Personalized Decentralized Communication

Tomas Olsson, Andreas Rasmusson and Sverker Janson

Swedish Institute of Computer Science
Box 1263
SE-164 29 Kista, Sweden
{tol, ara, sverker}@sics.se

Introduction

Search engines, portals and topic-centered web sites are all attempts to create more or less personalized web-services. However, no single service can in general fulfill all needs of a particular user, so users have to search and maintain personal profiles at several locations. We propose an architecture where each person has his own information management environment where all personalization is made locally. Information is exchanged with other's if it's of mutual interest that the information is published or received. We assume that users are self-interested, but that there is some overlap in their interests.

Our recent work has focused on decentralized dissemination of information, specifically what we call decentralized recommender systems. We are investigating the behavior of such systems and have also done some preliminary work on the users' information environment.

The Virtual Community Library

People working with information read, write and sort "documents" in a way that reflects their specific tasks and needs (c.f. the folder hierarchy on a personal computer).

From the work put down by the users in organizing their own documents, a user's personal agent can construct a user profile. This profile can be used to personalize interactions with other information sources, e.g. personalized ranking of search-engine results, personalized document recommendations, "intelligent" help in maintaining the document collection, personalized views of external information sources, subscriptions to and change notifications for external resources.

The restriction that we only make use of the information the user has put in her personal library is central to the architecture. A problem in most systems relying on sharing information is that they depend on users doing actions that are not directly beneficent to themselves (e.g. rating a read news-article only saves other people's time). Accessing or maintaining the personal library is assumed to have to be done by the user anyway. Since most of the information representing a user-stays with the user (except information that can be inferred from the agent's communicative acts), all profile information is kept and updated at one place and less private information is disclosed.

We call an individually controlled information collection a

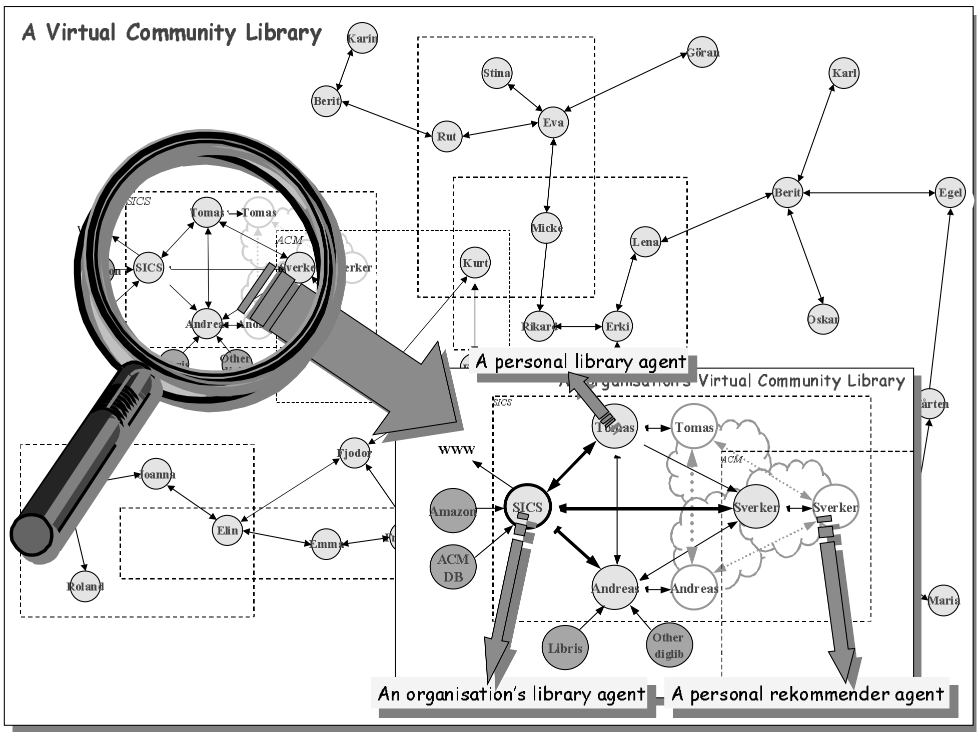

Figure 1: A Virtual Community Library where the light grey nodes are the users' personal library agents or organisations' library agents (thick lines). Dark grey nodes are web-sites and web-services while the unfilled circles are personal recommendation agents. The arrows indicate information exchange.

“Personal Library”. It is the fundamental building block in our architecture and every idiosyncratic categorization of documents (e.g. the documents on a web-server) is a Personal Library.

A Virtual Community Library (VCL) is modeled as a multiagent system consisting of Personal Library agents representing users and organizations (Rasmusson, Olsson and Hansen 1998), see figure 1. Other agents can be associated to the personal library, e.g. recommender agents that help the user finding relevant information. The virtual community is not static; users will enter, leave and change interests over time. The VCL requires means to publish and get access to information in the personal libraries. How useful the VCL will be to the users depends crucially on the degree to which the users can benefit from the work put down by the other users in maintaining their personal libraries.

In this architecture, no agent is necessarily considered more complete or authoritative. Still, large (institutional and organizational) libraries will most likely be quite influential (by virtue of them being used a lot) in how documents are categorized and described.

Prototype Personal Library

The Personal Library should be integrated into the user’s everyday document management environment. Although not completely unavoidable, it is preferable to minimize the constraints on what tools the users can use to manage their information. We have tested the applicability of intercepting an application level protocol (IMAP, www.imap.org) to transparently extract information about user behavior. We implemented a prototype personal library agent monitoring the user’s email management (Rasmusson 1999). Email is often sorted according the user’s personal view of what is relevant. Also, actions such as replying to and forwarding mails indicate social relationships with other people.

Using this information it should be possible to construct services based on the work-context (e.g. sort the results of a query based on the contents of the currently active folder) and also to create views of external services, e.g. by showing them as virtual mailboxes in the user’s folder hierarchy. Yet, only some smaller proof of concept services have been implemented, e.g. an application where users vote on the relevance of a document to a set of known people. Remembrance Agent (Rhodes 1996) and MailCat (Segal and Kephart 1999) are two related systems that unobtrusively monitor the user’s actions to suggest potentially relevant information sources (Remembrance Agent) and to suggest on mailboxes to sort emails into (MailCat). Remembrance Agent is currently implemented as an emacs-mode and hence depends on emacs. In a similar way the personal library agent implemented here depends on the use of IMAP. MailCat is built into Lotus Notes’ mail reader and can hence access the GUI whereas the personal library agent is not tied to any particular mail-reader, but is more restricted in how the user interaction can be implemented.

The Decentralized Recommender System

Each personal library agent may be associated with a recommender agent whose task is to automate the dissemination of information among the agents. A recommender agent proposes potentially relevant documents to its user based on a profile of the user’s interests. The recommender system we study mimics the word-of-mouth method for spreading information and is based on Collaborative Filtering (Shardanand and Maes 1995). The recommender agents send information to each other and, in the process, learn which others to trust for relevant information (Olsson 1998). When the user publishes a document, the agent will forward the message to those of the agents that it believes are most interested in the document. This, in turn will give the recipient agents information about the sender’s interests, allowing them to send relevant documents to that agent in the future. Since the agents prefer communicating with likeminded agents, clusters of likeminded agents will form, hence shortening the path a document must travel to reach the interested recipients.

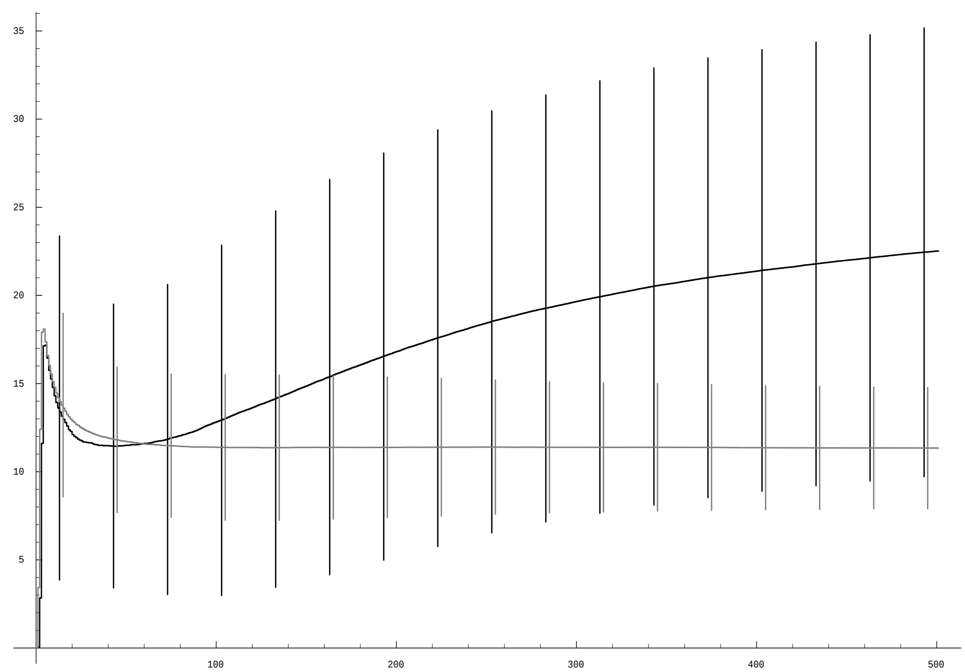

Figure 2: The mean recall for the users (upper line) and the received mean share of all documents for the recommender agents (lower line) (both in percent per time step) with ±1.0 standard deviation.

Yenta (Foner 1997) and ACORN (Marsh and Masrour 1997) are two related agent based systems. In Yenta the agents compare their user profiles with each other in order to find other users with similar interests. Yenta assumes that it is non-problematic to directly compare user profiles. In ACORN, the documents or search queries are the "smart" entities that try to locate interested users or matching documents, respectively. In our system, the agents can have non-comparable user profiles, because they share their profiles by sending document references and thereby get documents as well.

Preliminary simulation results

To analyze the behavior of the recommendation network we have implemented a simulation of a population of users with different interests. We ran 100 simulations with 100 users, each having 3 out of 30 interests.

We are currently analyzing the results. The initial results indicate that we get the expected clustering effect. In figure 2, it can be see that the mean recall[1] for the users is higher than the received mean share[2] for the recommender agents. If the documents were randomly sent to the agents the mean recall should be similar to the received mean share, but the clustering seems to make it non-random.

Research Issues

The aim with this research is do find guidelines for how to design this kind of decentralized systems and to increase the understanding of what trade-offs and assumptions that must be made.

We have not yet addressed the issues of how changing interests and differing quality of information may affect the resulting communication patterns. Under what conditions will subgroups dissolve or form as response to changing interests in the community? Will a decrease in the quality of information disperse previously functioning communities, similarly to the way newsgroups tend to be abandoned when the traffic increases too much?

Insights from these simulations may be used to estimate the potential gains and understand the impact of changes in the recommender system algorithms, e.g. in the group-formation strategy or in different reward mechanisms.

Acknowledgements

This work was funded by FIOL (www.fiol.siti.se).

[1] The *recall* is the number of received relevant documents divided by the total number of relevant documents in the system for a user.

[2] The *received mean share* is the total number of received documents for a recommender agent divided by the total number of documents in the system.

References

Foner, L. N. 1997. Yenta: A Multi-Agent, Referral-Based Matchmaking System. In *Proceedings of The First International Conference on Autonomous Agents*, 301-307. ACM Press.

Marsh, S. and Masrour, Y. 1997. Agent Augmented Community Information — The ACORN Architecture. In *Proceedings CASCON'97, Meeting of Minds*.

Olsson, T. 1998. Decentralized Social Filtering based on Trust. In Recommender Systems Workshop Papers. Technical Report WS-98-08, 84-88. AAAI Press.

Rasmusson, A. and Olsson, T and Hansen P. 1998. A Virtual Community Library: SICS Digital Library Infrastructure Project. Research and Advanced Technology for Digital Libraries. Second European Conference, ECDL'98, Heraklion, Crete, 19-23 September, Proceedings. Lecture Notes in Computer Science, Vol 1513. 677-678. Springer.

Rasmusson, A. 1999. Components of a Personal Digital Library - monitoring email management. Draft manuscript.

Rhodes, B. J. 1996. Remembrance Agent - A continuously running automated information retrieval system. In Proceedings of The First International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents, 487-495. The Practical Application Company, Ltd.

Shardanand, U. and Maes, P. 1995. Social Information Filtering: Algorithms for Automating "Word of Mouth". In *Proceeding of the Conference on Human Factors and Computing Systems*, 210-217. ACM Press.

Segal, R. and Kephart, J. 1999. MailCat: An Intelligent Assistant for Organizing E-Mail. In *Proceedings of the Third International Conference on Autonomous Agents*, 276-282 ACM Press.