Journal of Universal Computer Science, vol. 11, no. 11 (2005), 1780-1791
submitted: 1/9/05, accepted: 1/10/05, appeared: 28/11/05

Visualizing Recommendation Flow on Social Network

Jason J. Jung
(INRIA Rhône-Alpes, France
j2jung@intelligent.pe.kr)

Abstract: In contrast with centralized recommender systems, social recommendation algorithm is applied to the item rating data on social networks. Meaningful recommendation can be uncovered by the topology of social network as well as the similarity between users. More importantly, this information becomes propagated into the users in the estimated same groups. As the goal of this paper, we propose a novel method for visual explanation of the recommender system on social network. For experiments, we simulate the recommendation flow by using the *MovieLens* dataset on a social network constructed with FOAF.
Key Words: Reputation, social network, visualizing information flow
Category: J.4, H.5

1 Introduction

As the web environment has been popular, recommender systems have been emerged as one of the representative solutions for information retrieval and sharing tasks. These technologies especially benefit the eletronic commerce communities for advertising products and increasing their profits. However, there are several drawbacks in computing recommendations on the centralized client-server environment. The first problem is *cold starting*. The basic idea of recommendation schemes is based on stochastic processes of the rating datasets given by a group of users. This rating dataset is organized as a very sparse matrix (items×users) in which the user's opinion and uncertainties are included [Herlocker et al. 2000]. To recommend useful information to users from this matrix, the system has to learn the user preference in advance [Rashid et al. 1997]. Secondly, recent studies like [Ramakrishnan et al. 2001] and [Kim et al. 2003] mention the *privacy issues* in recommender systems. Most users want the private information like preferences to be preserved. These make the results of predicted recommendation unreliable and, more importantly, hardly understandable.

In this paper, we propose a visual explanation approach to the predicted recommendation on social network where each participant is assumed to be *explicitly* linked with the other users such as family members and like-minded friends. Once the personal agent system learns the preferences of the corresponding user, it can classify users with respect to their own preferences on real time, and also, it can recommend items that users are interested in to the rest of them in the same group on peer-to-peer network. Furthermore, we can increase the robustness of recommendation using RFR algorithm that helps users to decide whether the recommendation is reliable or not by considering other user's rating results in the same group. Finally, we can get satisfactory result of robustness [O'Mahony et al. 2004].

The outline of the paper is organized as follows. Section 2 describes basic concept of recommendation and some well-known recommender systems. Section 3 proposes the main idea of a recommender system based on FOAF. In Section 4, we address the implementation issues related to the visual explanation of this system and show experimental results. Finally, in Section 5, we draw conclusions and future work.

2 Related work

A recommender system can be regarded as a part of personalization technique [Resnick and Varian 1997], [Schafer et al. 1999]. This system is mainly used in personal information systems and E-Commerce to recommend products to customers. Tapestry is one of the earliest systems of collaborative filtering-based recommender system [Goldberg et al. 1992]. The *GroupLens* system developed by research group in University of Minnesota is based on rating for collaborative filtering of netnews in order to help people find relevant articles in the huge stream of available articles [Rensnick et al. 1994]. This system is also called ratings-based automated recommender system, because filtering is executed by the explicit opinions of people from a close-knit community [Sarwar et al. 2001]. There is an additional project, *MovieLens*, which is web-based re-commender system for movies, based on GroupLens technology. They also offer an experimental data source and a framework for studying user interface issues related to recommender systems [Sarwar et al. 2000]. Also there have been many studies to enhance privacy issue [Avesani and Massa 2002], [Canny 2002], [Sarwar et al. 2001]. In [Canny 2002], Canny tries to solve privacy problem with security technologies such as key sharing, encryption and decryption. Especially, Riedl suggested Local Profile Model to protect personal profiles by separating personal profiles from centralized server to each user's personal computer [Sarwar et al. 2001]. Although it can protect draining of personal profile, but the system is working on centralized server, so it is still weak in fraud rating and malicious server attack.

3 Distributed recommender systems

In this paper, we propose a novel approach to recommender system based FOAF on distributed environment such as peer-to-peer computing platform, in order to deal with some drawbacks of traditional recommender systems. A user can be recommended based on analyzing the rating dataset generated by other users.

This information becomes propagated along to the channels linking both nodes in FOAF network.

Recommendation process consists of mainly three tasks; i) collecting relevant feedback, ii) grouping like-minded users, and iii) propagating recommendation. Before making a recommendation to users, the users must be grouped according to their preferences. If user grouping is completed, recommender system can recognize which users should be received.

3.1 Relevance feedback on social network

Recommender system has to firstly collect each user's preference that is represented by FOAF document format. FOAF is a project derived from RDFweb and a document format. When a user publishes a document for some information with FOAF, machines are able to make use of that information, and especially users can make their own friends' network by interlinking FOAF. It is very extensible. We can find or connect to anyone who is in FOAF network by tracing links. So far, there have been several applications trying to use this document format for sharing user's profile [FOAF]. Especially, this contributes to build human network by interlinked file on peer-to-peer environment.

In order to extract more accurate preferences from each user, we employed a probabilistic method, as shown in Equation 1. We denote a set of users and movie genre as $U = \{u_1, u_2, \ldots, u_i\}$ and $G = \{g_1, g_2, \ldots, g_k\}$. A set of movies that is rated by a user u_i is represented as $M = \{m_1, m_2, \ldots, m_j\}$. The function $f(P_{ik})$ extracts i-th user's preference about a k-th genre.

$$f(P_{ik}) = \sum_{j=1}^{n} \left(\frac{g_k(m_j)}{\mu} \times R_{ij} \right) \tag{1}$$

where the variable μ is given by

$$\mu = \sum_{m=1}^{n} C_m(u_i, m_j) \tag{2}$$

where C_m is a set of genres included in a movie m_j and R_{ij} means a rating value of i-th user about a movie m_j. More importantly, μ means total count of genres that included in movies that are rated by a user u_i.

3.2 Measuring the similarity between users

In contrast with the centralized system, this recommender system is working under distributed computing environment. Therefor, we cannot group users easily. In order to enhance this shortcoming, there have been many studies clustering users in real-time [Nejdl et al. 2003], [Krishnamurthy and Wang 2000], [Krishnamurthy et al.]. In case of our system, user grouping can be regarded as assorting objects for the recommendation. As we mentioned earlier, each user's profile is distributed. That is, there is no server to manage all users' profile such as preferences. Hence, user grouping must be processed at the same time with the recommendation. Although it takes more time to group users compared with centralized system, we can overcome some drawbacks of the system. The user grouping (object selection procedure) is processed on real-time, so it can make distributed system more dynamically. For example, when a new friend is added on a user or when a user's preferences are changed, we don't need to update any part of the system because of the real-time manner. And the place where users' information is gathering does not exist. Therefore, distributed recommender system is less dependent on environment than centralized system. Also it guarantees users' privacy because all of each user's information is only kept on themselves [Canny 2002]. Grouping of users who will be recommended must be done before recommendation. For the grouping of users, we used cosine-based similarity that is the most common way to compute the similarity between two users. As we show in Equation 3, the similarity between A and B can be calculated with each vectors of A and B.

$$Sim(A,B) = \cos(\overrightarrow{A}, \overrightarrow{B}) = \frac{\overrightarrow{A} \cdot \overrightarrow{B}}{|\overrightarrow{A}| \times |\overrightarrow{B}|} \tag{3}$$

By using this similarity method, we can group users who are in the same preferences. As shown in the following Equation 4, the grouping is calculated by the function $f(G_i)$

$$f(G_i) = \sum_{i+1}^{n} S\left(\sum_{i}^{n-1} S(u_i)\right) \tag{4}$$

where $S(a)$ is indicating a set of users who are similar to the user a. The threshold value is used to control the size of group. The optimal threshold for similarity $T_{optimal}$ is obtained by using "F1-measure" function, It will be discussed in Section 4.

3.3 Propagating recommendation

The system determines whether m_i is good enough to recommend to the other users in same group or not. Once the system decides to recommend m_i to other users, the recommendation is only applied to level n. We can define the 'level' as the depth of friends to be known by the user directly. On the process of a recommendation, a rating can be regarded as an essential factor for the quality of the system. Although a user rates an item with good score, we cannot say the result is reliable, because it is just a user's subjective opinion. Therefore, for the quality of recommendation, we need to aggregate the same group users' rating.

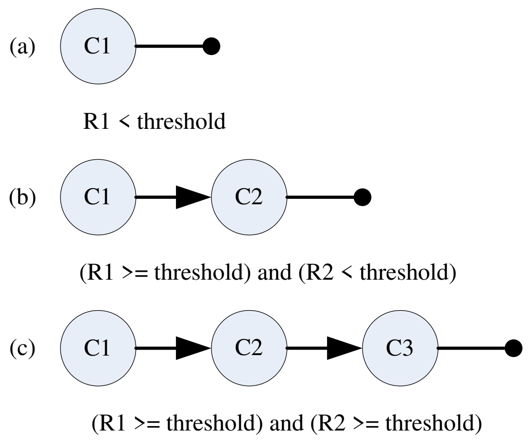

Figure 1: Three cases of recommendation flow

So we focused on the level. We assume that a level is a particular group of users who has known by a user directly. For example, if a user *Jason* knows *John* and *Tom*, these two users are on the same level, so we call this level as $level_1$. We do not care about *John* and *Tom*'s friends for the $level_1$. They will be in $level_2$.

$$\begin{aligned} level_n &= FriendOf(u_i) \\ level_{n+1} &= FriendOf(level_n) \end{aligned} \tag{5}$$

Recommended user C_1 will take three pieces of information from recommender in $level_{n-1}$. They will be sent by FOAF document involved the name of an item m_j, rating result and the number of rater. The rating result is both a factor that can influence a recommended user to make their decision and a criterion that can compare with threshold for recommendation. Once a user rates an item m_j, the rating result is also sent to members of $level_{n+1}$.

As an example, we can consider three cases as shown in Figure 1. In Figure 1(a), R_1 is a rating result from user C_1. If R_1 is less than *threshold*, no recommendation is being made. In Figure 1(b) and Figure 1(c), there can be two actions from the user C_2 according to their actions; i) C_2 rates m_j and ii) C_2 does not rate m_j. In case of i), the system needs to update the rating result. But the case of ii), the system only needs to recommend another level. And also to update rating result on real-time, we suggested RFR (Recommend-Feedback-Re-recommend) system, as shown in Figure 2.

If U_1 on $level_1$ rates an item m_j that satisfies with *threshold*, the system recommends the m_j with rating result to C_1, C_2 and C_n on $level_2$. Due to the independency between users, each user on the same level cannot recognize

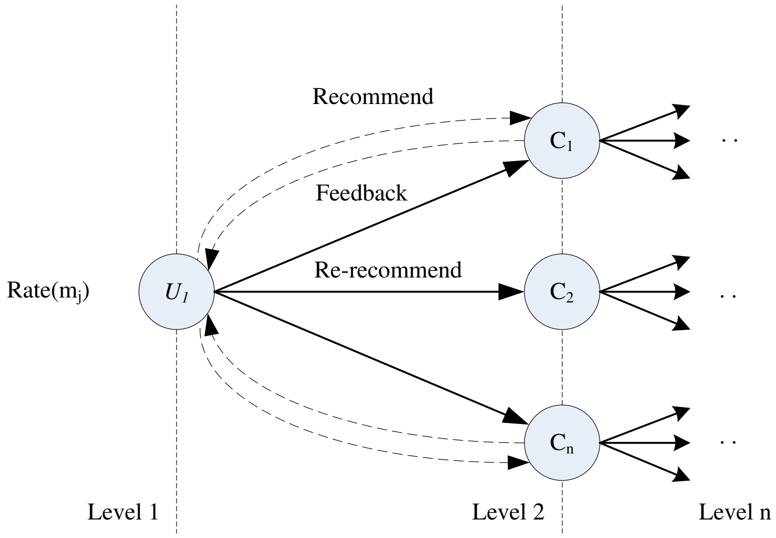

Figure 2: RFR (Recommend-Feedback-Re-recommend) algorithm

the other's rating result. In order to make possible for this peer-to-peer network updating rating result summarized from all users who have made rating, feedback process is needed. By this process, U_1 will take rating results from users on $level_2$. Feedback is a set of rating result from users on one level, $Feedback = \{Rate(C_1), \ldots, Rate(C_n)\}$. The first recommender U_1 can calculate the updated rating result after feedback. Then, it will be re-recommended to C_1, C_2 and C_n, and these processes are applied to all levels, as shown in Equation 6.

$$Re - recommend = \frac{Rate(User_i) + Feedback}{N + 1} \tag{6}$$

Therefore, as the level grows the rating result can be more reputable. Also the system shows how many users are participating in rating by numerical or graphical method. This helps users to decide whether a recommended item is really worthy for them or not.

4 Experimentation

We assume that our system can give us several advantages. First, there is no need for place to aggregate users' information such as personal details and preferences. So we analyze performance of proposed system using "F1-measure" and mean absolute error (MAE). Second, peer-to-peer system is stronger about malicious action than centralized system. We also examined performance of robustness against malicious action as level increases. Experiments were carried out on Pentium 1 GHz with 256MB RAM, run-ning Microsoft Windows 2000 Server. The recommender system was implemented by Microsoft Visual Basic 6.0, Active Server Page (ASP) and IIS 5.0.

4.1 Implementation

Especially, with the emergence of semantic web environment providing various services such as information integration and standardization [Davies et al. 2003] many studies have been developed using RDF (Resource Description Framework) and XML (eXtensible Markup Language). FOAF ('Friend Of A Friend') from RDFweb [RDFWeb] is a RDF project establishing a social network to make users possible to link themselves as a friend by using RDF-based documents describing the corresponding user's profiles. Due to the extensibility of the FOAF-based distributed environment, it is possible to aggregate information from other users who have the same interest. User preference is asserted as a property of the FOAF document format by learning user's behaviors.

In order to visualize the recommendation flow, we implemented a social network system based on FOAF. Figure 3 shows a snapshot of interface of this system. This user is linked with five friends. The initial level is set two. Red lines are indicating the recommendation flow for the corresponding user.

4.2 Experimental results

We used *MovieLens* data sets to experiment suggesting recommendation system. *MovieLens* is a web-based research recommender system. The data set contains 1000000 anonymous ratings of approximately 3900 movies made by 6040 users who joined the site in 2000. For the experiment, we selected 450 users to use only about 66926 rating dataset [Sarwar et al. 2000]. By users' rating data, we can extract all users' preferences. After we got the preferences, we made a matrix of user-user similarity that has 450 rows and 450 columns. And then, we also made the Most Similar Users (MSU) set by cosine-based similarity method.

In this experiment, first of all, we employ "F1-measure" widely used in information retrieval community to obtain the optimal threshold $T_{optimal}$ for user grouping procedure [Yang and Liu 1999]. The measurement "F1-measure" is calculated by *precision* and *recall*. As shown in the following equations, we defined *precision* and *recall* as the ratio of hit set size to the prediction set size and the ratio of hit set size to the MSU set size, respectively.

$$precision = \frac{\text{the size of hit set}}{\text{the size of prediction}} \tag{7}$$

$$recall = \frac{\text{the size of hit set}}{\text{the size of MSU set}} \tag{8}$$

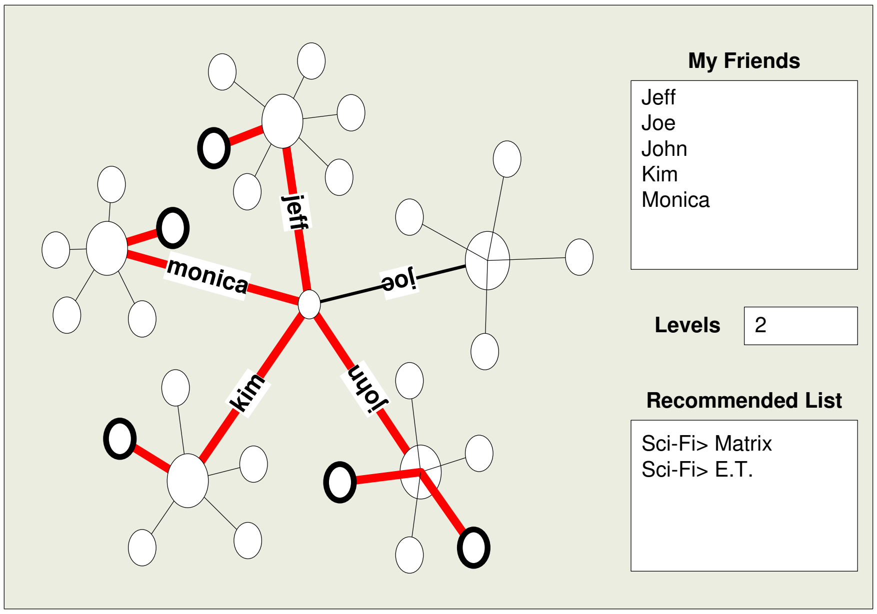

Figure 3: A snapshot of user interface

For example, if a user u_1 recommends an item m_j to u_2 and u_3, we can define prediction set as $prediction(m_j) = \{u_1 \to u_2, u_1 \to u_3\}$. And also, for the measurement of the system's accuracy, we used mean absolute error (MAE)

$$MAE = \frac{\sum |MSU - prediction|}{N_p} \tag{9}$$

where N_p is indicating the number of prediction.

For the $T_{optimal}$ of user grouping procedure, we used "F1-measure" with 124 items. After we made FOAF network, recommended each items to users who are randomly selected. In order to obtain the optimal threshold, we carried the experiment out while changing threshold from 0.1 to 1. Figure 4 shows the F1-measure as chaging the threshold value. In this figure, when we set threshold to 0.7, it shows the best result. It means this leads to the best performance for user grouping (object selection). When any users are recommended on FOAF network, the users propagate the recommendation to their friends who have similarity that is bigger than 0.7 of threshold.

Then, we used MAE to show how the accuracy changes as level increase. In Figure 5, we tested changing MAE through 6 levels. When we use a data set without any fraud rating, MAE is continuously getting lower than former level,

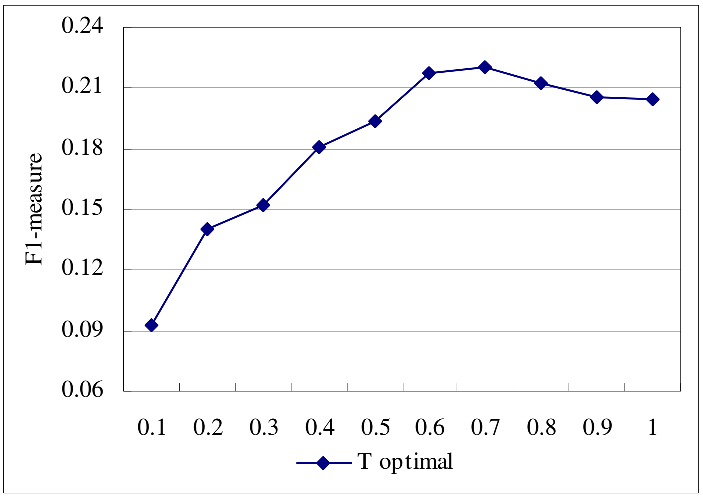

Figure 4: Variation of the F1-measure as changing threshold, $T_{optimal}$

$level_{n-1}$. Since number of users is exponentially increased as a level increase, the system can aggregate as many users' opinion. It means that even there are some of malicious ratings, users can take more accurate recommendation compare with centralized system as level increase.

Furthermore, we showed performance of the system in Figure 6 using "F1-measure" to show robustness. As illustrated in graph, we involved some of malicious ratings in terms of 10%, 20% and 30%. Generally, even when malicious ratings are included, system's performance is increased as level increase as we already show in Figure 5. When there is no fraud rating, in Figure 6, the performance evaluated by "F1-measure" is improved about 9.17%, and in case of 10%, 20% and 30% of fraud ratings are included, the performance is improved 8.02%, 11.35% and 15.69%, respectively. No matter how many fraud ratings are included in a rating data set, as level increase, the performance is improved 11.68% in terms of the three cases in average.

5 Concluding remarks and future work

As a growing demand of peer-to-peer environment, we have claimed a collaborative filtering system based on information propagation in distributed social network environment. The proposed system has shown two major improvements. First, it is able to collect the other user's rating information simply by *RFR* algorithm. Therefore, this helps users to make a better decision effectively. Second,

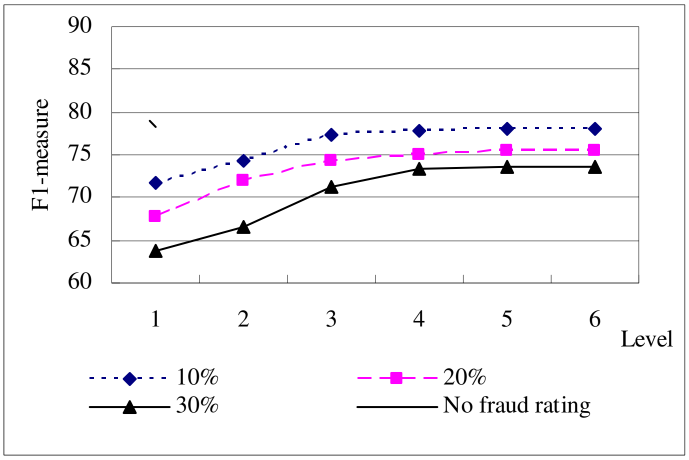

Figure 5: Variant of MAE as *level* is increasing

it was robuster than the centralized recommender system using clustering algorithms such as k-NN. When a malicious user tries to give any wrong information to other users or system "intentioanlly or not", in fact, there has been no way to protect these social phenomena. But the proposed system has an ability of self-refinement. As shown in Figure 6, this malicious effect can be filtered out with their ratings. As the level is getting increasing, users on FOAF network can be provided with the recommendation filtered from their friends. On the peer-to-peer environment by heterogeneous users, for high quality of recommendation, it needs as many users to rate items for recommendation as possible. Actually, this requirement is same as the conventional recommendation systems.

As a future work, we are planning to develop 3D visualization of our system for higher intuition of users. More seriously, we need to gather more reliable testing dataset and apply to the real-world applications.

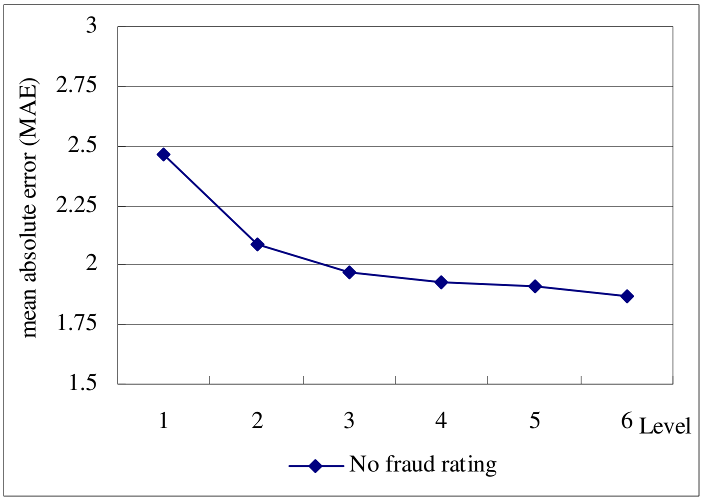

Figure 6: System performance with malicious ratings

References

[Avesani and Massa 2002] Avesani, P., Massa, P.: "Recommendation and Personalization From Item to Collection, from Client-Server to Peer-to-peer New Frontiers for Case-based Personalization"; Proc. Workshop Case Based Reasoning and Personalization in 6^{th} European Conf. Case-Based Reasoning, (2002).

[Canny 2002] Canny, J.: "Collaborative filtering with privacy"; Proc. IEEE Symposium on Research in Security and Privacy (2002), 45-57.

[Davies et al. 2003] Davies, J., Fensel, D., Harmelen, F.A.: "Towards the Semantic Web: Ontology-driven Knowledge Management"; John Wiley & Sons, Ltd (2003)

[FOAF] http://rdfweb.org/foaf/.

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

[Herlocker et al. 2000] Herlocker, J. L., Konstan, J. A., Riedl, J.: "Explaining Collaborative Filtering Recommendations"; Proc. ACM Conference on Computer Supported Cooperative Work (2000), 241-250.

[Jung et al. 2005] Jung, J.J.: "Collaborative web browsing based on semantic extraction of user interests with bookmarks"; Journal of Universal Computer Science, 11, 2 (2005), 213-228.

[Kim et al. 2003] Kim, H.-J., Jung, J.J., Jo, G.-S.: "Conceptual framework for recommendation system based on distributed user ratings"; Lecture Notes in Computer Science, 3032, (2003), 115-122.

[Krishnamurthy and Wang 2000] Krishnamurthy, B., Wang, J.: "On network-aware clustering of web clients"; Proc. ACM SIGCOMM 2000 Conf. Applications, Technologies, Architectures, and Protocols for Computer Communication (2000), 97-110.

[Krishnamurthy et al.] Krishnamurthy, B., Wang, J., Xie, Y.: "Early measurements of a cluster-based architecture for peer-to-peer systems"; Proc. 1^{st} ACM SIGCOMM

Workshop on Internet Measurement (2001), 105-109.
[Nejdl et al. 2003] Nejdl, W., Wolpers, M., Siberski, W., Schmitz, C., Schlosser, M., Brunkhorst, I., Loser, A.: "Super-Peer-Based Routing and Clustering Strategies for RDF-Based Peer-To-Peer Networks"; Proc. 12^{th} Int. World Wide Web Conf. (2003), 536-543.
[O'Mahony et al. 2004] O'Mahony, M.P., Hurley, N.J., Kushmerick, N., Silvestre, G.C.M.: "Collaborative Recommendation: A Robustness Analysis"; ACM Trans. Internet Technology, 4, 4 (2004), 344-377.
[Ramakrishnan et al. 2001] Ramakrishnan, N., Keller, B.J., Mirza, B.J., Grama, A.Y., Karypis, G.: "Privacy Risks in Recommender Systems"; IEEE Internet Computing, 5, 6 (2001), 54-62.
[Rashid et al. 1997] Rashid, A.M., Albert, I., Cosley, D., Lam, S.K., McNee, S.M., Konstan, J.A., Riedl, J.: "Getting to Know you: Learning New User Preferences in Recommender Systems"; Proc. 7^{th} Int. Conf. Intelligent User Interfaces (2002), 127-134.
[RDFWeb] http://rdfweb.org/.
[Resnick and Varian 1997] Resnick, P. and Varian, H.R.: "Recommender Systems"; Communications of the ACM, 40, 3 (1997), 56-58.
[Rensnick et al. 1994] Rensnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: "GroupLens : An open architecture for collaborative filtering of Netnews"; Proc. ACM Conf. Computer Supported Cooperative Work (1994), 175-186.
[Sarwar et al. 2000] Sarwar, B.M., Karypis, G., Konstan, J., Riedl, J.: "Analysis of recommendation algorithms for E-commerce"; Proc. ACM 2000 Conf. Electronic Commerce (2000), 158-167.
[Sarwar et al. 2001] Sarwar, B.M., Karypis, G., Konstan, J., Riedl, J.: "Item-Based Collaborative Filtering Recommendation Algorithms"; Proc. 10^{th} Int. World Wide Web Conf. (WWW10) (2001), 285-295.
[Sarwar et al. 2001] Sarwar, B.M., Konstan, J.A., Riedl, J.: "Distributed Recommender Systems: New Opportu-nities for Internet Commerce"; Internet Commerce and Software Agents: Cases, Technologies and Opportunities, Chap. 20, Idea Group Pubs (2001), 372-393.
[Schafer et al. 1999] Schafer, J.B., Konstan, J.A., Riedl, J.: "Recommender Systems in E-Commerce"; Proc. ACM Conf. Electronic Commerce (1999), 158-166.
[Yang and Liu 1999] Yang, Y., Liu, X.: "A Re-examination of Text Categorization Methods"; Proc. 22^{nd} ACM SIGIR Conf. Research and Development in Information Retrieval (1999), 42-49.
[Yimam and Kobsa 2000] Yimam, D., Kobsa, A.: "Centralization vs. Decentralization Issues in Internet-based Knowledge management System: Experiences from Expert Recommender Systems"; Proc. Workshop Organizational and Technical Issues in the Tension Between Centralized and Decentralized Applications on the Internet (2000)