would earn extra credit points. Students presented their final work in more details in last week of the course. At that time, teachers introduced trust-based peer-assessment service and gave a deadline for completing the process.

4.3 Results

Table 1 gives statistics data on number of students active in the peer-assessment process for each of the three groups.

Table 1. Number of studens active in the peer-assesment process

Group	Total number of students	Number of students active in the peer-assessment	Average number of assessments received by a student
pp1	30	18	8.76
pp2	30	16	9.22
pp3	25	14	8.12

Teaching assistant has assessed students over the semester by using traditional grading scores method and at the end of semester calculated the total students grading scores. In order to compare mean grade scores of teachers and grade scores obtained by trust-based peer-assessment, we use the paired samples t-test that compares the means of two variables [40]. This test computes the difference between the two variables for each case. We compare results obtained with two different methods on the same sample in order to confirm whether both methods provide similar results or not.

We used SPSS v17.0 statistical tool to obtain results of analysis. We apply paired sample t-test on each of the groups (pp1, pp2 and pp2), since the peer-assessment was used for each group separately. Results of paired t-test are shown in Table 2. For each students group, we paired data obtained via traditional assessment method (labelled as TRAD) and data obtained by using service for trust based peer-assessment (labelled as PAT). Table 2 shows mean of the differences, standard deviation of the differences, standard error of the mean, the confidence interval for the difference, and the obtained value for t and df (degrees of freedom). The confidence level of a confidence interval is an assessment of how confident we are that the true mean is within the interval. The precision of the interval is given by its width (the difference between the upper and lower endpoint). If df is lower, t-value should be higher to reach significance. The2-tailed Significance (known as p-value) which is stated as a probability is shown

Table 2. Results of two-tailed paired samples t-test applied on three separated students groups: pp1, pp2 and pp3

	Paired Differences							
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1 pp1TRAD - pp1PAT	-.40741	1.18514	.22808	-.87623	.06142	-1.786	26	.086
Pair 2 pp2TRAD - pp2PAT	-.21429	1.52406	.28802	-.80525	.37668	-.744	27	.463
Pair 3 pp3TRAD - pp3PAT	-.25000	1.25974	.25714	-.78194	.28194	-.972	23	.341

in the last column of the table. Probabilities less than $\alpha = 0.05$ (where α is a significance level of a test) indicate that the hypothesis should be rejected [40].

Based on the results of the paired t-test given in Table 2 we have:

- A paired samples t-test not failed to reveal a statistically reliable difference between the mean number of pp1TRAD (M=8.0741,S=0.30628) and pp1PAT (M=8.4815, S=0.20234) that the students have t(27)=-1.786, p=0.086.
- A paired samples t-test not failed to reveal a statistically reliable difference between the mean number of pp2TRAD (M=7.8571,S=0.32819) and pp2PAT (M=8.0714, S=0.20528) that the students have t(28)=-0.744, p=0.463.
- A paired samples t-test not failed to reveal a statistically reliable difference between the mean number of pp3TRAD (M=8.0833,S=0.29437) and pp3PAT (M=8.3333,S=0.13003) that the students have t(24)=-0.972, p=0.341,

where M is mean value and S is standard deviation.

The results of the three tests has shown that p-value is grater than 0.05, which indicate that hypothesis H1 can be accepted: There are no significant differences between results obtained by using traditional grading scores method and results obtained by using service for trust-based grading scores peer-assessment.

5 Conclusion

This paper gives short overview on teachers experience of using wiki as social computing platform in teaching and learning process in last few years from the group of Information Systems related courses at Faculty of Electronic Engineering. Involvement of such tool in teaching and learning process has revealed new problems in students assessments. Our approach to addressing the problems as based on peer-assessment principles. We used grading peer-assessment method and involved trust in this method with a goal of reliable assessment process. A new approach to students assessment, based on grading scores method with trust-based decision making process, is proposed.

No statistically significant difference was detected between results obtained using traditional grading scores method and results obtained using method proposed in the paper. Thus, proposed trust-based grading scores peer-assessment approach can be adopted as reliable in the students assessment process.

Results and ideas presented in this paper open several interesting research questions that we plan to address in our future work. First, privacy and security of educational record that includes student scores based on peer-assessment should be investigated in more details. Next, larger and less compact groups of student should be examined with a goal to find out how the proposed assessment method performs in large heterogeneous groups. Particularly interesting would be to provide scientific evidence for our implicit assumption that the social knowledge within the group under test is the main source of power that makes the proposed algorithm successful. It may be done by testing the algorithm on a group of people with weak or none social connections. Robustness with respect to attempt of manipulation (for example to increase the average grades) should be investigated in more details. Also, we plan to test the method to application domains other then teaching such as finding expertise level distribution in a given community of researchers.

Acknowledgement

We would like to thank anonymous referees for their valuable comments and their constructive suggestions that help us improve the final version of the paper.

References

1. Guth, S.: Wikis in education: is public better? In: WikiSym 2007: Proceedings of the 2007 International Symposium on Wikis, pp. 61–68. ACM, New York (2007)
2. Dalsgaard, C.: Social networking sites: Transparency in online education (2008), `http://eunis.dk/papers/p41.pdf`
3. Messinger, P.R., Stroulia, E., Lyons, K., Bone, M., Niu, R.H., Smirnov, K., Perelgut, S.: Virtual worlds — past, present, and future: New directions in social computing. Decision Support Systems 47(3), 204–228 (2009)
4. Nov, O.: ch. 1. Information Sharing and Social Computing: Why, What, and Where? Advances in Computers 76, 1–18 (2009)
5. Kwai, R., Wagner, C.: Weblogging: A study of social computing and its impact on organizations. Decision Support Systems 45(2), 242–250 (2008)
6. Parameswaran, M., Whinston, A.B.: Research Issues in Social Computing. Journal of the Association for Information Systems 8(6), 336–350 (2007)
7. Parker, K.R., Chao, J.T.: Wiki as a teaching tool. Interdisciplinary. Journal of Knowledge and Learning Objects 3 (2007)
8. Forte, A., Bruckman, A.: Constructing text: Wiki as a toolkit for (collaborative?) learning. In: WikiSym 2007: Proceedings of the 2007 international symposium on Wikis, pp. 31–42. ACM, New York (2007), `http://dx.doi.org/10.1145/1296951.1296955`

9. Mader, S.: Using Wiki in education. Wiley, Chichester (2006), http://www.amazon.com/Using-Wiki-Education-Stewart-Mader/dp/B00262SOWQ/ref=sr_1_1?ie=UTF8\&s=books\&qid=1249307776\&sr=1-1
10. Motteram, G., Sharma, P.: Blending Learning in a Web 2.0 World. International Journal of Emerging Technologies & Society 7(2), 83–96 (2009), http://www.swinburne.edu.au/hosting/ijets/journal/V7N2/pdf/Article2-Motteram\&Sharma.pdf
11. Cobb, P.: Where is the mind? A coordination of sociocultural and cognitive constructivist perspectives. In: Fosnot, C.T. (ed.) Constructivism: Theory, perspectives, and practice, pp. 34–52. Teachers College Press, New York (1994)
12. Cubric, M.: Using Wikis for Summative and Formative Assessment. In: From Assessment Design for Learner Responsibility, pp. 29–31 (2007)
13. Wilson, S., Sharples, P., Popat, K., Griffiths, D.: Moodle Wave: Reinventing the VLE using Widget technologies. In: Wild, F., Kalz, M., Palmér, M., Müller, D. (eds.) Proceedings of 2nd Workshop Mash-Up Personal Learning Environments (MUPPLE 2009). Workshop in conjunction with 4th European Conference on Technology Enhanced Learning (EC-TEL 2009), Synergy of Disciplines, CEUR workshop proceedings, Nice, France, September 29, pp. 47–58 (2009), http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-506
14. AL-Smadi, M., Guetl, C., Kappe, F.: Peer Assessment System for Modern Learning Settings: Towards a Flexible E-Assessment System. iJET, Special Issue 2: "MIPRO 2009" 5, 5–11 (2009)
15. Kollar, I., Fischer, F.: Peer assessment as collaborative learning: A cognitive perspective. Learning and Instruction 20(4), 344–348 (2010)
16. Elliott, N., Higgins, A.: Self and peer assessment – does it make a difference to student group work? Nurse Education in Practice 5(1), 40–48 (2005)
17. Topping, K.J.: Methodological quandaries in studying process and outcomes in peer assessment. Learning and Instruction 20(4), 339–343 (2010)
18. Chang, T., Chen, Y.: Cooperative learning in E-learning: A peer assessment of student-centered using consistent fuzzy preference. Expert Systems with Applications 36(4), 8342–8349 (2009)
19. Gielen, S., Peeters, E., Dochy, F., Onghena, P., Struyven, K.: Improving the effectiveness of peer feedback for learning. Learning and Instruction 20(4), 304–315 (2010)
20. Reily, K., Finnerty, P.L., Terveen, L.: Two peers are better than one: aggregating peer reviews for computing assignments is surprisingly accurate. In: Proceedings of the ACM 2009 International Conference on Supporting Group Work, GROUP 2009, Sanibel Island, Florida, USA, May 10-13, pp. 115–124. ACM, New York (2009), doi: http://doi.acm.org/10.1145/1531674.1531692
21. Sondergaard, H.: Learning from and with peers: the different roles of student peer reviewing. In: Proceedings of the 14th Annual ACM SIGCSE Conference on innovation and Technology in Computer Science Education, ITiCSE 2009, Paris, France, July 06-09, pp. 31–35. ACM, New York (2009), doi: http://doi.acm.org/10.1145/1562877.1562893
22. Herbert, N.: Quantitative peer assessment: can students be objective? In: Proceedings of the Ninth Australasian Conference on Computing Education, vol. 60, pp. 63–71 (2007)
23. Falchikov, N., Goldfinch, J.: Student peer assessment in higher education: a meta-analysis comparing peer and teacher marks. Review of Educational Research 70, 287–322 (2000)

24. Tillema, H.H.: Assessment for learning to teach: appraisal of practice teaching lessons by mentors, supervisors, and student teachers. Journal of Teacher Education 60, 155–167 (2009)
25. Lin, C., Standing, C., Liu, Y.: A model to develop effective virtual teams. Decision Support Systems 45(4), 1031–1045 (2008)
26. Luo, S., Xia, H., Yoshida, T., Wang, Z.: Toward collective intelligence of online communities: A primitive conceptual model. Journal of Systems Science and Systems Engineering 18(2), 203–221 (2009)
27. Wang, F.-Y., Carley, K.M., Zeng, D., Mao, W.: Social Computing: From Social Informatics to Social Intelligence. IEEE Intelligent Systems 22(2), 79–83 (2007), doi:10.1109/MIS.2007.41
28. Cole, M.: Using Wiki technology to support student engagement: Lessons from the trenches. Computers and Education (in press, corrected Proof)
29. Tseng, S., Tsai, C.: On-line peer assessment and the role of the peer feedback: A study of high school computer course. Computers and Education 49(4), 1161–1174 (2007)
30. Topping, K.: Peer Assessment Between Students in Colleges and Universities. Review of Educational Research 68(3), 249–276 (1998)
31. Gennip, N., Segers, M., Tillema, H.: Peer assessment as a collaborative learning activity: The role of interpersonal variables and conceptions. Learning and Instruction 20, 280–290 (2010)
32. Sluijsmans, D.: Establishing learning effects with integrated peer assessment tasks (2002), Article published December 13 on the member's area of the ILT-website, `http://www.palatine.ac.uk/files/930.pdf`
33. Zundert, M., Sluijsmans, D., Merriënboer, J.: Effective peer assessment processes: Research findings and future directions. Learning and Instruction 20(4), 270–279 (in press, 2010)
34. Luxton-Reilly, A.: A systematic review of tools that support peer assessment. Computer Science Education 19(4), 209–232 (2009)
35. De Paoli, S., Gangadharan, G., Kerr, A., D'Andrea, V., Serrano, M., Botvich, D.: Toward Trust as Result: An Interdisciplinary Approach. In: Proceedings of ALPIS. Sprouts: Working Papers on Information Systems, vol. 10(8) (2010), `http://sprouts.aisnet.org/10-8`
36. Artz, D., Gil, Y.: A survey of trust in computer science and the Semantic Web. Journal of Web Semantics: Science, Services and Agents on the World Wide Web 5, 58–71 (2007)
37. Wierzbicki, A.: Trust and Fairness in Open, Distributed Systems. SCI, vol. 298. Springer, Heidelberg (2010)
38. Wang, Y., Singh, M.P.: Trust representation and aggregation in a distributed agent system. In: Proceedings of the 21st National Conference on Artificial Intelligence (AAAI), pp. 1425–1430. AAAI Press, Menlo Park (2006)
39. Hang, C., Singh, M.: Trust-Based Recommendation Based on Graph Similarity. In: Proceedings of the 13th AAMAS Workshop on Trust in Agent Societies, Trust (2010)
40. Wielkiewicz, R.: SPSS Online Guide. McGraw-Hill Companies, New York, `http://www.mhhe.com/socscience/psychology/runyon/spss/ttest.html` (accessed 15/6/2010)

Exponential Ranking: Taking into Account Negative Links*

Vincent A. Traag[1], Yurii E. Nesterov[2], and Paul Van Dooren[1]

[1] Department of Mathematical Engineering
4 Avenue G. Lemaître, 1348 Louvain-la-Neuve, Belgium
[2] CORE, 34 Voie du Roman Pays, 1348, Louvain-la-Neuve, Belgium
vincent.traag@uclouvain.be

Abstract. Networks have attracted a great deal of attention the last decade, and play an important role in various scientific disciplines. Ranking nodes in such networks, based on for example PageRank or eigenvector centrality, remains a hot topic. Not only does this have applications in ranking web pages, it also allows peer-to-peer systems to have effective notions of trust and reputation and enables analyses of various (social) networks. Negative links however, confer distrust or dislike as opposed to positive links, and are usually not taken into account. In this paper we propose a ranking method we call *exponential ranking*, which allows for negative links in the network. We show convergence of the method, and demonstrate that it takes into account negative links effectively.

1 Introduction

The ranking of nodes, or assigning some 'importance' or 'trust' scores to nodes, has attracted a great deal of attention when networks are being studied. Already in the 1970s, various researchers from the social sciences have introduced concepts such as betweenness [1], closeness [2] and eigenvector centrality [3,4] to measure how central or important a node in the network was. For example, centrality-like measures are shown to have an influence on spreading processes on networks, such as failing cascades [5], or the infection process of sexually transmitted diseases [6,7]. Furthermore, it helps to identify different roles nodes might play in a network [8].

In the 1990s several alternative ranking measures were added, notably Kleinbergs HITS-algorithm [9], and Googles PageRank [10]. When filesharing and especially peer-to-peer applications grew, these measures, and variants thereof, became popular to keep 'good' peers in the sharing network, and exclude 'bad' peers [11,12]. Reputation and trust also plays a vital role in online markets such as eBay [13].

* This research was funded by "Actions de recherche concertes "Large Graphs and Networks" of the "Communauté Française de Belgique" and from the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office.

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 192–202, 2010.

Negative links however, are usually not taken into account by these ranking measures, or worse, they break down when negative entries appear as weights of the links. However, the signs of links (positive or negative) should not be ignored, since they may bear important consequences for the structure of the network, not in the least for the ranking of nodes. Proposals have been made to include such semantic information in hyperlinks on the World Wide Web [14]. Negative links are also present in various other settings such as reputation networks [15], sharing networks [11], social networks [16] and international networks [17], and play a key, if not vital, role in these networks. Studying how negative links influence the 'importance' of nodes may help the understanding of such systems, and such a concept of 'importance' might facilitate the analyses of such networks again.

Recently there has been more attention to negative links in ranking measures, for example PageTrust [18]. The difference between PageTrust and PageRank is that in the random walk in PageTrust nodes that are negatively pointed to during the random walk are blacklisted, and are visited less often, thereby lowering their PageTrust score. Another suggestion was to first calculate some ranking using only the positive links (e.g. using PageRank), and then apply one step of distrust, so that the negative links are taken into account [19,14].

It was also suggested to introduce a propagation of distrust [19], implying that if i distrusts j, and j distrusts k, then i should trust k (the adagium that the enemy of my enemy is my friend). The authors noted that this could lead to situations in which a node is its own enemy (if one considers a cycle of three negative links). This phenomenon is studied in the social sciences under the denominator of 'social balance theory' [20,21]. A network is considered as balanced, if all triads (a cycle of three nodes) in the network are either completely positive, or have only one positive link, and more recently some models have tried to capture dynamics based on social balance [22]. Already in 1953 it was shown that a network is balanced in this sense if and only if it divides neatly in two clusters with negative links appearing only between the two clusters [20]. Later, this idea was extended to include the possibility to cluster nodes in more than one cluster, by only demanding there is not exactly one negative link in any cycle [21], although for practical clustering in networks with negative links other methods have been devised [23]. The triad with only negative links is found to appear more often than expected by social balance theory [16,17], so that they (potentially) divide in more than two clusters.

In this short paper we introduce a ranking measure based on discrete choice theory, that can also be used when negative links are present in the network. The goal is to infer some global ranking of nodes, based on a particular given network (with possibly some negative links present). We do so in terms of reputation and trust, but the application of the measure need not be restricted to the domain of trust management. It might find also applications in collaborative or iterative filtering, where items such as movies or products need to be recommended[1] [24],

[1] Often recommendation is personalized based on existing preferences or purchases of movies or products. We do not currently consider such a personalization in this short paper, but there seem to be some possibilities for doing so.

or somewhat related, predict the sign of links [25]. However, such a measure might also be of interest for characterizing nodes in various networks, such as the international network of conflict and alliances [17], or in an online social network [16].

Given the considerations of social balance provided above—that indeed the enemy of your enemy need not be your friend—it would be undesirable to assume so in any ranking scheme. That is, if a node were to have a negative reputation, his links should not be distrusted, only trusted less. In other words, we should not assume a node with a negative reputation is not trustworthy (if he points negatively towards someone, we should not interpret it as positive, and vice versa), we should only trust his judgements less. This will actually follow from the derivation of the measure based on a discrete choice argument, which we will present in the following section. Most of the existing algorithms dealing with negative links do not apply distrust in such a recursive manner, thereby limiting their effect. Furthermore, none of the algorithms can actually deal with negative reputations, while this negativity can actually provide additional insight. For example, a negative reputation would signal that such a node should be blocked from the network.

The reputation of nodes is based on judgments by other nodes, which is detailed in Sec. 3. Convergence and uniqueness of the proposed measure is proven in Sec. 4. We give some conclusion and indications of further research in the final section.

2 Discrete Choice

Let $G = (V, E)$ be a directed graph with $n = |V|$ nodes and $m = |E|$ edges. Each edge (ij) has an associated weight w_{ij} which can possibly be negative. By A we denote the $n \times n$ adjacency matrix associated to the graph, such that $A_{ij} = w_{ij}$ if there is an (ij) edge and zero otherwise. Furthermore, let k_i be some reputation of node i (we will make this explicit in the next section). We consider the links to indicate a certain trust: if node i points positively (negatively) to node j, this indicates that i trusts (distrusts) j. The goal is to infer some global trust values from the local trust links.

Suppose we are asked which node to trust, if we were to choose one. We assume that a higher reputation indicates some degree of trust, so we should preferably choose nodes which have a high reputation k_i. However, there might be some errors in choosing the one with the highest reputation. This is where the framework of discrete choice theory comes in.

The usual background for discrete choice theory is the following [26]. Suppose there are n different choices (in our case, nodes), which have a different associated utility u_i. We observe the utility o_i and have some error term ϵ_i such that

$$u_i = o_i + \epsilon_i. \tag{1}$$

We would like to choose the object with the maximum utility. However, since we only observe o_i, it is uncertain which item actually has the maximum real utility. So, the question becomes: what is the probability we will select a certain object? That is, what is the probability that $u_i \geq u_j$ for all $i \neq j$, or

$$\Pr(u_i = \max_j u_j), \tag{2}$$

depending on the observed utility o_i and the error term ϵ_i. In our case, we equate the observed utility o_i with some reputation k_i. We assume the real reputation is then $u_i = k_i + \epsilon_i$, where ϵ_i is the error made in observing the reputation.

The probability of choosing the node with the highest reputation depends on the distribution of the error term ϵ_i. Using the following assumption for the error term, we arrive at the well known multinomial logit model [26]. Suppose the ϵ_i are i.i.d. double exponentially distributed[2] according to

$$\Pr(\epsilon_i \leq x) = \exp - \left[\exp - \left(\frac{x}{\mu} + \gamma\right)\right], \tag{3}$$

where $\gamma \approx 0.5772$ is Euler's constant. The mean of (3) equals zero, and the variance equals $1/6\pi^2\mu^2$. With this error distribution it can be proven [26] that the probability node i has the highest real reputation becomes

$$p_i = \frac{\exp \frac{k_i}{\mu}}{\sum_j \exp \frac{k_j}{\mu}}. \tag{4}$$

The probability a node i has the highest reputation, increases with higher reputation k_i, depending on the amount of noise characterized by μ. There are two extreme scenarios depending on μ. If $\mu \to \infty$ the variance goes to infinity, and the contribution of the observed reputation in $u_i = k_i + \epsilon_i$ becomes negligibly small. In that case, the probability a node has the highest real reputation becomes uniform, or $p_i = 1/n$. In the other extreme, $\mu \to 0$, there is essentially no error, and we will always be correct in choosing nodes with a maximum k_i. That is, if there is a set of nodes M with $k_i = \max_j k_j$ for $i \in M$, then $p_i = 1/|M|$ for $i \in M$, and zero otherwise.

The probabilities p shows how much we should trust nodes. Nodes with a higher reputation are more trustworthy than nodes with a lower reputation. The difference in trust becomes more pronounced with decreasing μ, up to the point where we only trust nodes with the highest reputation. We shall call these probabilities the trust probabilities.

3 Reputation and Judges

The trust probabilities defined in the previous section depend on the reputation k_i, which we will define now. We will ask a certain node j to provide the reputation values of the other nodes. That is, we ask node j to be the judge of his peers. Since we consider A_{ji} to be the trust placed by node j in node i, we

[2] This distribution is also known as the Gumbel distribution.

Table 1. Overview of variables used in this paper

A	Adjacency matrix of given network
k_i	Reputation of node i
ϵ_i	Error term in reputation
μ	Parameter influencing variance of error
p_i	Trust probability of node i
k_i^*	Final (fixed point) reputation
p_i^*	Final (fixed point) trust

will assume that if node j is the judge, he would simply say that $k_i = A_{ji}$. The general idea is that the probability to be a judge depends on the reputation, which then influences that probability again.

To choose a judge, we will again consider a discrete choice argument. We would like to select a judge j for which $k_j + \epsilon_j$ is maximal, where ϵ_j is again distributed according to (3). Obviously, the probability to select a judge will then be the same as in (4).

Using those probabilities p_i, we select a judge at random, and let him give his opinion on the reputation of his peers. We thus allow trustworthy nodes a higher probability to judge their peers. The expected reputation can then be written as

$$k_i = \sum_j A_{ji} p_j, \tag{5}$$

or in matrix notation, the column vector k can be expressed as

$$k = A^{\mathsf{T}} p, \tag{6}$$

where A^{T} is the transpose of A and p is a column probability vector (i.e. $\|p\|_1 = 1$ and $p_i \geq 0$). If we plug this formulation of the reputation into (4) we obtain a recursive formulation of trust probabilities

$$p(t+1) = \frac{\exp \frac{1}{\mu} A^{\mathsf{T}} p(t)}{\| \exp \frac{1}{\mu} A^{\mathsf{T}} p(t) \|_1}, \tag{7}$$

for some initial condition $p(0)$, with $\exp(\cdot)$ the elementwise exponential. We will prove in the next section that this iteration actually converges to a unique fixed point p^*, i.e. independent of the initial conditions, for some range of values for μ. The final values of the trust probabilities can thus be defined as the limiting vector $p^* = \lim_{t\to\infty} p(t)$ or, equivalently, the fixed point p^* for which

$$p^* = \frac{\exp \frac{1}{\mu} A^{\mathsf{T}} p^*}{\| \exp \frac{1}{\mu} A^{\mathsf{T}} p^* \|_1}, \tag{8}$$

and the final reputation values as

$$k^* = A^{\mathsf{T}} p^*. \tag{9}$$

Notice that these reputation values are also a fixed point of the equation

$$k^* = A^\mathsf{T} \frac{\exp \frac{1}{\mu} k^*}{\| \exp \frac{1}{\mu} k^* \|_1} \tag{10}$$

and that the trust probabilities are related to the reputation values as

$$p^* = \frac{\exp \frac{1}{\mu} k^*}{\| \exp \frac{1}{\mu} k^* \|_1}. \tag{11}$$

In this sense, the trust probabilities and the reputation values can be seen as a dual formulation of each other.

Upon closer examination of (10), a certain node j might indeed get a negative reputation, but his judgements are taken less into account, they are not reversed. That is, as soon as a node has a negative reputation, we do not assume he his completely untrustworthy, and that his negative judgements should be taken positive, but only that he is less trustworthy. This means we indeed do not assume that the enemy of my enemy is my friend. A node could get a negative reputation for example if he is negatively pointed to by trustworthy nodes. This approach can be summarized in the idea that the reputation of a node depends on the reputation of the nodes pointing to him, or stated differently, a node is only as trustworthy as the nodes that trust him. Notice that this idea is similar to that of eigenvector centrality [4] namely that nodes are as central as the neighbours pointing to him, a recursive notion also present in PageRank [10].

Let us take a look at a small example to see what the effect is of negative links in the network as shown in Fig. 1. There is only one negative link, from a to d. The effect of the negative link becomes more penalizing when μ is decreased, as shown in Table 2(a). That has also consequences for node e, who is only pointed to by d, who receives little trust, which then also leads to little trust for e. The PageRank for these nodes (for which we did not take into account the negative link, and used a zapping factor of 0.85) are provided as comparison, which assigns nodes d and e actually higher rankings.

We will now show that indeed this limit converges (for some range of μ) and is unique, i.e. does not depend on the actual initial condition $p(0)$.

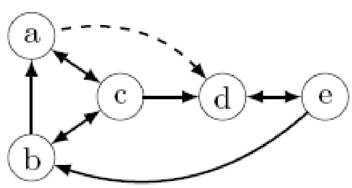


Fig. 1. A small example with one negative link (the dashed one), showing trust values for various values of μ. The weights for the positive links are +1 and for the negative link −1.

Table 2. Trust probabilities for the example network in Fig. 1. Decreasing μ also decreases the trust for d and e, but as $\mu \to 0$ we obtain cyclic behavior. We also provide the ordinary PageRank (PR) with a zapping factor of 0.85 for comparison. We removed the negative link for the calculation of the PageRank.

(a) Trust for various values of μ

	PR	$\mu=1$	$\mu=1/5$	$\mu=1/8$
a	0.183	0.223	0.384	0.424
b	0.184	0.213	0.179	0.142
c	0.263	0.223	0.384	0.424
d	0.184	0.171	0.026	0.005
e	0.186	0.171	0.026	0.005

(b) Cyclic behavior for $\mu = 0$

	$t=0$	$t=1$	$t=2$	$t=3$
a	0.20	0.50	0.3	0.5
b	0.20	–	0.3	–
c	0.20	0.50	0.3	0.5
d	0.20	–	–	–
e	0.20	–	–	–

4 Convergence and Uniqueness

More formally, let us define the map $V : S^n \to S^n$, which maps

$$V(p) = \frac{\exp \frac{1}{\mu} A^\top p}{\| \exp \frac{1}{\mu} A^\top p \|_1}, \tag{12}$$

where $S^n = \{y \in \mathbb{R}^n_+ : \|y\|_1 = 1\}$, the n-dimensional unit simplex. For the proof of convergence we rely on mixed matrix norms, or subordinate norms, which are defined as

$$\|A\|_{p,q} = \max_{\|x\|_q = 1} \|Ax\|_p. \tag{13}$$

Denoting by $\|A\|_{\max} = \max_{ij} |A_{ij}|$, we have the following useful inequality

$$\|Ax\|_\infty = \max_i \|e_i^\top Ax\| \leq \|A\|_{\max} \cdot \|x\|_1, \tag{14}$$

hence

$$\|A\|_{\infty,1} \leq \|A\|_{\max} \tag{15}$$

where e_i is the i-th coordinate vector. Let us now take a look at the Jacobian of V, which can be expressed as

$$\frac{\partial V(p)_i}{\partial p_j} = \frac{\exp(\frac{1}{\mu}A^\top p)_i \frac{1}{\mu} A_{ji}}{\sum_l \exp(\frac{1}{\mu}A^\top p)_l} - \frac{\exp(\frac{1}{\mu}A^\top p)_i \sum_l \exp(\frac{1}{\mu}A^\top p)_l \frac{1}{\mu} A_{jl}}{\left(\sum_l \exp(\frac{1}{\mu}A^\top p)_l\right)^2}. \tag{16}$$

Now let $u = \exp(\frac{1}{\mu}A^\top p)$, and $q = \|u\|_1$. Then $V(p) = u/q$, and $\frac{\partial V(p)_i}{\partial p_j}$ can be simplified to

$$\frac{\partial V(p)_i}{\partial p_j} = \frac{1}{\mu}\left(\frac{u_i}{q} A_{ji} - \frac{1}{q^2} \sum_l u_i u_l A_{jl}\right) \tag{17}$$

or in matrix notation

$$V'(p) = \frac{1}{\mu}\left(\frac{1}{q}\mathrm{diag}(u) - \frac{1}{q^2}uu^\intercal\right)A^\intercal \tag{18}$$

at which point the following lemma is useful.

Lemma 1. *Denote by $M(p)$ the matrix $M(p) = \mathrm{diag}(p) - pp^\intercal$ where $p \in S^n$, then $\|M(p)\|_{1,\infty} \leq 1$.*

Proof. Note that $\|M(p)x\|_1 = \sum_{i=1}^n p_i|x_i - p^\intercal x|$. We need to find the maximum of this function on the unit box (that is, where $\|x\|_\infty = 1$). Clearly this is attained at some vector $\sigma \in R^n$ with coordinates ± 1. Denoting by $I_+ = \{i : \sigma_i = 1\}$ the set of positive entries, and by $S_1 = \sum_{i \in I_+} p_i$ and $S_2 = 1 - S_1$. Then $p^\intercal\sigma = S_1 - S_2$, and we have

$$\begin{aligned}\|M(p)\sigma\|_1 &= \sum_{i=1}^n p_i|\sigma_i - S_1 + S_2| = \sum_{i \in I_+} p_i|1 - S_1 + S_2| + \sum_{i \notin I_+} p_i|1 + S_1 - S_2| \\ &= S_1(1 - S_1 + S_2) + S_2(1 + S_1 - S_2) = 1 - (S_1 - S_2)^2.\end{aligned}$$

Since $(S_1 - S_2)^2 \geq 0$, $\|M(p)\sigma\|_1 \leq 1$. □

This immediately leads to the following proof that the map V converges.

Theorem 1. *For $\mu > \frac{1}{2}(\max_{ij} A_{ij} - \min_{ij} A_{ij})$ the map V has a unique fixed point $p \in S^n$.*

Proof. By the Banach fixed point theorem, this map has a unique fixed point if it is contractive. That is, there should be a $c \leq 1$ such that

$$\frac{\|V(p) - V(u)\|_1}{\|p - u\|_1} < c, \tag{19}$$

for $p, u \in S^n$. That is, if $\|V'(p)\|_{1,1} < c$. Since we can write $V'(p) = \frac{1}{\mu}M(V(p))A$, using the lemma and (15) we arrive at

$$\|V'(p)\|_{1,1} = \frac{1}{\mu}\|M(V(p))A\|_{1,1} \leq \frac{1}{\mu}\|M(V(p))\|_{1,\infty}\|A\|_{\infty,1} \leq \frac{1}{\mu}\|A\|_{\max}.$$

Since adding a constant to our matrix A does not change the vector $V(p)$, we can subtract $\frac{1}{2}(\min_{ij} A_{ij} + \max_{ij} A_{ij})$, and arrive at

$$\|V'(p)\|_{1,1} \leq \frac{1}{2\mu}(\max_{ij} A_{ij} - \min_{ij} A_{ij}).$$

Hence, if

$$\mu > \frac{1}{2}(\max_{ij} A_{ij} - \min_{ij} A_{ij}),$$

the map V is contractive and by the Banach fixed point theorem, it will have a unique fixed point, and iterates will converge to that point. □

For this lower bound on μ, we can guarantee convergence of the iteration. Below this lower bound, we choose nodes with more and more certainty. As we said in Sec. 2, when $\mu \to 0$ the probabilities $p_i = 1/|M|$ for i in some set M of nodes with maximal reputation k_i. In the iteration this means only nodes with the highest reputation can become judges. Since we completely trust his judgments, to whatever node(s) he assigns the highest reputation will be the next judge. Unless everyone always agrees on the node with the highest reputation, cycles of judges pointing to the next judge will emerge.

For example, if we take $\mu \to 0$ for the example network given in Fig. 1, we cycle as follows. We start out with $p(0) = 1/n$, and the average reputation will be highest for nodes a and c, and they will be chosen as judge with probability 1/2. In the next iteration the average reputation will be 1/2 for nodes a, b and c and zero for d and e. Hence, one of the nodes a, b and c will be selected as judge, and the average reputation is 2/3 for a and c, and 1/3 for b. Now we are back where we were after the first iteration, since a and c both have the same maximal reputation, and they are chosen as judge each with probability 1/2, as summarized in Table 2(b).

5 Conclusions and Further Work

In this short paper we have suggested a new measure to compute global trust values and reputation, which can be used on networks that have negative links. We have shown that it converges linearly for some parameter range. The measure takes into account negative links effectively, penalizing nodes which are negatively pointed to, thereby decreasing their trust value. This might have applications in peer-to-peer systems [11], but also in online markets such as eBay. Furthermore, it might be used to analyze networks where negative links are present, such as social networks [16] and international networks [17]. In that sense, it is an alternative to measures such as betweenness [1] and eigenvector centrality [3].

The analysis offered here is rudimentary, and further experiments are need to investigate the performance of exponential ranking. We would for example need to compare its performance with other ranking methods [18,11,19]. One possible way to test performance is to create test networks with both good and bad nodes, where the methods would need to predict whether the nodes are good or bad based on some positive and negative link topology. A short preliminary analysis shows that the suggested method ought to perform well. Extending this method by including some personalization could possibly allow for prediction of signs of links, and we could thus test performance by replicating earlier experiments [19,25].

References

1. Freeman, L.: A set of measures of centrality based on betweenness. Sociometry 40(1), 35–41 (1977)
2. Freeman, L.: Centrality in social networks conceptual clarification. Soc. Networks 1, 215–239 (1979)
3. Bonacich, P.: Some unique properties of eigenvector centrality. Soc. Networks 29, 555–564 (2007)
4. Bonacich, P.: Factoring and weighting approaches to status scores and clique identification. J. Math. Sociol. 2, 113–120 (1972)
5. Watts, D.J.: A simple model of global cascades on random networks. Proc. Nat. Acad. Sci. USA 99(9), 5766–5771 (2002)
6. De, P., Singh, A., Wong, T., Yacoub, W.: Sexual network analysis of a gonorrhoea outbreak. Sex Transm. Infect. 80, 280–285 (2004)
7. Christley, R., Pinchbeck, G., Bowers, R.: Infection in social networks: using network analysis to identify high-risk individuals. Am. J. Epidemiol. 162(10), 1024–1031 (2005)
8. Perra, N., Fortunato, S.: Spectral centrality measures in complex networks. Phys. Rev. E 78(3), 036107+ (2008)
9. Kleinberg, J.: Authoritative sources in a hyperlinked environment. J. ACM 46(5), 604–632 (1999)
10. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Networks ISDN 30, 107–117 (1998)
11. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The eigentrust algorithm for reputation management in p2p networks. In: Proc. 12th Int. Conf. on WWW, pp. 640–651 (2003)
12. Abrams, Z., McGrew, R., Plotkin, S.: Keeping peers honest in eigentrust. In: 2nd Workshop on Eco. of P2P Syst. (2004)
13. Resnick, P., Zeckhauser, R., Swanson, J., Lockwood, K.: The value of reputation on ebay: A controlled experiment. Exp. Econ. 9(2), 79–101 (2006)
14. Massa, P., Hayes, C.: Page-reRank: using trusted links to re-rank authority. In: Proc. IEEE/WIC/ACM Int. Conf. on Web Int., pp. 614–617 (2005)
15. Massa, P., Avesani, P.: Controversial users demand local trust metrics: An experimental study on epinions.com community. In: Proc. of the Nat. Conf. on Artif. Int., vol. 20(1), pp. 121–126 (2005)
16. Szell, M., Lambiotte, R., Thurner, S.: Trade, conflict and sentiments: Multi-relational organization of large-scale social networks. arXiv:1003.5137 (2010)
17. Maoz, Z., Terris, L.G., Kuperman, R.D., Talmud, I.: What is the enemy of my enemy? causes and consequences of imbalanced international relations, 1816–2001. J. Polit. 69(1), 100–115 (2008)
18. De Kerchove, C., Van Dooren, P.: The PageTrust algorithm: how to rank web pages when negative links are allowed. In: Proc. SIAM Int. Conf. on Data Mining, pp. 346–352 (2008)
19. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proc. 13th Int. Conf. on WWW, pp. 403–412 (2004)
20. Harary, F.: On the notion of balance of a signed graph. Michigan Math. J. 2, 143–146 (1953)

21. Cartwright, D., Harary, F.: On the coloring of signed graphs. Elemente der Mathematik 23, 85–89 (1968)
22. Antal, T., Krapivsky, P.L., Redner, S.: Dynamics of social balance on networks. Phys. Rev. E 72(3), 036121+ (2005)
23. Traag, V.A., Bruggeman, J.: Community detection in networks with positive and negative links. Phys. Rev. E 80(3), 36115 (2009)
24. Sarwar, B., Karypis, G., Konstan, J.: Item-based collaborative filtering recommendation algorithms. In: Proc. 10th Int. Conf. on WWW, pp. 285–295 (2001)
25. Leskovec, J., Huttenlocher, D., Kleinberg, J.M.: Predicting positive and negative links in online social networks. In: Proc. 19th Int. Conf. on WWW (2010)
26. Anderson, S.P., De Palma, A., Thisse, J.-C.: Discrete choice theory of product differentiation. MIT, Cambridge (1992)

Social Email: A Framework and Application for More Socially-Aware Communications

Thomas Tran, Jeff Rowe, and S. Felix Wu

Department of Computer Science
University of California, Davis
{trantho,rowe,wu}@cs.ucdavis.edu

Abstract. As online social networks (OSN) attempt to mimic real life social networks, we have made progress towards using OSNs to provide us with data to allow for richer and more robust online communications. In this paper, we present a novel approach towards socially-aware email. Currently, email provides too little control to the recipient. Our approach, dubbed SoEmail, provides social context to messages using an OSN's underlying social graph. This not only gives the recipient control over who can message her, but it also provides her with an understanding of where the message originated from, socially. Furthermore, users who do not wish to use the built-in social aspect of SoEmail, can send and receive emails without any changes to their behavior. The modifications we made to the email application to provide this social context are not invasive and can be easily ignored by any existing email client. We built SoEmail on top of an existing framework, known as Davis Social Links, which allows SoEmail to be completely agnostic about the underlying OSN. We created a web-based, standards-based web client for SoEmail using Facebook and Gmail as the underlying systems which has been released for public use and has had a good adoption rate.

1 Introduction

As the Web becomes more interactive, a number of venues exist to facilitate communication and meeting new people. The most ubiquitous of these medias are online social networks (OSNs) which has immensely grown in popularity. While OSNs do a great job of mapping out a user's social connections, the friendships are, in general, symmetric and binary. It is binary in the sense that two users either are friends, or they are not. There is no system for qualitatively rating friendships other than controls to allow specific friends to access parts of a user's profile. All friendships are essentially the same at the system level, which is obviously inconsistent with real world social networks. We believe that without an ability to rate friendship and trust, OSNs are incomplete. This is crucial n order to more accurately model how real world networking works. Our belief has been echoed by previous research by Krishnamurthy et al. [5] and Gross et al. [4].

Since the underlying graphs of OSNs is a rough reflection of real world social networks in a digital format, we believe that the small world hypothesis [8] also

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 203–215, 2010.

should apply to OSNs. As a result, social networks can be used for more than just communicating with immediate friends. In fact, it is theoretically possible for a user to connect to any other user with about six intermediary friends. While it may be difficult to manually search for a set of connecting intermediary friends between two users, our system can efficiently and automatically find these social routes on behalf of the user.

In this paper, we attempt to rectify the current status of OSNs in order to improve online communication by adding social context. Our work in the past includes an instantiation of Davis Social Links[1] which adds a layer above OSNs (Facebook, in our implementation) that allow for extended features such as maintaining trust and finding social routes to other users. In this paper, we show how we improve upon DSL and develop a web-based, standards-based email client that adds in social context without being intrusive to the end users. In section 2, we discuss in more detail the problem statement and things that have been done in the past to alleviate some of the shortcomings of OSNs. In section 3, we present a quick overview of DSL and how it connects with SoEmail along a discussion on some improvements to DSL. Once the reader has an understanding of DSL, we then delve into how we implemented SoEmail in section 4. We continue the discussion in section 5 by examining challenges and points for improvement. Finally we conclude our paper in section 6 and highlight some avenues for future work related to SoEmail and DSL.

2 Problem and Prior Work

Social networks have gone a long way towards mapping social connections between friends and peers in the real world. However, they are currently inadequate at adding social context to user-to-user communication. Most social networks, such as Facebook [2] and MySpace [9], provides no social context for their messaging platforms at all. LinkedIn [7] has built in functionality to be introduced to another user through a mutual connection. However, we believe that even LinkedIn's feature is too conservative and narrow in its goals. Instead of only being able to communicate to users two hops away (i.e. a friend of a friend), we envision a system in which users can communicate to others through many intermediary hops. Furthermore, while social networks can model connections in real life networks, they are currently unable to practically rate friendships or model the level of trust between two users.

Finally, in most social networks, a user can control who communicates to them only by using very sweeping policies that are neither dynamic nor easy to control with great precision regarding who can communicate with the user and who cannot. For example, in Facebook, a user can specify who can message to them by setting a global policy regarding everyone who is not their friend and by blocking specific users.

To this end, we have previously designed and published work on a system which adds social context to multiple-hop messages named Davis Social Link (DSL). However, for reasons to be discussed in the next section, DSL was incomplete as a messaging platform. We will later see how SoEmail utilizes DSL's functionalities to build a complete, more robust and standard-compatible messaging system.

3 Overview of Davis Social Links (DSL)

Since SoEmail relies on DSL, it would be useful to first give a brief overview of DSL along with mentioning some additional features since [14].

Davis Social Links (DSL) is built on top of existing OSNs. Users can sign up for DSL service, which provides their social graph to DSL, allowing the system to help the user establish social context in online communications. In our proof of concept, we have utilized Facebook's API to gather data on each user's social connections. Using this information, DSL publishes it's own API through SOAP. This allows third party applications to leverage the power of social networks while being relatively agnostic about which network (eg. Facebook vs. MySpace) it is using. Furthermore, DSL differentiates itself from other APIs such as OpenSocial [10] by providing additional functionality that the other APIs do not. The biggest amongst these are the social router and the trust management system, which we will focus our overview on.

3.1 Trust Management

In today's OSNs, friendships are very basic: they are both binary and symmetric. They are binary in the sense that two people can either be friends or not. There are no levels of friendship, although many social networks have built in privacy controls that allow users to prevent certain friends from being able to access sensitive profile information or pictures. Unfortunately, this modeling of friendship is not very comparable to real world friendships. Therefore, DSL modifies each friendship, or edge, by adding a trust value τ between 0 and 1 in accordance to [12]. Trust, in our system is asymmetric, meaning that while a user, Alice, may completely trust her friend, Bob, Bob may not feel the same way about Alice. Trust is currently roughly the ratio of good messages (as rated by the message recipients) to total messages. When an edge's trust value is below a threshold, messages become dropped with probability equal to $1-\tau$. This deters spammers as each piece of spam they send out drops their trust value with their friends, which will eventually prevent the spammer from messaging others as their messages will have a high chance of being dropped. The spammer may remedy this situation by sending out good messages in addition to spam, but the spammer must send about two good messages for each piece of spam, greatly hampering his spamming ability.

We show an example of this in Fig. 2. Here, we show values marking how much each person trusts the previous hop. For example, Bob completely trusts Alice and therefore Alice's trustworthiness, as judged by Bob, is 1.00. On the other hand, Bob is not very well trusted by Carol and his trustworthiness is only 0.40. Note that each person in the relationship may judge the other differently. In this instance, while Bob completely trusts Alice, Alice may not reciprocate. In fact, Bob's trustworthiness, as judged by Alice, may only be 0.30, for example.

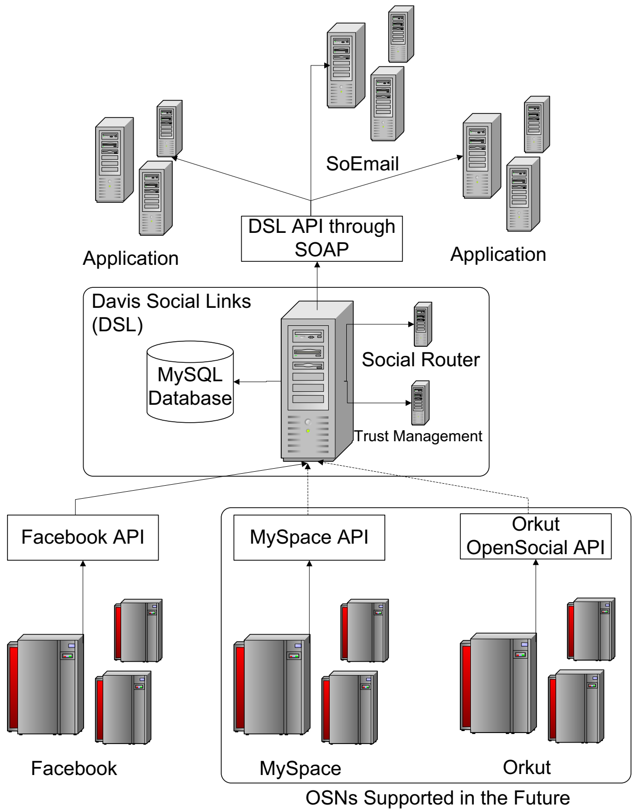


Fig. 1. Davis Social Links Architecture. Currently, DSL only supports Facebook, but it can be later expanded to handle different and even multiple OSNs. DSL is modular, and can be divided on to different servers to better handle high loads. DSL publishes an API through SOAP that other applications, such as SoEmail, can utilize to provide social context to the users.

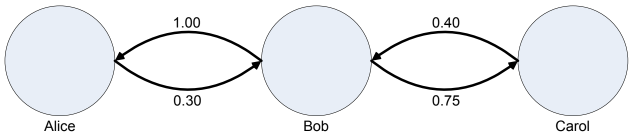


Fig. 2. An example social path. The arrows represent trust. For example, the arrow originating from Bob and going to Alice represent the trust that Bob places in Alice. Note that there are two arrows for each friendship, representing the fact that friendships are not symmetric. Here we see that two friends may have very different trust values for each other.

3.2 Social Router

In a social network, it is often advantageous to be able to communicate with people who are not direct friends. Facebook recognizes this and allows users to message people they are not friends with. However, these messages carry with them no social context and we argue that they carry no more weight nor should they be trusted more than an email from the same person. LinkedIn remedies this by allowing a mutual friend to introduce the message sender to the recipient, thus providing some social context to the messages that the recipient will be getting. We consider the recipient, in this example, to be two hops away from the sender while their mutual friend is one hop away from both. But what if a user wishes to communicate to someone more than two hops away? The social router in DSL will help the user by automatically finding a social path from the sender to the recipient, even if the recipient is an arbitrary number of hops away. Included in the social path are the trust values for each edge.

In addition to returning the social path, DSL also gives the trust value of each edge. As we noted in the previous subsection, if the trust between two nodes is sufficiently low, then the probability that the message gets through the edge connecting the two nodes is equal to τ. As a result, the probability that the message gets from it's source to it's destination is equal to the product of the probability that it gets through each hop. Note that it is NOT equivalent to the product of all the trust values, as if the trust value is high between two nodes, then the message will have 100% chance of reaching the next hop, even if the trust is not perfect (ie. the $\tau < 1$).

An application using DSL should then attach that social path to the message itself so that the recipient may understand where, socially, the message is coming from. Furthermore, the trust values will give the user a quick understanding of how credible the information in the message is. We will demonstrate how SoEmail utilizes the social path later on. The social router, coupled with the trust management system, forms the foundation of DSL and together, can effectively combat spam by excising the spammer by reducing his trustworthiness in relationship to his friends. This system is strong even against sybil attacks, as it is nontrivial to gain a new set of friends from scratch.

4 Social Email

4.1 Why Social Email?

In order to better understand how SoEmail can improve online communications, we examine a couple scenarios to compare SoEmail against standard email and communication through existing OSNs.

Receiving Message from Unfamiliar Sender. Through human networking, we often meet new contacts that may or may not be beneficial to our work or research. As a result, humans have created many social mechanisms to establish their credentials and reputation to a new person. In the scenario with Alice, Bob, and Carol, where Alice and Carol do not know each other, Bob can provide one such mechanism. Bob can simply introduce Alice to Carol and by leveraging their trust of him, he can establish Alice's and Carol's credentials and reputation to each other. This is a very common social tool in the real world. In email, this can be emulated by having Bob email Carol, speaking on Alice's behalf. However, in the online universe, this can be greatly simplified and improved. Instead of having to find only one intermediary node for which both Alice and Carol are friends, we can find chains of nodes, linking Alice and Carol together. By leveraging the trust of that chain, we can ensure Alice and Carol's reputation to each other, without having to involve everyone in the middle. Furthermore, through SoEmail, we completely eliminate the need for the end users to know anything about the intermediary nodes. They can simply find a path, and send a message along it, with the understanding that SoEmail will modify the appropriate trust values automatically.

SoEmail vs. OSN Message Systems. The idea that we can automatically link complete strangers through a series of friends on an OSN is now new. LinkedIn already allows users to introduce themselves to a stranger through a mutual friend. However, unlike SoEmail, there is no feedback mechanism inherent in these OSN systems. If a user abuses the social graph to reach new users for the purposes of spam, there is little recourse. It may be difficult, for example, for the recipient of the spam to retaliate, especially if the sender is several hops away. However, SoEmail addresses this issue by making trust and reputation an integral part of the system, allowing the recipient to punish the sender and the sender's friends who are enabling the spam to be propagated.

4.2 Building SoEmail

At its core, SoEmail is a web-based, standards-based email client. We considered the usability of SoEmail and decided that the user should not have to learn anything about DSL or social routing if they did not want to. To this end, we strived to make SoEmail as similar to normal email clients as possible with the addition of the social context for users who are interested. SoEmail acts like any other email client: it allows users to access, send, and read emails in a manner consistent with their prior experience. Users do not have to learn anything new to use the basic functionality of SoEmail. We believe that other applications building on top of DSL can and should strive for the same goals. Since DSL does not intrude upon the communication itself nor upon the usage patterns, we believe that the addition of social context to any form of existing electronic communication (eg. Skype) should not sufficiently hinder the usability of the medium.

SoEmail is a web-based IMAP client built on top of SquirrelMail [13], a standards-based webmail package, which is also used by the UC Davis computer science department. We chose SquirrelMail because it was open source, simple and easy to customize. The web client connects to an IMAP server and can retrieve, read, and send emails like a normal client. We have further installed a plugin, multilogin [6], to allow SoEmail to support more than one

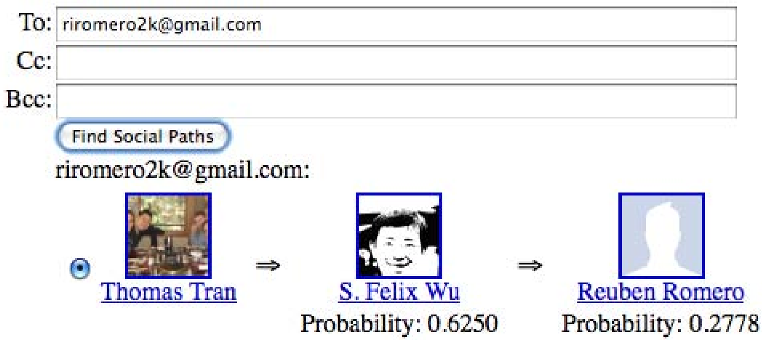


Fig. 3. The message sender can see the social path that will be used by SoEmail to send to the recipient along with the associated trust value. The shown path is a real one in our system between two DSL researchers.

email server. SquirrelMail, and as a result, SoEmail, is written in PHP, which is an excellent language for this project since it is easy to update and maintain and we can deploy SoEmail on any LAMP (Linux, Apache, MySQL and PHP) server, allowing for easy portability and replication. Since LAMP servers are so ubiquitous, SoEmail requires very little additional resources to run. SoEmail utilizes a MySQL database on the backend that stores the association between email address and Facebook ID along with, as we shall see, message IDs for each email sent through SoEmail to track the social path used by each email. As we will discuss later, we have leveraged the technology underlying SoEmail to allow for an easy port to the ProtoGENI clusters.

In addition to the plugin, our copy of SquirrelMail has been heavily modified to connect to DSL. Users can optionally use the social aspects of DSL in the client, adding social context to emails being sent out. SoEmail tags each message with an additional header field, X-DSL. This field contains a message identifier, which SoEmail associates with the path used by the sender. SoEmail tracks all the message identifiers and the corresponding social path used for that message. We decided against putting the social path in the header itself, since it can be easily forged or tampered. When the recipient opens the email in SoEmail, they can see the social context associated with the message. However, the user can open the email in any other web client without any compatibility concerns; the user would simply not see the social context as other web clients would not know what to do with the X-DSL field. As shown in Fig. 3, the message sender can see the social path that will be used by SoEmail to send to the recipient along with the associated trust value. We are currently exploring algorithms that would provide a few good social paths to the user so that they may further customize the social context of the message if they please. For example, there may exist a social path to the recipients along the sender's personal friends and another path along the sender's professional contacts. If the sender can choose which path to take, she can improve the quality of the social context being sent to the recipient.

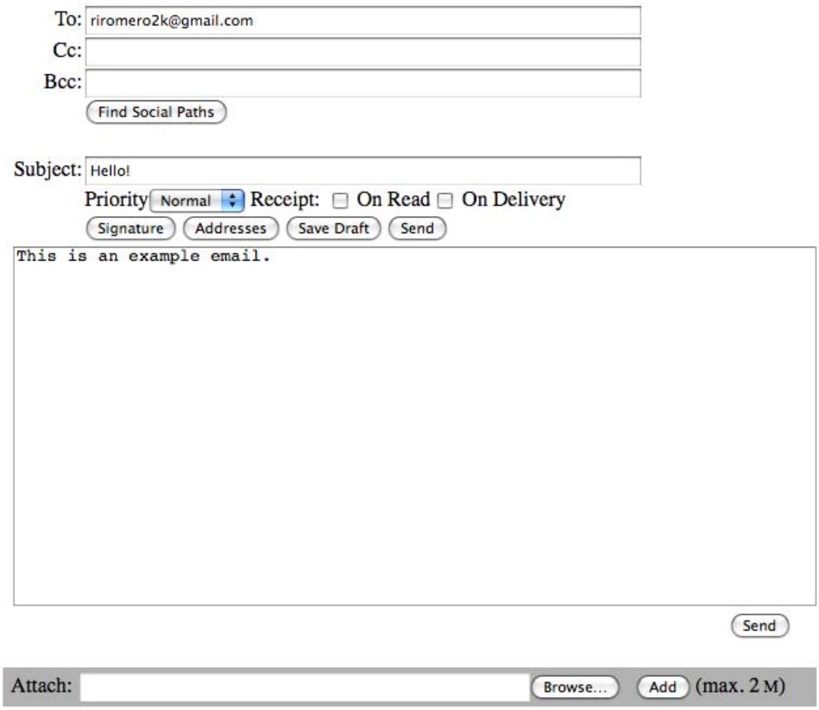


Fig. 4. The sender fills out the list of recipients that the email is intended for, he needs to only click one button, the "Find Social Paths" button to view the social path. Even if he were to not look at the social path, one will be found automatically and attached to the message. As a result, the sender can benefit from the social context provided by SoEmail and DSL without even realizing it.

4.3 SoEmail as a Service

SoEmail has been deployed as a service for anyone to use. As such, we strove to make it more robust than a typical proof-of-concept. We chose to support the Gmail servers first since they are by far the most popular email provider amongst our peers to have free IMAP servers. We are also working with the UC Davis computer science IT staff to add the UC Davis computer science department's server as well. We further encourage anyone interested in using SoEmail with a different email provider to contact us. The process for adding more email hosts has been streamlined and new ones can be added with relative ease.

In addition to ensuring that SoEmail caters to a large number of users, we also kept the DSL aspect of sending and receiving emails as un-intrusive as possible. As seen in Fig. 4, after the sender fills out the list of recipients that the email is intended for, he needs to only click one button to view the social path. Even if he were to not look at the social path, one will be found automatically and

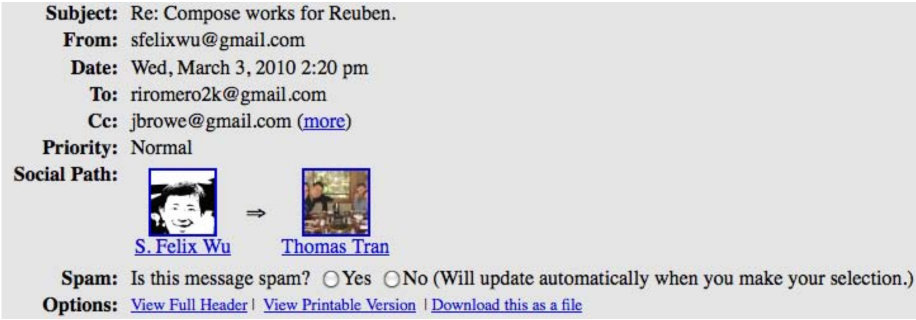


Fig. 5. A screenshot of the recipient's view of the message. Here, the social path is automatically displayed in the header, complete with links to the Facebook profiles of the nodes leading up to the recipient. This aids the recipient but for users who do not wish to utilize the social aspects of SoEmail can simply ignore the header and use the web application like any other email client.

attached to the message. As a result, the sender can benefit from the social context provided by SoEmail and DSL without even realizing it. In Fig. 5, we show a screenshot of the recipient's view of the message. Here, the social path is automatically displayed in the header, complete with links to the Facebook profiles of the nodes leading up to the recipient. This aids the recipient but for users who do not wish to utilize the social aspects of SoEmail can simply ignore the header and use the web application like any other email client.

In order to improve the functionality and usability of SoEmail, we strive to gather usage data and feedback from our users. To this end, we have created a Facebook fan page to gather feedback and functionality requests along with demonstrating SoEmail at various GECs and talking with people willing to try out our application at the conferences. Users have provided invaluable feedback regarding the system and their usage pattens have helped us identify areas where we can improve usability and stability. For example, many users have expressed concerns with providing their email password to a third party. While SoEmail never stores passwords, we understand their concerns and are working to hand over the system to the UC Davis computer science support staff, adding credibility and trust to the system since it will be managed by professionals who also handle the computer science email system.

While the application is still in its infancy, we have had moderate success getting users to test and utilize the system. Currently, there are 48 users with 55 email addresses associated with SoEmail, despite the fact that Gmail is the only IMAP server currently supported. These users have sent 277 emails to 574 different recipients.

SoEmail is currently open to anyone with a Gmail address and we invite anyone interested to join and test our system at http://soemail.cs.ucdavis.edu.

4.4 SoEmail on ProtoGENI

Our larger vision is to investigate techniques for incorporating application level social context into next-generation Internet architectures. To realize this, a framework for testing new social routing schemes that use information from social-network based applications is needed. We choose the emerging GENI [3] testing infrastructure as a basis for our investigations. In particular, we have installed and tested both our DSL trust based social routing service, and our SoEmail social email client in the evolving protoGENI [11] testbed under development at the University of Utah. ProtoGENI is an NSF-funded set of connected computer nodes users can utilize to experiment with new internet protocols. The protoGENI testbed provides an emulated testing environment similar to Emulab, but distributed across a variety of sites connected via high-speed backbone network links. To install our DSL service and SoEmail application, a small experimental testbed with several emulated hosts was created. One host was configured with the Red Hat 9 Enterprise operating system running an Apache 2 web server and a MySQL database to provide the DSL core service. We transfer our operational DSL database to this server running inside the protoGENI testbed and operate it as a mirror DSL trust base routing system. A second protoGENI host was configured as a SoEmail IMAP client and linked to the DSL server inside the testbed to provide the necessary social services. Finally, this second machine was configured to make the SoEmail client running inside protoGENI available to users in the current Internet. By distributing the DSL core service across multiple protoGENI sites, we can test new, next-generation Internet routing schemes using realistic networking components, and with traffic generated by external users interacting with SoEmail and other social networking applications from the existing Internet. Our experience shows no significant performance penalties from running inside protoGENI versus running on the Internet itself.

5 Discussion

While SoEmail has been stable so far, we recognize that there are some issues that we would like to address in the near future. Most of these issues have been brought up to us by our users. Below, we have them outlined roughly in ascending order of complexity.

5.1 Supported Email Providers

Many people have requested that we expand our list of support email providers. To this end, we have made efforts to streamline the process of adding new email servers to SoEmail and we are working closely with the UC Davis computer science IT staff to add our department's email. This has been an eye opening experience for us as some email servers, such as the UC Davis computer science one, requires additional protective measures not fully supported by SquirrelMail, such as tunneling and VPN. However, our IT support staff has been extremely helpful with workarounds to these privacy methods. We expect to fully support the UCD CS email server soon.

Afterwards, we will apply our experience from setting up SoEmail to work with the UCD CS server to connect to other IMAP servers.

5.2 Preventing Replay Attacks through X-DSL Field

Currently, we cannot prevent someone from replaying the message with the X-DSL field or forging the field, setting the identifier to a different message that has been sent in the past. We would like to eventually replace the X-DSL field with a cryptographically secured string and token that cannot be forged, perhaps encrypting a timestamp in the field to ensure that the message was not forged. Some methods for this have been discussed in theory and we hope to settle on one and implement it by the publication date of this paper.

5.3 Slow Initial Load Time

Many users of SoEmail, ourselves included, have noticed that loading the initial page that lists all of our emails in our inbox takes an unusually long time. Rough experiments have put the time from the moment the user logs in to when the inbox finishes loading at around 30 seconds, much longer than what today's internet users are familiar with. This has had a very strong impact on the usability and user experience regarding SoEmail. While working with the UCD CS IT staff, they have confirmed with us that the issue also appears in their instantiation of SquirrelMail. Our goal is to contribute to the SquirrelMail project if possible by addressing and attempting to optimize the load times. This would not only be beneficial to SoEmail, but to all administrators who deploy SquirrelMail, including our IT staff here at UC Davis.

5.4 Trusting SoEmail with Passwords

Many users have expressed concerns with providing their email password to a third party. While SoEmail never stores passwords, as security minded researchers, we understand their concerns and are working to hand over the system to the UC Davis computer science support staff, adding credibility and trust to the system since it will be managed by professionals who also handle the computer science email system.

5.5 Social Path Privacy Concerns

When a user is selected as a node for a social path, some information about the user's friends (namely, that they are friends with the node preceding and proceeding them) and the trust values are revealed to the sender and recipient. Currently, we display all this information, but we have been discussing hiding the social path or limiting its visibility so that the sender and recipient can only see the nodes on the social path that are close to them.

5.6 Supporting More than One OSN

A few users believe that by adding in one or more social networks in addition to Facebook can really flesh out the social graph along with providing better, more meaningful social paths that possibly take advantage of both OSNs. This issue is also partially related to DSL. However, the solution for this would be non-trivial as each user must also link all of their profiles on different OSNs together. However, if we can achieve this goal, we believe that the data and paths we can provide to the users will be much richer.

5.7 Trust and What It Means

One point of confusion for users of SoEmail and any DSL related application is the meaning of the trust value. The numbers are somewhat relative and do not easily translate to real world understanding of trust. For example, what does a trust value of 0.75 mean to the user? More often than not, it just means that the trust is high, but no more than that. Most users We have been experimenting with associating definitions with ranges of trust (eg. a trust rating of 0.9-1.0 can be called "Extremely Strong Trust") as a way of aiding users to understand trust and its impact on the social context provided to them.

6 Conclusion and Future Work

As seen in Section 5, SoEmail has some room for improvement, but as demonstrated here in this paper and through user feedback, our application has been performing well and is well received by its users. Simulations have shown that our system should be stable with even up to 40% of users being spammers, giving the system incredible robustness against spammers. We built SoEmail with usability in mind, and while we have been asked for clarification on how things such as trust and social paths work, we have not had one user tell us that the interface is too confusing.

We would like to keep user satisfaction high by addressing some of the shortcomings of SoEmail that we have already discussed. In particular, we are confident that two of the more popular issues will be fully addressed soon and we are internally discussing solutions to the other issues. The continued improvement of SoEmail is important to us because not only is it used as a service by dozens of people on a daily basis, but also because we would like to use SoEmail as a means to gather more data on how trust changes over time in a real environment with real users. To that end, we also plan on adding more anonymizing, data collecting functionality within both SoEmail and DSL. Finally, we hope to recruit more users to the system so that we can get more data and feedback. We have been actively advertising SoEmail at the conferences we attend and will continue to do so through demonstrations or presentations.

Once we have SoEmail in a state that we are satisfied with, we would like to polish SoEmail to the point where we can hand it off to the UC Davis computer science support staff to deploy and maintain. This will lend credibility to the system which would hopefully attract people otherwise deterred from passing SoEmail their password.

On the theory side of DSL, we have been working closely with the department of sociology to better map out online social networks to real life social networks. We would like to get a better understanding of social interactions and feedback in the real world so that we may model it better in DSL.

Acknowledgments

We would like to thank Ken Gribble for his tireless help with setting up the SoEmail server, configuring it to our specifications, and providing patient support as we break everything he set up. This research is funded in part by National Science Foundation under CNS-0832202, Intel, and an MURI grant from Army Research Office under the ARSENAL project and we would like to thank them for continued support.

References

1. Banks, L., Bhattacharyya, P., Wu, S.F.: Davis social links: A social network based approach to future internet routing. In: FIST 2009: The Workshop on Trust and Security in the Future Internet (July 2009)
2. Facebook, http://www.facebook.com/
3. GENI, http://www.geni.net/
4. Gross, R., Acquisti, A., Heinz, J.H.: Information revelation and privacy in online social networks. In: WPES 2005: Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society, pp. 71–80 (2005)
5. Krishnamurthy, B., Wills, C.E.: Characterizing privacy in online social networks. In: WOSP 2008: Proceedings of the 1st Workshop on Online Social Networks, pp. 37–42 (2008)
6. Lesniewski, P.: http://squirrelmail.org/plugin_view.php?id=148
7. LinkedIn, http://www.linkedin.com/
8. Milgram, S.: The small world problem. Psychology Today 61, 60–67 (1967)
9. MySpace, http://www.myspace.com/
10. OpenSocial, http://www.opensocial.org/
11. ProtoGENI, http://www.protogeni.net
12. Spear, M., Lu, X., Matloff, N., Wu, S.F.: Karmanet: Leveraging trusted social paths to create judicious forwarders. In: IFCIN 2009: Proceedings of the First International Conference on Future Information Networks (2009)
13. SquirrelMail, http://squirrelmail.org
14. Tran, T., Chan, K., Ye, S., Bhattacharyya, P., Garg, A., Lu, X., Wu, S.F.: Design and implementation of davis social links osn kernel. In: SNAS 2009: Proceedings of the Workshop on Social Networks, Applications, and Systems, held in conjunction with the 2009 International Conference on Wireless Algorithms, Systems and Applications (2009)

Measuring Message Propagation and Social Influence on Twitter.com

Shaozhi Ye and S. Felix Wu

Department of Computer Science, University of California, Davis
2063 Kemper Hall, 1 Avenue Shields, Davis CA 95616
sye@ucdavis.edu, wu@cs.ucdavis.edu

Abstract. Although extensive studies have been conducted on online social networks (OSNs), it is not clear how to characterize information propagation and social influence, two types of important but not well defined social behavior. This paper presents a measurement study of 58M messages collected from 700K users on `Twitter.com`, a popular social medium. We analyze the propagation patterns of general messages and show how breaking news (Michael Jackson's death) spread through Twitter. Furthermore, we evaluate different social influences by examining their stabilities, assessments, and correlations. This paper addresses the complications as well as challenges we encounter when measuring message propagation and social influence on OSNs. We believe that our results here provide valuable insights for future OSN research.

1 Introduction

Online social networks (OSNs) have become a major Internet service for people to communicate with each other. To measure the information propagation process and the driving force behind it, social influence, this paper presents a detailed analysis of 58M messages over `Twitter.com`. More specifically, the following problems are investigated in this paper.

- How do messages propagate on Twitter?
- How can we measure social influence on Twitter?
- How do different influences correlate with each other?

To answer these questions, we first need to formally define message propagation. If a user u_0 sends a message M to a set of users $U = \{u_1, u_2, \dots\}$, we say M is *originated* from u_0 and *propagated* to U.

Furthermore, we are interested in message propagation over multiple hops, i.e., message cascading [2,3]. For example, if a user $u_1 \in U$ also announces M after receiving it from u_0 and another user $v \notin U$ receives from u_1, then M is propagated by two hops, i.e., $u_0 \rightarrow u_1 \rightarrow v$. Each hop increases the chance for M to reach more users. Our analysis shows that OSNs are able to propagate messages quickly, which is an indicator of small-world effect (Section 3).

Different from the forwarding/broadcasting models in computer networks or peer-to-peer (P2P) networks, the content of M is often modified by users during

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 216–231, 2010.

the propagation over OSNs, which makes it hard to trace M. In this paper, we model the propagation of a message as a *message flow*, which is a tree of messages. Each node in the tree is a message, which is a reply to its parent node. When a message has multiple originators who receive this message outside of the social network of interest, we may model it as a forest by making each originating message the root of a tree. Section 3 presents our analysis of message flows on `Twitter.com`.

In some cases, especially in the event of breaking news, it is possible to identify a set of closely related messages by keywords. These messages are all about the same event although they may belong to different message flows. Section 4 presents our analysis of 500K messages related to Michael Jackson's death.

Furthermore, as previous studies have shown that during information propagation some users are more influential than others [1], we evaluate several social influences by developing a set of metrics to compare different influences. More specifically, we address the following problems.

– How to evaluate social influence metrics (Section 5.1)
– The change of social influence over time (Section 5.2)
– How to assess social influence (Section 5.3)
– Correlations between different influences (Section 5.4)

The rest of this paper is organized as follows. Section 2 describes how we collect the data set for analysis. Section 3 presents our measurement results for message propagation and Section 4 shows how the breaking news of Michael Jackson's death spread through the social network. Then Section 5 evaluates five social influence metrics. After reviewing prior work in Section 6, the paper concludes with Section 7.

2 Data Collection

`Twitter.com` is a "real-time information network" for people to share news and communicate with others. A user (twitter) may send messages (tweets) via any of the following channels:

– Twitter website or its API
– Cellphone short message services (SMS)
– Instant messenger (IM)
– E-mail

Many Twitter clients have been developed on top of these channels. Messages are broadcast by Twitter to the sender's followers through the channels they choose. The way Twitter distributes messages provides a lot of flexibilities for users to send and receive messages. Twitter has almost doubled its users in 2009 and is estimated to have 70M users as of Jan. 2010 [12].

OSNs are built on top of social relationships or friendships while the definition of friendship varies from one OSN to another. The relationship between two users on `Twitter.com` is *following*. More specifically, if Alice is following Bob, Alice is able to receive all Bob's (public) messages. In this case, Alice is a *follower* of Bob and Bob is a *followee* of Alice.

In June 2009, the news of Michael Jackson's death spread all over the world. Many online social networks (OSNs) were flooded by messages/news related to this breaking event. We started collecting related messages from `Twitter.com` on June 27th, 2009, two days after the tragedy. The collection process was performed as follows.

1. Send the following two queries to Twitter's search service[1]: "Michael Jackson" and "MJ." Twitter allows us to send up to 20K queries per hour per whitelisted IP.
2. Parse the returned results (messages) to find the users who posted these messages. From June 27th, 2009 to August 16th, 2009, 716,588 users were collected.
3. Crawl the profiles and follower/followee lists of the users we found in Step 2. We found 683,160 users with valid and unprotected profiles[2].
4. Crawl all the messages these users have posted. Twitter allows us to collect up to 3,201 most recent messages for each user. We crawled 58,500,320 messages all together, i.e., each user posted 81 messages on average. Step 3 and 4 took us about 17 days.

Among all the messages we crawled, we select the tweets containing "Michael Jackson" or "MJ" as MJ related messages. To filter the false positives introduced by the query "MJ," which mostly are generated by URL shortening services such as `tinyurl.com` and `bit.ly`, we require that for each message there is no leading character before "MJ." After removing the noise, we found 549,667 MJ related messages (about 1% of the entire data set we crawled) posted by 305,035 users. 548,102 messages were posted after Jun 25, 2009 12:21:04 p.m. (PST), when the 911 call was made to save Michael Jackson's life. We assume that these messages are related to the breaking event.

Thus we collected two datasets. One is the entire data set for analyzing the overall (or average) message propagation patterns and the other, referred as the "MJ" dataset, is used for breaking news propagation analysis.

In addition, to get the social graph for computing propagation distance, we crawled 61,907,902 nodes with 1,520,959,208 links, which covers 88% of the entire Twitter network according to the estimation given by Moore [12].

2.1 Message Format

A message (status or tweet) provided by Twitter API contains the following fields.

– User_id: Unique identifier for the user who posted this message.
– Id: The message ID, which is unique for messages posted by the same user. Two messages posted by different users may share the same message ID.

[1] `http://search.twitter.com`
[2] A protected profile is not available to public thus our crawler can not crawl it.

– Text: The content of the message, up to 140 characters.
– Created_at: The creation time for this message.
– Source: The Twitter client software which was used to post the message.
– Truncated: Messages having more than 140 chars will be truncated by Twitter and have this field set. None of the 58.5M messages are truncated.
– In_reply_to_status_id: The message ID which this message replies to.
– In_reply_to_user_id: The user ID which this message replies to.
– Favorited: Indicating if the message is a favorited one, which actually is rarely used. 261 out of the 58.5M messages are marked as favorited.

2.2 Popular Sources of Messages

Among the 58.5M messages we crawled, the top three sources (Twitter clients) and the fractions of messages they contribute are as follows.

1. `Twitter.com` (38.0%);
2. TweetDeck (11.8%), a popular Twitter client;
3. TwitterFeed (5.7%), a service which enables users to send messages by posting to their blogs.

Besides, five major mobile Twitter clients, including Tweetie, UberTwitter, and TwitterFon, contribute to 21.1% together. In our data, there are 112 sources via which at least $10,000$ messages were sent.

For the MJ data set, 58.5% messages were posted through the Twitter website, and the five top mobile clients contributed only 11.0%. Further investigation in the data suggests that users are more likely to use mobile devices for sharing events related to themselves, for example, where they are or what they are doing.

2.3 Popular Tweets

We find that popular tweets, i.e., the tweets being sent by many users, fall into two categories.

– Internet slang or short phrases, e.g., the top 10 most popular tweets in our data set shown as Table 1.
– Automatically generated messages, most of which are sent by virus/worms or online services for advertisement or spam purposes. For example, there is a website which allows users to customize the background of their Twitter pages. After a user enables this service, it automatically posts an advertisement on behalf of the user to his/her followers. There are also many messages advertising for websites which claim to boost the number of followers for users. Some of them are in fact performing phishing or “bait and switch” attacks. Extensive worms/epidemiology studies have been devoted to the propagation of such messages therefore we do not look further into it in this paper.

Table 1. Top 10 most popular tweets

Number of messages	Content
74,141	“LOL”
25,041	“:)”
19,455	“Thanks!”
18,005	“LMAO”
12,535	“” (Empty)
11,917	“LOL!”
10,295	“:D”
9,881	“:(”
9,861	“Thank you!”
9,207	“Thanks”

3 Measuring Message Propagation

On `Twitter.com`, a user may reply to a message sent by another user. Such reply message has the fields of “in_reply_to_status_id” and “in_reply_to_user_id” thus allows us to match it with the message it is replying to and further identify the message flow.

28.1% messages in our dataset are replies. Huberman *et al.* [6] reported a similar portion of replies in their dataset (25.4%). The MJ dataset has a much smaller portion of replies, 9.4%, which suggests that it is more common for people to express their own feelings or opinions about MJ’s death instead of discussing it with their friends.

Message flows are identified as follows.

1. Sort all replies according to their timestamps with the earliest message on the top.
2. Walk through the sorted message list from the top to the bottom. For each message i, assuming j is the message which i replies to, i.e., j is i’s parent. If there exists a tree which has j, make i a child of j. Otherwise, create a new tree with j as its root, and attach i to j.
3. When we reach the end of the message list, output all message flows we have discovered.

We found $1,538,698$ message flows in the entire data set and each flow has 10.7 messages on average. Now we can analyze how these messages propagate by answering the following questions.

– How far away is a message propagated?
– How fast is a message replied?
– How long does a message flow last?

3.1 How Far Away Is a Message Propagated?

On `Flickr.com`, a popular photo sharing website, users may choose to become a "fan" of a photo. Cha *et al.* [3] observed that even for popular photos, only 19% of fans are more than 2 hops away from the uploaders. On Twitter, however, we find that 37.1% message flows spread more than 3 hops away from the originators, shown as Table 2. The large propagation distance indicates that Twitter is a better medium for propagating information. Meanwhile, text messages are probably easier to propagate than photos. The longest message flow in our data set consists of 1,555 replies made by 1,512 users.

Table 2. How far a message is propagated

Hops propagated	1	2	3	≥ 4
Fraction of messages	45.1%	9.4%	8.4%	37.1%

3.2 How Fast Is a Message Replied?

25% replies were made within 67 seconds and 75% were made within 16.5 minutes. This indicates that the communications on `Twitter.com` are mostly real time. The mean time to reply a message is 2.9 hours while the median is only 3.5 minutes, which indicates that there exists large delays for some replies. For example, we observed that a message got replied 20 months after it was sent.

3.3 How Long Does a Message Flow Last?

25% message flows lasted less than two minutes and 75% lasted less than an hour. In other words, most conversations ended quickly. There is also a huge gap between its mean (6 hours) and median (8.9 minutes), indicating some outliers with long life time.

Therefore we conclude that on Twitter, messages get replied quickly and propagated relatively far away, although most conversations last for a short period of time.

4 Measuring the Propagation of Breaking News

As we have stated before, the MJ dataset has fewer replies (9.4%) than the entire dataset (28.1%). More importantly, all the messages in the MJ dataset are considered to be related to Michael Jackson's death. Therefore we can examine the aggregated propagation patterns of all these messages, which are more interesting than those of individual message flows.

Shown as Fig. 1, these messages cover about two months after the tragedy. Initially there were many speculations and shocks, which corresponds to the first spike. The initial spike would be larger if we started our crawling at Day 0 instead of Day 2. There is a spike in Day 3, probably because it is the first day

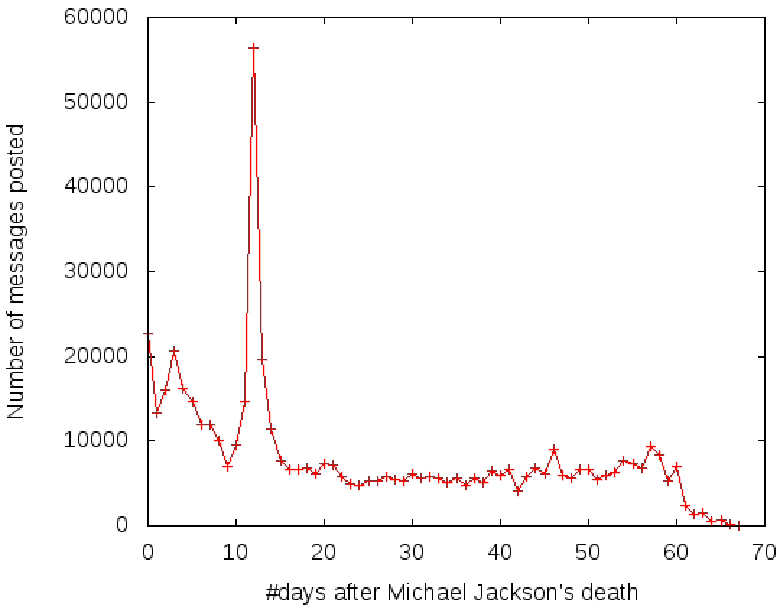


Fig. 1. Number of messages versus days after MJ's death

we queried Twitter for entire 24 hours. The largest spike occurred during the day of Michael Jackson's memorial service, July 7th, 2009, i.e., 12 days after his death. Comparing to an average day within our data collection window, there were 10 times more MJ related messages posted at that day. After that spike, the number of messages kept steady between Day 16 and Day 60, although we stopped querying Twitter to find new users (not tweets) around Day 50. This "ripple" effect suggests that breaking news does not always disappear quickly, a counterexample of the "15 minutes of fame" theory.[3]

Propagation distance can not be used here because this message (MJ's death) was widely covered in the real world. There are many originators for this message and it is hard to tell if a user gets it from a followee or not. Thus we evaluate the coverage of the message within the Twitter network. More specifically, if a user posts a message related to MJ's death, all of his/her followers will receive this message, i.e., they are *covered* by this message. As a user may receive MJ related messages multiple times, we only consider the number of unique users who are covered.

Shown as Fig. 2, the number of covered users increases quickly in the first 15 days, with a large jump around the memorial service. Within 70 days about 9M users were covered, roughly 12% of the entire Twitter graph. The number of covered users keeps increasing after the memorial service although with a

[3] According to Wikipedia, "15 minutes of fame" is "short-lived, often ephemeral, media publicity or celebrity of an individual or phenomenon."

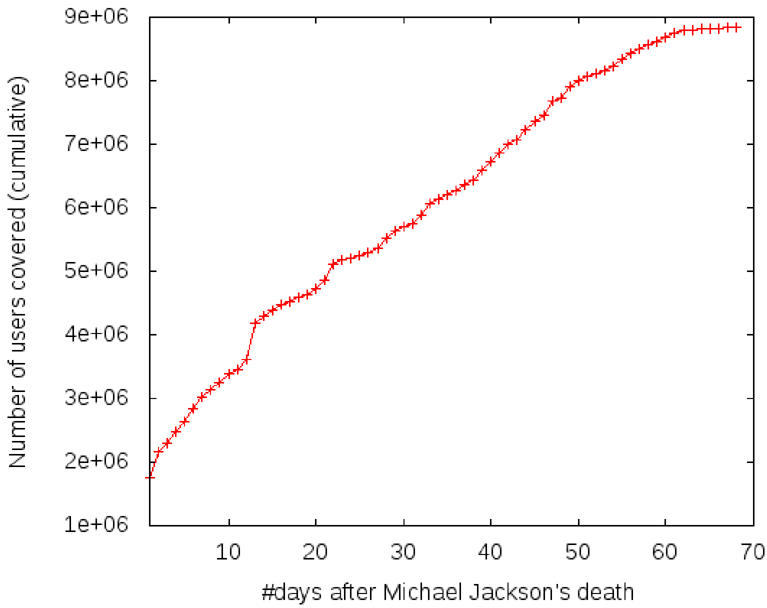


Fig. 2. Number of users covered (cumulative)

slower rate. This suggests that these messages are sent by new originators. Examining the differences between initial shocks and afterward thoughts will be an interesting direction for future work.

To further analyze the message propagation process, we define a *poster* as a Twitter user who posted at least one MJ related message. Fig. 3 shows the cumulative distributions for the number of of covered users and the number of posters. The first 5% posters covered about 20% of the users, which demonstrates the amplification power of OSNs as a medium, i.e., a small number of users may push a message to a large number of users. As more and more users post messages, few new users are covered since most users have already been covered by earlier posters. This is a strong signal of the small-world effect.

5 Measuring Social Influence

According to Wikipedia, *social influence* "occurs when an individual's thoughts or actions are affected by other people." When it comes to specific fields or application scenarios, this vague definition needs to be clarified, i.e., what is the *action* and how to determine whether an action is *affected by other people*.

For `Twitter.com`, we examine the following metrics for social influence.

– **Follower influence** (F)**:** The action here is receiving messages (following). The more followers a user has, the more influential this user is. It is also

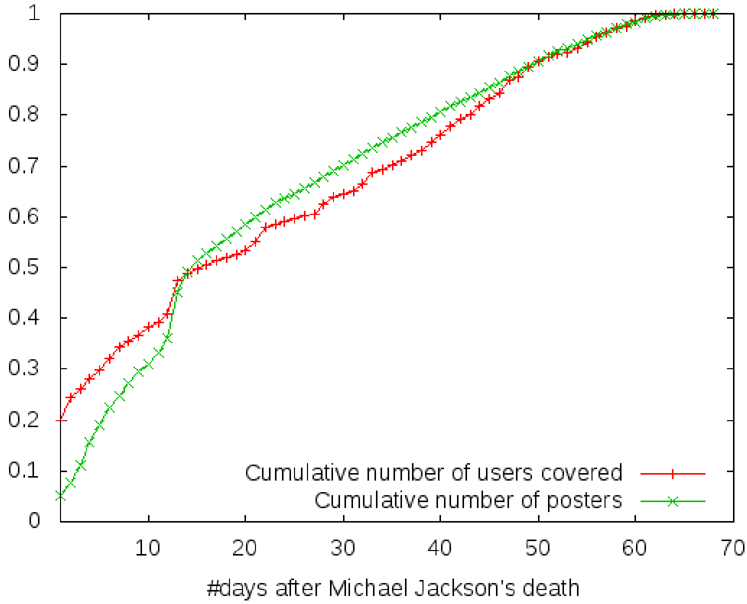


Fig. 3. Covered users versus posters (CDF)

known as *degree influence*, which corresponds to the size of the audience a user has.

- **Reply influence** (R)**:** The action here is replying. The more replies a user receives, the more influential he/she is. This influence can be quantified by the number of replies (R_M) the user receives or the number of users who make these replies (R_U). R_U is less biased towards the users who make lots of replies. We evaluate both of them in this paper.
- **ReTweet influence** (RT)**:** The action here is retweeting. Similarly, the more frequently the user's messages are retweeted by others, the more influential this user is. This can also be quantified by the number of retweets (RT_M) or the number of users who retweet (RT_U).

5.1 How to Evaluate Social Influence Metrics

A social influence metric gives a score for each user being measured. For example, F gives the number of followers of a user. The score itself does not tell us how influential a user is, while when we compare the scores of two users, the one with larger score is likely to be more influential. Thus what really matters is the relative order of the users according to the metric, which can be represented by a list of users ranked by the metric. With ranking lists, it is possible to compare social influence given by two metrics which capture different actions, for example, the number of followers (F) and the number of replies (R_M).

To evaluate how stable these metrics are, we split the dataset into two sets according to when the messages were posted and compare their ranking lists. If with a certain social influence metric, the ranking lists of these two sets are close to each other, this metric is relatively stable.

Secondly, we also compare the ranking lists given by different metrics to examine their correlations. If two ranking lists are close to each other, these two metrics are similar. We want to identify the metrics which are similar and those which are unique.

To quantify the difference between two ranking lists, the following two measures are widely used. Given n different items and their two permutations (ranking lists) x and y,

- Spearman's rank correlation coefficient (ρ) [13]:

$$\rho = 1 - \frac{6\sum(x_i - y_i)^2}{n(n^2-1)} \tag{1}$$

  where x_i and y_i are the ranks of item i in list x and y respectively.
- Kendall Tau rank correlation coefficient (τ) [7]:

$$\tau = \frac{n_c - n_d}{0.5n(n-1)} \tag{2}$$

  where n_c is the number of concordant pairs and n_d is the number of discordant pairs. Given two items i and j, if $x_i > x_j$ and $y_i > y_j$ (or $x_i < x_j$ and $y_i < y_j$), i and j are a concordant pair, otherwise i and j are a discordant pair.

Both ρ and τ are inside the interval $[-1, 1]$ and assume the value:

- -1 if a ranking list is the reverse of the other;
- 0 if the ranking lists are completely independent;
- 1 if the ranking lists are the same.

Furthermore, we are interested in the top k most influential users. It is problematic to compare users with small influentials. On OSNs many users are likely to have small influence, i.e., the long tail effect. Small variance may change their rankings a lot. For example, lots of users have only a couple of replies, getting one more reply may improve a user's rank by several thousands or more. In this paper we evaluate the ranking distance with top $1,000$, $5,000$, and $10,000$ most influential users.

When comparing the top k items of two ranking lists, there may exist $x_i < k$ while $y_i > k$ hence the two top k lists are not two permutations of the same set of items anymore. To resolve this problem, we perform a matching process to compute $\rho_k(x, y)$ and $\tau_k(x, y)$ as follows.

1. Select the top k items from list x, denoted as $\{u_i\}$, $i = 1, 2, \ldots, k$;
2. For each u_i, get its ranking in list y, denoted as y_i.
3. Let y'_i be the rank of u_i in the list of $\{y_i\}$.
4. Compute the rank distance between the top k lists of x and y'.

This matching process generates two lists with the same set of k items while it is asymmetrical to list x and y. In other words, by switching the processing order of x and y, we may get different set of k items. To get symmetrical results, $(\rho_k(x,y) + \rho_k(y,x))/2$ and $(\tau_k(x,y) + \tau_k(y,x))/2$ are reported as ρ and τ respectively in the rest of this paper unless explicitly stated.

We also report the overlap between the top k lists, i.e., the number of items which rank top k in both lists. The overlap is normalized by k for comparison between different ks.

5.2 The Change of Social Influence Over Time

To show the change of social influence over time, we sort the messages according to their timestamps and split them into two by selecting the first 50% messages as Dataset I and the second 50% messages as Dataset II. For each dataset we compute ρ, τ, and overlap for their top k lists, shown as Table 3. The results on F is not available because we only have one snapshot of `Twitter.com` thus do not know the number of friends of each user before and after a certain time.

Table 3. The change of social influence over time

Metric	Top 1,000			Top 5,000			Top 10,000		
	ρ	τ	Overlap	ρ	τ	Overlap	ρ	τ	Overlap
R_U	0.722	0.526	**73.9**%	0.688	0.496	**72.8**%	0.669	0.470	**71.9**%
R_M	0.614	0.433	57.8%	0.541	0.374	54.8%	0.521	0.358	55.7%
RT_U	**0.802**	**0.997**	54.1%	**0.778**	**0.999**	53.0%	**0.777**	**1.000**	53.2%
RT_M	0.634	0.451	65.6%	0.608	0.419	62.5%	0.574	0.393	61.1%

The maximum of each column is highlighted. For ρ, τ, and overlap, we always have the following order:

$$R_U > RT_M > R_M$$

The overlap of two lists ranked by R_U is the largest among all 4 metrics being evaluated here, i.e., the set of users with most repliers (not replies) is most stable. Its corresponding ρ and τ are the second largest (only smaller than RT_U), implying that R_U is a reliable metric. Huberman *et al.* [6] found that R_U is also a good estimator for the number of messages a user posts. Hence R_U seems to be a promising candidate for measuring social influence.

The large ρ and τ for RT_U indicates that the action of *retweeting* is more consistent over the time compared to other actions. As reported by Cha *et al.* [1], the most retweeted users were content aggregation services, businessmen and news sites. Our results suggest that such users are likely to keep their influence stable. On the other hand, users with large number of replies are mostly celebrities, whose influence fluctuates as R_M being the most unstable metric.

Except RT_U, both ρ and τ decrease as the list gets longer (i.e., k gets larger). This validates the motivation to use the top k lists instead of the entire list, i.e., the top k users are more stable than the users with lower ranks.

5.3 Assessing Influence: Messages versus Users

Both the number of messages and the number of users can be used to assess the influence. To see the difference, we compare the top k lists generated by the same influence metric with different assessments.

Table 4. Rank distance between RT_M and RT_U

Metric	Top $1,000$	Top $5,000$	Top $10,000$
ρ	0.833	0.817	0.795
τ	0.655	0.628	0.604
Overlap	82.5%	81.6%	82.1%

Shown as Table 4, the difference between RT_M and RT_U is small, and more importantly, it does not get much larger as k increases.

Table 5. Rank distance between R_M and R_U

Metric	Top $1,000$	Top $5,000$	Top $10,000$
ρ	0.749	0.574	0.533
τ	0.571	0.409	0.369
Overlap	61.1%	55.5%	57.2%

R_M and R_U, on the other hand, are relatively far away from each other. We find that the gap between them gets larger as k increases, shown as Table 5. This suggests that the distribution of replies is highly skewed, i.e., some users make many more replies than others. Checking the pair of users involved in a reply, we find that 47% replies are between the top 10% most frequently communicated user pairs.

Hence we conclude that for retweet influence, either the number of messages (RT_M) or users (RT_U) can be used whereas for reply influence, it would be prudent to evaluate the choice of R_M or R_U with the specific application scenario.

5.4 Correlations between Different Influences

In this section, we examine the correlations between these social influence metrics. More specifically, let L_i and L_j be the ranking lists generated by social influence metric i and j respectively, and $D(i, j)$ be the distance between L_i and L_j where $i, j \in \{F, R_U, R_M, RT_U, RT_M\}$ and $D \in \{\rho, \tau, \text{overlap}\}$. For each metric i, we compute $\eta = \sum_{j \neq i} D(i, j)$, i.e., summation of the distance between the ranked list generated by metric i and the ranking lists generated by the other 4 metrics. A large η means that the corresponding metric i is close to the rest of 4 metrics.

To our surprise, for all D, η of 5 metrics follows the same order, i.e.,

$$R_M > R_U > RT_U > RT_M > F$$

This order also holds for all ks we have tested. The first four metrics are close to each other with F being far away, which shows that F is a poor estimator for other social influences. Meanwhile, it suggests future social influence studies that analysis on any of the first four metrics (R_M, R_U, RT_U, and RT_M) might apply to the other three but F is likely to be an outlier thus needs to be examined carefully.

One promising application for such correlation analysis is to reduce the computation cost for expensive social influence metrics, such as the ones introduced in Section 6.3. Although some work has been proposed to compute social influences efficiently such as [15], some metrics are still expensive for large scale OSNs. Fortunately in many cases, only top influential users (i.e., users with large x) are concerned. If we can identify two close metrics x and y, where x is the expensive one we are interested in and y is a simple metric such as R_U, then y can be used as an estimator for x. More specifically, as users with large x probably have large y too, we may select top k users with metric y and compute metric x for these k users. As long as x and y are close to each other, we will not miss many users with large x.

6 Related Work

There is a large body of literature on information propagation and social influence, most of which looks at traditional social networks. Here we limit our discussion to OSN related models and measurements.

6.1 Content Generation

Guo *et al.* [5] measured how user content is generated on three popular OSNs (the name of these OSNs are not revealed in their paper). They found that the number of active posters (i.e., users who post a lot of messages) is much larger than that of a power-law distribution, which suggests that the network is not dominated by a small number of users.

The Web Ecology Project [8] examined 1.8M tweets about Michale Jackson's death and show how users express emotion on Twitter. This work focuses on content and semantic analysis and provides us with insights for how users encode emotional content with a small number of words. Our analysis in Section 4 complements their work.

6.2 Information Propagation

Huberman *et al.* [6] defines *friends* of a user A as the set of users with whom A has exchanged directed messages. By examining 300K Twitter users, they observed that the correlation between the number of messages and the number of friends is larger than the correlation between the number of messages and the number of followers/followees. The friend network is shown to be much sparser than the follower/followee network, which may alleviate the scalability problem for OSN studies.

Kwak *et al.* [9] examined the propagation of 106M retweets and found that up to 1,000 followers, the average number of users being covered by a retweet is not affected by the number of followers the originator has. Another interesting observation is that although the median time to retweet for the first hop is large (about an hour), that of the second hop is quite small (less than 10 minutes).

Lerman and Ghosh [10] measured how popular news spread on Digg (a social news aggregator) and Twitter. Voting on Digg and retweeting on Twitter are used to identify related messages. The number of votes/retweets is shown to increase quickly within a short period of time and saturates after about a day. The distribution of story sizes (number of votes/retweets a story has) is approximately a normal distribution instead of the widely observed power-law distribution. Their observations are based on the aggregated patterns of many stories thus does not focus on breaking news. A key difference is that breaking news such as Michael Jackson's death lasts much longer time and covers many more users. Lerman and Hogg [11] further developed a model to estimate the popularity of news. It is not clear if their model is able to predict the spread pattern for breaking news as such stories tend to be outliers compared to average popular stories.

6.3 Social Influence

Kwak *et al.* [9] ranked Twitter users by the number of follows and by PageRank and found that these two rankings are similar. The ranking by retweets differs from the previous two rankings, which is similar to our findings in Section 5.1. A variation of τ is used to compare these rankings. Their comparison is preliminary as only these two influences are considered. They also did not look at the change of influences over time.

Weng *et al* [14] examined top 1,000 Singapore-based twitters plus their followers and followees. They found the following relationship is highly symmetrical and developed a topic-sensitive PageRank-like influence measure, TwitterRank. The basic idea is that given a topic, the social influence of a twitter is the sum of the social influences of his/her followers. It would be interesting to extend TwitterRank by substituting following with replying/retweeting.

Ghosh and Lerman [4] proposed a model to predict the number of votes a user's post generates on Digg by considering graph properties of the network, which decide how news can be delivered. Voting influence on Digg is similar to replying and retweeting influence on Twitter. Applying their model to Twitter would be another direction for future work.

Cha *et al.* [1] investigated three social influence metrics (in-degree, retweets, and mentions) across topics and time with the messages of 6M Twitter users, which is probably the closest work to this paper. They found: 1) Popular users with large number of followers may not necessarily get more retweets or mentions. 2) Users can hold influences over multiple topics. 3) Limiting tweets to a single topic may help users gain influences.

The differences between their work and our analysis in Section 5 are as follows.

- Cha *et al.* [1] did not have similar analysis as Section 5.3 and 5.4.
- We discuss in detail how to evaluate social influence metrics, such as the matching process to compare the top k lists.
- Cha *et al.* [1] only considered ρ and overlap.

7 Conclusions

This paper presents a measurement study of 58M messages sent by 700K users on `Twitter.com`.

First, by examining message flows, we find that replies arrive quickly and a significant portion of messages propagate far away from the originator, i.e., the discussions are not restricted to his/her followers. Further, we show how messages related to Michael Jackson's death spread through the network, which demonstrates the power of Twitter as a social medium.

Secondly, we evaluate different social influences by examining how they change over time, how to assess them, and how they correlate with each other.

The discussions in this paper reveal the complications we have to deal with to characterize message propagation and evaluate social influence. We believe that the analysis and measurements presented here pave the way for systematic measurements and investigations of OSNs.

Acknowledgements

We thank `Twitter.com` for whitelisting our crawlers during the past year. Min Liu provided insights to some of our observations. Dr. Norm Matloff made valuable comments on an earlier version of this paper. We are also grateful to anonymous reviewers for the suggestion of using tree/forest to model message flows and pointers to related work.

References

1. Cha, M., Haddadi, H., Benevenuto, F., Gummadi, K.P.: Measuring user influence in Twitter: The million follower fallacy. In: ICWSM 2010: Proceedings of International AAAI Conference on Weblogs and Social Media (2010)

2. Cha, M., Mislove, A., Adams, B., Gummadi, K.P.: Characterizing social cascades in Flickr. In: WOSP 2008: Proceedings of the First Workshop on Online Social Networks, pp. 13–18 (2008)
3. Cha, M., Mislove, A., Gummadi, K.P.: A measurement-driven analysis of information propagation in the Flickr social network. In: WWW 2009: Proceedings of the 18th International Conference on World Wide Web, pp. 721–730 (2009)
4. Ghosh, R., Lerman, K.: Predicting influential users in online social networks. In: Proceedings of KDD Workshop on Social Network Analysis, SNA-KDD (2010)
5. Guo, L., Tan, E., Chen, S., Zhang, X., Zhao, Y.E.: Analyzing patterns of user content generation in online social networks. In: KDD 2009: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 369–378 (2009)
6. Huberman, B., Romero, D., Wu, F.: Social networks that matter: Twitter under the microscope. First Monday 14(1-5) (January 2009)
7. Kendall, M.G.: A new measure of rank correlation. Biometrika 30(1/2), 81–93 (1938)
8. Kim, E., Gilbert, S., Edwards, M.J., Graeff, E.: Detecting sadness in 140 characters: Sentiment analysis and mourning Michael Jackson on Twitter. Tech. Rep. 3, Web Ecology Project (2009)
9. Kwak, H., Lee, C., Park, H., Moon, S.: What is Twitter, a social network or a news media? In: World Wide Web Conference (2010)
10. Lerman, K., Ghosh, R.: Information contagion: An empirical study of the spread of news on Digg and Twitter social networks. In: International AAAI Conference on Weblogs and Social Media (2010)
11. Lerman, K., Hogg, T.: Using a model of social dynamics to predict popularity of news. In: Proceedings of 19th International World Wide Web Conference (2010)
12. Moore, R.J.: New data on Twitter's users and engagement (2010), `http://themetricsystem.rjmetrics.com/2010/01/26/new-data-on-twitters-users-and-engagement/`
13. Spearman, C.: The proof and measurement of association between two things. The American Journal of Psychology 15, 72–101 (1904)
14. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: WSDM 2010: Proceedings of the Third ACM International Conference on Web Search and Data Mining, pp. 261–270 (2010)
15. Xie, B., Kumar, A., Ramaswamy, P., Yang, L.T., Agrawal, S.: Social behavior association and influence in social networks. In: Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing, pp. 434–439 (2009)

SocialWiki: Bring Order to Wiki Systems with Social Context

Haifeng Zhao, Shaozhi Ye, Prantik Bhattacharyya, Jeff Rowe, Ken Gribble, and S. Felix Wu

Department of Computer Science University of California, Davis
One Shields Avenue, Davis, CA 95616
{hfzhao,sye,pbhattacharyya,rowe,gribble,wu}@ucdavis.edu

Abstract. A huge amount of administrative effort is required for large wiki systems to produce and maintain high quality pages with existing naive access control policies. This paper introduces SocialWiki, a prototype wiki system which leverages the power of social networks to automatically manage reputation and trust for wiki users based on the content they contribute and the ratings they receive. SocialWiki also utilizes interests to facilitate collaborative editing. Although a wiki page is visible to everyone, it can only be edited by a group of users who share similar interests and have a certain level of trust with each other. The editing privilege is circulated among these users to prevent/reduce vandalisms and spams, and to encourage user participation by adding social context to the revision process of a wiki page. By presenting the design and implementation of this proof-of-concept system, we show that social context can be used to build an efficient, self-adaptive and robust collaborative editing system.

Keywords: wiki, collaborative editing, social context modeling, trust management, social network applications.

1 Introduction

Wiki systems are widely used since they allow multiple contributors to create and edit a set of documents collaboratively, such as manuals, design documents, and knowledge bases. Among the numerous wiki systems, Wikipedia is probably the best known. With three million articles (in English) and over 150K users contributing to Wikipedia (as of April 2010[1]), it is challenging for administrators to insure the quality of all these pages and edits, especially when dealing with vandalism and unreliable edits. Notable examples include content creation to disrupt the neutrality of a page, hiding and suppression of facts, malicious advertisement placement, and referral to other websites to boost search engine rankings. Access control has been adopted to alleviate the review workload for editors and administrators, but it does not scale with large systems such as Wikipedia. This paper presents a prototype system, SocialWiki, which has

[1] http://en.wikipedia.org/wiki/Wikipedia

L. Bolc, M. Makowski, and A. Wierzbicki (Eds.): SocInfo 2010, LNCS 6430, pp. 232–247, 2010.

built-in reputation and trust management to set edit privileges for each article automatically using social context. The previous contributions from individuals as well as the ratings they have received from others are used to determine who is a good candidate for editing a particular article.

Furthermore, after an article is created, it may take a while before this page is revised by another editor. The absence of active contributor selection slows improvement of document quality and fails to engage the interest of potential contributors. According to the Wall Street Journal [2], in the first quarter of 2009, the English-language Wikipedia had a net loss of more than 49,000 editors, compared to a net loss of 4,900 during the same period a year earlier, despite that the number of Wikipedia visitors grew 20% from Sep. 2008 to Sep. 2009.

SocialWiki allows users to specify what kinds of pages they would like to review/revise and actively finds potential contributors for new pages. This recommendation process is achieved with the help of social context. More specifically, SocialWiki clusters users into communities with shared common interests and then employs trust management and review recommendation to help these communities grow and boost collaborations among users. The trust model is initialized by the social network of wiki users, which can be any existing social network, such as Facebook, and refined as users rate the editing of each other.

To reduce vandalisms and spams, SocialWiki restricts the edit access of a wiki article with *editing certificates* (EC). A user can only edit an article when he/she has an EC of this article. According to specific application scenarios, the maximum number of EC which an article is allowed to have can be set by the system or customized by the article creator. In this paper, one EC for each article is chosen to simplify our discussion. After a user finishes editing, the EC will be circulated by the system to another user according to interests and trust. More specifically, SocialWiki assigns an EC with high probability to users who are interested in editing this article and are trusted by other users. To keep users from holding EC forever without contributing, SocialWiki circulates ECs to other users after a certain period of time. SocialWiki not only finds contributors to improve the quality of pages but also encourages social interactions between them. The communities established through collaborations can be used to work on things other than wiki editing, such as patent design, paper review, and social bookmaking.

The major contribution of this paper rests on a new wiki model which tackles traditional wiki weaknesses by the following measures:

– Proposing an active contributor selection mechanism for large wiki systems to improve document quality and better engage contributors.
– Building an adaptive and automated trust model into wiki systems to identify good contributors, reducing vandalism and spam.
– Utilizing a user interest model to match documents with relevant contributors.
– Developing a probabilistic EC circulation algorithm to combine interest and trust models.

The rest of this paper is organized as follows. Section 2 introduces the architecture of SocialWiki and presents its key building blocks, user interest, trust model and EC circulation. Section 3 demonstrates how SocialWiki employs user feedback to generate high quality articles and prevent spams. Section 4 presents a prototype implementation of SocialWiki; After reviewing related work in Section 5, the paper concludes with Section 6.

2 The Design of SocialWiki

In SocialWiki, an article is associated with one or more editing certificates (EC). A user can only edit this article when he/she has its EC. After a user submits his/her revision, his/her EC will be transferred to another user by SocialWiki such that interested and trustworthy users can have a better chance to review the changes and revise the document. In this section, we present the architecture of SocialWiki and the social context model used to circulate ECs.

2.1 Architecture

Fig. 1 shows the architecture of SocialWiki. The bottom layer of SocialWiki is an existing online social network (OSN). Our prototype chooses `Facebook.com` as its OSN layer.

On top of the OSN layer is the Davis Social Link (DSL) layer [15], which was initially designed to prevent spams by using interest-based trust to control the

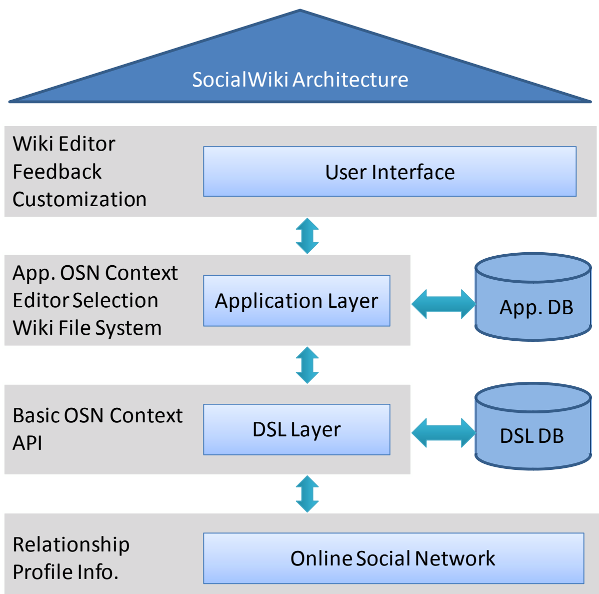


Fig. 1. The Architecture of SocialWiki

messages user receive [3]. DSL includes a database and a set of APIs to provide high level applications with basic trust and user interest computed from OSN.

The application layer consists of three components: application social context, editor selection module and wiki system. Application social context obtains general interest and trust information from the DSL layer and generates application-specific interest and trust models. Circulation module probabilistically assigns an EC to interested and trustworthy users according to their social context and the ratings they have received in previous collaborations. All the articles and their revision history are stored in wiki.

Through the user interface, users can customize their social context and provide ratings to other contributors.

The rest of this section focuses on two key components of the application layer, application social context (Section 2.2 and 2.3) and editor selection model (Section 2.4).

2.2 User Interest

Current wiki systems passively wait for users to contribute, adding new pages or improving existing ones. To accelerate this process, SocialWiki actively recommends articles to users based on their interests. The basic idea is that if Alice contributes to an article, the users who share common interests with Alice have a higher probability than a random user to be interested in this article as well. Therefore after Alice submits her changes, her EC will be transfered to one of these users with high probability.

User interest can be either stated explicitly by users or learned automatically from user profiles. Current SocialWiki asks users to provide a set of keywords to represent their interests.

Keyword-based Interest Description. Each user u_i maintains a list of keywords ℓ_i which describe his/her interested topics. For example, if u_i is interested in soccer, ℓ_i may contain "soccer" and "FIFA". The synsets provided by WordNet [7] are used to match synonyms. A synset is a set of synonyms which are interchangeable in some context without changing the meaning of the proposition in which they are embedded. For each u_i, his/her keyword list is mapped to a list of WordNet synsets. More specifically, we construct a user interest matrix $V(n,m)$, where n is the number of users and m is the number of all synsets. $V[i][j]$ denotes whether u_i is interested in synset s_j, shown as follows.

$$V[i][j] = \begin{cases} 1, u_i \text{ is interested in } s_j \\ 0, u_i \text{ is not interested in } s_j \end{cases} \quad (1)$$

Computing User Similarity. Another matrix $U(n)$ is used to represent user similarity, where $U[i][j]$ is the Pearson Correlation Coefficient (PCC) of user u_i and u_j [11], and n is the number of users. Formally, given the interest matrix $V(n,m)$, let the $\overline{V[i]}$ and $\overline{V[j]}$ denote the mean of the i^{th} and j^{th} rows of V, respectively, then PCC is defined as (2):

$$U[i][j] = \frac{\text{cov}(V_i, V_j)}{\sigma_i \sigma_j} \tag{2}$$

where $\text{cov}(V_i, V_j)$ is the covariance of V_i and V_j, and σ_i and σ_j are the standard deviation of V_i and V_j.

U is a symmetric matrix as $U[i][j] = U[j][i]$. $U[i][j] \in [-1, 1]$. $U[i][j] > 0$ implies a positive association, i.e., in our application, nonzero values of $V[i]$ tend to be associated with nonzero values of $V[j]$ and zero values of $V[i]$ tend to be associated with zero values of $V[j]$. $U[i][j] < 0$ implies a negative or inverse association, i.e., nonzero values of $V[i]$ tend to be associated with zero values of $V[j]$ and zero values of $V[i]$ tend to be associated with nonzero values of $V[j]$.

2.3 Trust Model

As EC is circulated to interested users, however, malicious users such as vandalists and spammers may target certain topics by marking related keywords as their interests. Meanwhile, among the benign users who are interested in a certain topic, some have more expertise than the others. To reduce vandalisms and spams as well as find high quality contributors, SocialWiki employs a trust protocol to assure reliable collaboration, such as the DSL trust protocol in our prototype. Many OSN trust protocols can be used as well.

DSL Trust Protocol The DSL trust protocol is initially proposed in [15]. It propagates trust along the social network and enables communication between two users if there is a trustworthy social path between them. Here we illustrate how DSL works with a simple example. Interested readers please refer to [15] for details.

Fig. 2. An Example Social Path

DSL helps users find trustworthy social paths between a message sender and a recipient, such as the path Alice→Carol→ Bob in Fig. 2. The decimal value between two directly connected users, ranging from 0 to 1, represents the trust between them. In this example, even though Alice does not know Bob well (notice that there is no direct social connection between Alice and Bob), she can trust her friend Carol to deliver her message to the right Bob. If Bob dislikes the message forwarded by Carol, Bob can punish the social path between himself and Alice. So the set of trust values along the social path decrease, i.e., Alice has a smaller probability to reach Bob in the future. Thus, as the receiver, Bob can control who can reach him.

The trust relationship is directed, i.e., the trust value from Alice to Carol may be different from the value from Carol to Alice. DSL reflects real world communications, where people can communicate with each other if they can develop a social path between each other.

SocialWiki's EC circulation bears a similar idea behind the DSL trust communication protocol. If Alice is the current EC holder and she likes the article, she would like to circulate the EC to some trustworthy people who have social paths to her. These trusted EC receivers have the right to decide whether this article is interesting to them. If they do not like it, they can punish the trust values along their social paths to Alice, so that next time they have less probability to receive and revise the articles liked by Alice.

SocialWiki Trust Context. SocialWiki integrates user interest into the DSL trust protocol.

First, there needs to be a social path between the current EC holder and the next EC holder. Second, all the users in this social path need to be interested in one or more synset related to the article. In addition, these synsets must be within a certain perimeter, because some users may only be willing to share their interests with users close to them in the social graph. SocialWiki allows users to set the perimeter with the following three options.

1. Direct friends, i.e., within one hop;
2. Direct friends + their friends, i.e., within two hops;
3. Entire social network.

Each interest keyword is associated with one of these options.

Let us use Fig. 2 as an example again. Alice currently has the EC for a wiki article that is labeled with three synsets $C = \{s_1, s_2, s_3\}$. The Common Interest Set (CIS) of her, Carol and Bob is $\{$Synsets(Alice)$\bigcap$Synsets(Carol)$\bigcap$Synsets(Bob)$\}$ equal to $\{s_1, s_3\}$, which is not an empty set. Assuming that all the three people define s_1 and s_3 with Option (2) or (3), then this social path is considered as a candidate. If Bob defines s_1 and s_3 with Option (1), this social path is not a candidate.

Among all the candidate social paths, we select one to deliver the EC with the following criteria.

- The trust values along the social path need to exceed a certain threshold.
- The larger the CIS is, the larger probability that the social path will be selected.

2.4 Editing Certificate (EC) Circulation

After a current EC holder submits his/her changes, SocialWiki will transfer his/her EC to another user, thus other users get the chance to review/revise this article. This process is called EC circulation, which tries to push articles to high quality contributors automatically and meanwhile prevent/reduce vandalisms and spams. EC circulation is also triggered when a user holds an EC for a

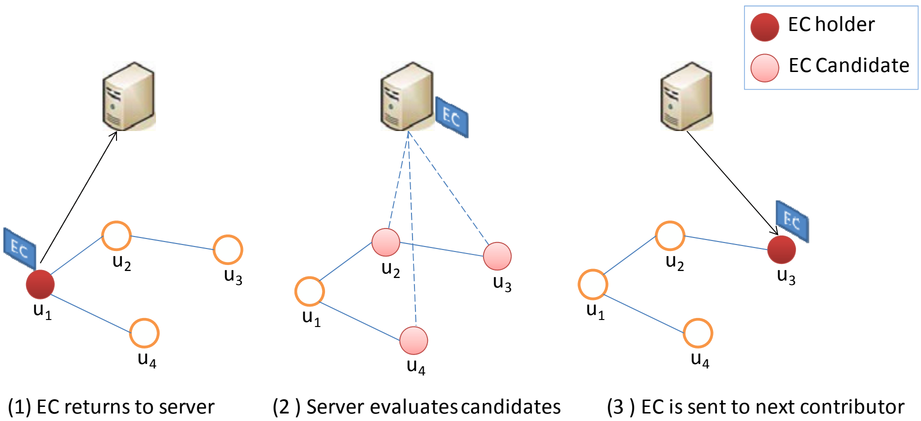


Fig. 3. EC Circulation

certain period of time without contributing to the article. Fig. 3 illustrates the process of EC circulation.

To avoid monopoly, SocialWiki employs a probabilistic model to circulate ECs. More specifically, every user will receive an EC with a probability determined by user interests and trust. Established users with larger CIS and trust have a better chance to get the EC but new users also get the opportunity to contribute and build their trust.

Besides user interests and trust, previous collaboration between the current EC holder and the candidate is also taken into account. The intuition behind this is that if two users have collaborated many times before and have affirmed each other's work, they are likely to work together again.

SocialWiki provides a "dislike" option for users who receives the EC due to either language ambiguity or user interest model's poor understanding of semantics. After the current EC holder specifies that he/she dislikes the article, SocialWiki restarts the EC circulation process by returning to the user who created this article, and uses his/her social context to find the next contributor. Therefore, even if poor contributors were chosen previously, we are still able to find good ones by staring the EC circulation all over again.

Formally, given the current EC holder u_i, SocialWiki computes all social paths starting from another user u_j to u_i (recall that in the DSL trust protocol, the recipient decides if he/she is willing to receive the message instead of the sender selecting which recipient to send the message to). Assume that Q social paths (SP) are found from u_j to u_i. Let $TV[k]$ denote the smallest trust value along the social path $SP[k]$, $cis_{SP}[k]$ denote the CIS (common interest synsets) along $SP[k]$, and syn_{EC} denote the synsets related to the wiki article (as an EC is always tied to an article). Then we compute $TE[i,j]$ as (3):

$$TE[i,j] = \{max_{1 \leq k \leq Q} TV[k]\} \times \frac{|cis_{SP}[k] \bigcap syn_{EC}|}{|syn_{EC}|} \tag{3}$$

In other words, $TE[i,j]$ is the weighted minimum trust value on the most reliable social path from u_j to u_i.

Then Bayesian inference is used to compute the probability that u_j likes an article revised by u_i according to their previous collaborations. Given an article, let $P(i)$ denote the probability that u_i likes the article and $P(i|j)$ be the probability that u_i likes the article given that u_j likes it. If we define related articles as those whose synsets share common words with the given article's synset, $P(i)$ can be calculated by the number of related articles which u_i has received and is interested in. Similarly, $P(i|j)$ is the number of related articles that u_i is interested in, given u_j has received and is interested in these articles. Thus, the probability that u_j likes the article given that u_i likes it can be computed as follows:

$$P(j|i) = \frac{P(i|j) \times P(j)}{P(i)} \tag{4}$$

With the interest similarity matrix, weighted trust, and the likelihood that u_j likes the article, we can compute the priority(RP) that u_j receives EC as follows.

$$RP[i,j] = U[i,j] \times \alpha + TE[i,j] \times \beta + P(j|i) \times \gamma \tag{5}$$

where α, β and γ are weight factors, which can be constants or dynamically adjusted. Current SocialWiki initializes α, β and γ as 0.3, 0.5 and 0.2 respectively, as we believe that trust relationship is the most important. These parameters need to be tuned according to the specific application scenarios, which is beyond the scope of this paper.

Technically the RP given by (5) is not a probability, as it may be larger than 1. Thus we need to normalize it with the sum of the RPs of all u_j, i.e., $CP[j]$. The chosen probability of u_j to receive EC is computed as follows.

$$CP[j] = \frac{RP[i,j]}{\sum_{k=1}^{n} RP[i,k]} \tag{6}$$

3 The System Analysis of SocialWiki

To identify high quality contributors, SocialWiki allows users to provide the following three types of feedbacks.

- Indicating the intention of contribution ("Like", "Neutral", "Dislike"): This adjusts the trust values associated with the social path between two consecutive EC holders, which is used in (3). It is also used as feedback to previous collaborations and further an indicator for future collaborations, as computed in (4).
- Rating previous contributors: As the revision history for an article is available to all users, users may be able to tell whether they like or dislike to collaborate with other users from the content they contribute. Such ratings allow users to specify their preference for potential collaborators.

– Indicating if an article is a spam page: This is a punishment to an article. If a user marks an article as spam, he/she will never receive its EC again.

Supporting Diversity. It is possible that two communities holding opposite opinions on the same issue provide poor feedbacks to each other, i.e., the EC is less likely to be transferred from one community to another. The probabilistic EC circutaltion given by (6) makes it possible for the other community to revise the article, although with a small probability. Once the EC is assigned to a user within the other community, it will have a large probability to stay within that community for a while. As a result, opinions from both sides can be reflected in the article.

Preventing Vandalisms. Traditional wiki systems need huge human effort to fight vandalism. SocialWiki limits editing privileges with EC. With the probabilistic circulation model, the probability for vandalists to receive EC can be rapidly reduced if other users express discontent with their behavior. Therefore, vandalism can be reduced in SocialWiki without requiring a lot of attention from wiki administrators.

Since each EC is routed through a social path, when it is circulated to a vandalist who is connected to a reliable user, the EC might circulate within an unreliable user group if the vandalist cuts off his/her connection with all reliable users right away. However, under the assumption that we have many more reliable users than vandalists in the system, there still exists some probability that the EC will be circulated to reliable users. Once reliable users get the EC, the whole vandalism group will be identified and their reputation will be punished.

Preventing Spams. Spam is usually defined as unsolicited advertising for self-promotion or promotion of a business or organization. There are two types of spams in SocialWiki:

– External link spamming that provides nothing more than a link to the spammer's commercial web site.
– Advertisement articles that sell stuff for a business, product or service, or are individual self-promotion.

The first type of spams commonly exists in traditional wiki systems. In SocialWiki, the probabilistic circulation model restricts it in the same way as treating vandalism. However, the probabilistic circulation model may amplify the destruction of the second type of spams.

To prevent/reduce the second type of spams, SocialWiki employs a reputation system to identify both spam users and spam articles. Given a topic (synset) j, the reputation for user u_i is computed as follows.

$$\mathrm{Repu}[i,j] = 1 - \frac{|F_{spam}(i,j)|}{|F(i,j)|} \tag{7}$$

where $|F_{spam}(i,j)|$ denotes the number of spam ratings for all articles created by u_i in topic j. $|F(i,j)|$ denotes the number of feedbacks for articles created by

u_i in topic j. A small Repu$[i, j]$ means that u_i is considered to be a spammer in topic j by a majority of users. SocialWiki will stop circulating the ECs for articles created by u_i on the topic j and send these articles to administrators to review. Meanwhile, u_i will not be able to create similar articles within a certain period of time t_j.

To compromise this reputation system, a spammer u_i could launch a Sybil attack by creating lots of accounts and providing positive feedbacks to u_i. Such an attack, however, does not work well on SocialWiki as social context is used to compute the trust between users. A Sybil attack only gets u_i high trust within his/her own accounts and does not boost trust along a social path.

For an article a_k, its reputation is the ratio of spam ratings (denoted by $|G_{\text{spam}}(i)|$) among all ratings it receives (denoted by $|G(i)|$), i.e.,

$$\text{Repu}[k, j] = 1 - \frac{|G_{spam}(i)|}{|G(i)|} \tag{8}$$

Once an article a_k's reputation is below a certain threshold, it is marked as spam and the system hides a_k from the readers.

4 MinB: A SocialWiki Prototype

4.1 Generalization

We divide SocialWiki into two components, one to construct social context and circulate ECs, the other to store all articles and manage feedbacks.

We implement the first component as a Facebook application. With Facebook API, SocialWiki can use Facebook as the OSN layer to collect basic user information, such as profiles and friend lists. A social graph is built with the friendship information extracted from Facebook.

To improve user experiences, the first component is provided as a game named **"Message in a Bottle" (MinB)**. Each EC is embodied as a "floating bottle" in the "ocean". Every user has a private beach for the system to deliver bottles. When a bottle arrives at u_i's beach, u_i is able to edit the wiki article bound to this bottle.

The second component is implemented with MediaWiki[2], an open source wiki package written in PHP. We modified MediaWiki to enable access control and user feedback. MediaWiki has an access control table for user groups and their privileges on each article. Restricting an article's editing privilege to some users can be implemented by adding these users to its editing group.

To connect these two components, each bottle in MinB contains a link to an article in our wiki system with credentials to authorize the user to edit it. After submitting his/her changes to our MediaWiki, the user tosses the bottle back into the "ocean" and the system will deliver it to another user. Fig. 4 shows how these two components work with each other.

[2] `http://www.mediawiki.org/wiki/MediaWiki`

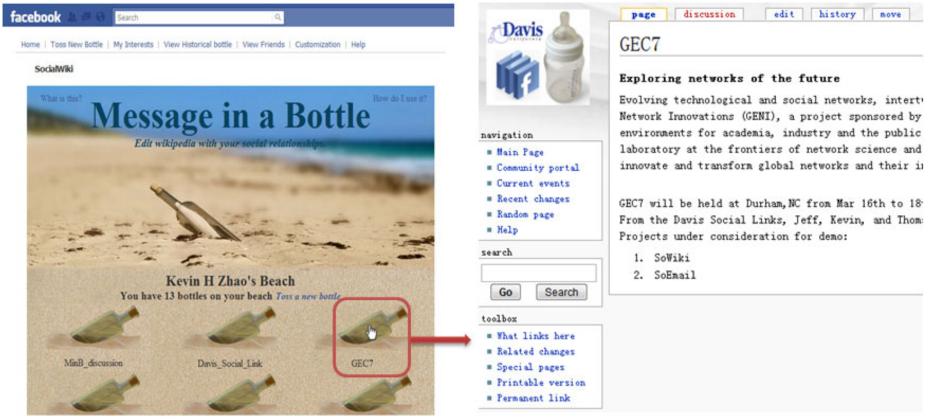


Fig. 4. User Interface

4.2 Workflow

Creating an Article. To create a wiki article, a user u_i first creates a bottle (EC) in the MinB application, by setting some article parameters, including subject, keywords (synsets), expiration date, and maximum holding time. Expiration date defines a time limit after which nobody can edit this article. It is an option for special collaborations, such as paper reviews. Maximum holding time prevents a user from holding a bottle forever, therefore other users get the chance to review/revise the article.

After initializing the bottle, MinB creates a relevant article with access control in our MediaWiki and adds u_i to the article's editing group. Then, MinB provides u_i with the article's link encrypted by u_i's login information so that u_i can access and edit the article through the hyperlink inside the bottle. Fig. 5 shows the process of creating a bottle and initializing its configuration.

After u_i tosses the bottle into the "ocean," MinB sends the bottle to the "beach" of another user u_j (EC circulation) and replaces u_i with u_j in the article's editing group.

Providing User Feedback. Once a bottle (EC) arrives at user u_j's beach, u_j may follow the link inside the bottle and edit the corresponding wiki article. u_j can also provide feedbacks to previous revisions, which updates the trust and EC circulation as we have discussed in Section 3. Providing feedback to historical revisions may directly influence the trust relationship. Feedbacks are anonymous to other users thus true opinions can be reflected. Since u_j's feedback may have influence on the intermediate users between u_j and other contributers, Social-Wiki enables u_j to view their social paths before make a punishment decision. Fig. 6 shows the user feedback interface. In this example, if Kevin does not like Felix's revision, but he does not want to reduce his trust with Prantik, he may choose not to submit a negative feedback.

Fig. 5. Creating a Bottle

Fig. 6. User Feedback Interface

Initializing User Context. When a Facebook user u_i joins the MinB application, two steps are performed to initialize the social context for u_i. The first step is to build social paths for u_i, i.e., import his/her friend list from Facebook and then update the social graph MinB has. The second step is to initialize u_i's interests. u_i needs to provide a set of keywords to represent his/her interests. Then the system groups them according to synsets and updates V and U. Fig. 7 shows the keyword management page in MinB.

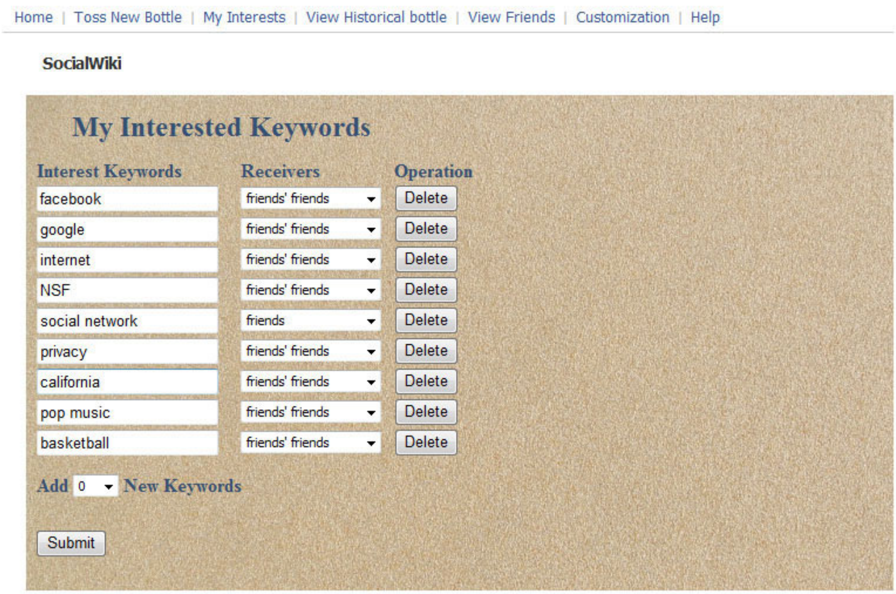


Fig. 7. Keywords Management Interface

4.3 Current Progress

MinB was released in early March 2010. The prototype has user interest, trust management and EC circulation components implemented. A MediaWiki site with access control and user feedback has also been developed. Currently, interest description is based on keywords and we are planning to switch to synsets. Meanwhile, the feedback for previous contributors are not considered in EC circulation yet. Although we still have some components to improve, there are already 40 early users joining MinB and more than 80 wiki articles have been created. As this project is still in its infancy stage, we expect to report an evaluation after collecting more user data and feedbacks.

5 Related Work

Many efforts have been made to integrate user interests and trust into social networks, especially in recommender systems.

One of the most commonly used algorithms to estimate user interest context is the k-nearest neighborhood approach. By calculating the Pearson correlation [13] between a user u_i and all other users in the system, the top k nearest ones are selected as the set of similar users of u_i. Such algorithms are known as collaborative filtering (CF) [13].

However, the generic collaborative filtering method alone exhibits poor scalability to large number of users and items [12]. A pre-filtering or scalable neighborhood formation (such as a subgraph or community) is preferred to assist CF in real world systems. Sinha and Swearingen [14] found that people prefer receiving recommendations from people they know and trust. Recently, approaches incorporating trust models into online recommender systems have been launched [5,6,8]. Olsson [10] proposed an architecture combining trust, collaborative filtering and content-based filtering. Trust has become a supplementary or even a replacement of existing filtering mechanisms.

A trust-based approach has also been investigated in wiki systems. Adler et al. [1] proposed a content-driven reputation system for Wikipedia authors. In their system, authors gain reputation when their edits are preserved by subsequent authors, and lose reputation when their edits are reverted quickly. Thus, author reputation is computed based on content evolution only and user-to-user comments or ratings are not used.

SuggestBot developed by Dan [4] performs intelligent task routing (matching people with tasks) in Wikipedia by using broadly applicable strategies including text analysis, collaborative filtering, and hyperlink.

The difference between SocialWiki and previous work lies in an enhanced trust model considering strong social context.

– Trust propagates along social paths, which ensures online collaboration with a large number of reliable contributors.
– The trust in SocialWiki is associated with topics and provides flexibilities to specific collaborations.

The name “SocialWiki” was used before in [9] and by Socialwiki.org[3], both of which are quite different from our work. [9] targets collaborative workflow design and activity management. It presents a workflow system that is based on wiki and changes the wiki context from public to organizational. Socialwiki.org provides a collaborative and open platform with MediaWiki for social media discussions. Both of them employ the traditional wiki system as a platform for their applications, rather than remodeling the wiki system.

6 Conclusions

This paper presents a collaborative editing system based on social interest and trust. Social context, such as user interest and trust, is used to select high quality and trustworthy contributors. We discuss how to construct user interest and trust with the information provided by online social networks. We also describe methods to prevent spams and vandalisms.

A prototype of SocialWiki, MinB, has just been released on Facebook. Lots of improvement is needed to make it more user friendly, efficient and scalable. Currently we are working in several directions. Getting more users to test our system and give us feedback is certainly important. At the same time, we plan to

[3] http://socialwiki.org/

use simulations to expose the problems and limitations of our current system, for example, how fast the system detects and prevents spam under various attacks and how the weight factors in (6) can be tuned accordingly. Meanwhile, we are considering to evaluate how soon valuable editors will get the chance to edit an article. We plan to allow users to request an EC based on their reputation, historical contribution, and social context. We also consider to bring in more valuable editors from other online social networks such as Linkedin.

Acknowledgement

The authors would like to thank Thomas Tran for his valuable contribution and feedback during system development. This work was supported by the National Science Foundation Future Internet Design program under Grant No. 0832202, MURI under Army Research Office and Network Science CTA under Army Research Laboratory.

References

1. Thomas Adler, B., de Alfaro, L.: A content-driven reputation system for the Wikipedia. In: WWW 2007: Proceedings of the 16th International Conference on World Wide Web, pp. 261–270 (2007)
2. Angwin, J., Fowler, G.: Volunteers log off as Wikipedia ages (2009)
3. Banks, L., Ye, S., Huang, Y., Wu, S.F.: Davis social links: integrating social networks with internet routing. In: LSAD 2007: Proceedings of the 2007 Workshop on Large Scale Attack Defense (2007)
4. Cosley, D.: Suggestbot: Using intelligent task routing to help people find work in wikipedia. In: Proceedings of the 12th International Conference on Intelligent user Interfaces, pp. 32–41 (2007)
5. Guha, R.: Open rating systems, Tech. report, Stanford Knowledge Systems Laboratory (2003)
6. Kinateder, M., Rothermel, K.: Architecture and algorithms for a distributed reputation system. In: Nixon, P., Terzis, S. (eds.) iTrust 2003. LNCS, vol. 2692, pp. 1–16. Springer, Heidelberg (2003)
7. Miller, G.A.: Wordnet: A lexical database for english. Communications of the ACM 38, 39–41 (1995)
8. Montaner, M., Lopez, B., de la Rosa, J.L.: Opinion-based filtering through trust. In: Klusch, M., Ossowski, S., Shehory, O. (eds.) CIA 2002. LNCS (LNAI), vol. 2446, p. 164. Springer, Heidelberg (2002)
9. Neumann, G., Erol, S.: From a social wiki to a social workflow system. In: Proceedings of the first Workshop on BPM and Social Software, pp. 698–708 (2008)
10. Olsson, T.: Decentralized social filtering based on trust. In: Proceedings of the AAAI 1998 Recommender Systems Workshop (1998)
11. Rodgers, J.L., Nicewander, A.W.: Thirteen ways to look at the correlation coefficient. The American Statistician 42, 59–66 (1988)
12. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW 2001: Proceedings of the 10th International Conference on World Wide Web (2001)

13. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.: Application of dimensionality reduction in recommender systems – a case study. In: ACM WebKDD Workshop (2000)
14. Sinha, R., Swearingen, K.: Comparing recommendations made by online systems and friends. In: Proceedings of the DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries (2001)
15. Tran, T., Chan, K., Ye, S., Bhattacharyya, P., Garg, A., Lu, X., Wu, S.F.: Design and implementation of davis social links osn kernel. In: Liu, B., Bestavros, A., Du, D.-Z., Wang, J. (eds.) WASA 2009. LNCS, vol. 5682, pp. 527–540. Springer, Heidelberg (2009)

Author Index