Wi-Fi Walkman: A wireless handhold that shares and recommends music on peer-to-peer networks

Jun Wang, Marcel J.T. Reinders, Johan Pouwelse, Reginald L. Lagendijk
Faculty of EWI, Delft University of Technology,
Mekelweg 4, 2628 CD Delft, The Netherlands
{j.wang, m.j.t.reinders, j.a.pouwelse, r.l.lagendijk}@ewi.tudelft.nl

ABSTRACT

The Wi-Fi walkman is a mobile multimedia application that we developed to investigate the technological and usability aspects of human-computer interaction with personalized, intelligent and context-aware wearable devices in peer-to-peer wireless environments such as the future home, office, or university campuses. It is a small handheld device with a wireless link that contains music content. Users carry their own walkman around and listen to music. All this music content is distributed in the peer-to-peer network and is shared using ad-hoc networking. The walkman naturally interacts with the users and users' interest with each other in a peer-to-peer environment. Without annoying interactions, it can learn the users' music interest/taste and consequently provide personalized music recommendation according to the current situated context and user's interest.

Keywords: Peer-to-peer networks, Recommendation, Personalization, User's interest

1. INTRODUCTION

Recently, with the rapid progress in information processing, communications, and storage technologies, the amount of information that we deal with in our daily lives has been rapidly increased and even more the types of information have been changed from homogeneous (textual) data only to heterogeneous data (audio, video, image, etc). We enjoy the entertainment and convenience brought to us by a variety of sources coming from, amongst other, digital TV, mp3 player, digital still image/video camera, but we are hampered to access this data due to its sheer amount.

Not only the availability of the sufficient types of the information is changing, but also the way people consume information is also changing. Peer-to-peer and ad-hoc networks, as new network topology, become a new way for people to distribute, exchange, and consume resources from their local storage devices in many different locations, such as the future home, office, or university campuses. There are two significant advantages of peer-to-peer and ad-hoc networks: 1) the replicas of the content among peers increases the content availability, 2) for the exchange of information, no requirements of centralized storage and management from third parties is necessary which makes these networks to have very low costs. Recently, those attributes attract a large body of people in the internet domain. For instance, the internet peer-to-peer networks, such as Freenet[3] and Gnutella[4], make it possible that a large number of people have access to each other's shared files. Furthermore, we believe that, in the recent future, the wireless communication technology will make those peer-to-peer networks wireless and exist in any place, and at any time.

In ad-hoc network environments, the volume of information is increasing far more quickly than our ability to digest it. The traditional textual keywords-based information retrieval approaches [5,6,7,8] can not longer be used as filter mechanism since they suffer from three major problems. Firstly, the transition from textual data to heterogeneous data requires large amount of textual Meta data on the one hand. It is practically intractable to ask people to provide content as well as associated Meta data at the same time. On the other hand, automatic content analysis on the non-textual data is far from being efficient to get the Meta data that we need. Secondly, keywords are not semantically expressive enough to enable a seamless search, i.e. people hardly issue a textual query when they can not exactly express what they are

Embedded Processors for Multimedia and Communications II, edited by Subramania Sudharsanan,
V. Michael Bove, Jr., Sethuraman Panchanathan, Proc. of SPIE-IS&T Electronic Imaging,
SPIE Vol. 5683 © 2005 SPIE and IS&T · 0277-786X/05/$15

looking for. Thirdly, in mobile environments, the user interface is constrained and consequently does not permit complex interactions between users and their handheld devices.

Automatically assisting the user to acquire information and/or services that fits his/her interests is a non-trivial problem. To achieve this, it is necessary to increase the ability of computers to interpret the user's interests and select relevant information on the user's behalf. To this regard, the research on information filtering has aroused to filter out, refine and systematically represent the relevant information. One of the solutions for overcoming the information overload is to provide personalized suggestions based on a history of a user's likes and dislikes.

The *Wi-Fi* walkman that we developed is a case study that investigates the technological and usability aspects of human-computer interaction with personalized, intelligent and context-aware wearable devices in ad-hoc wireless environments such as the future home, office, or university campuses. It is a small handheld device with a wireless link that contains music content in the environment or from the user. Users carry their own *Wi-Fi* walkman around and listen to the music content. All this music content can be shared using mobile ad-hoc networking. The *Wi-Fi* walkman is situated in a peer-to-peer environment and naturally interacts with the users. Without annoying interactions with users, it can learn the users' music taste and consequently provide personalized music resources to fit the user's interest according to the user's current situated context.

2. RELATED WORK

Internet based peer-to-peer networks increase rapidly and it has given a large number of people the possibility of sharing resources in their local storage devices [1,2]. Recently, sharing resources in wireless networks has received some attention. In [2], the TunA system allows users to "tune in" to other nearby TunA music players and listen to what someone else is listening to. Another system, SoundPryer [1] allows drivers to jointly listen to music shared between cars on the road. Interestingly, these two applications show that the upcoming technologies start to take care about their social impact on everyday life, i.e. they bring people together that have been socially separated by the technologies for the last decades (such as TV, Internet, portable music player, etc.) Clearly, those technologies [1,2,3,4] are different from the traditional technologies in that they encourage people to make social interactions such as sharing and exchanging information. However, those applications are implemented far away from being called intelligent devices which aims to provide personalized services on user's behalf. Differently, we present here a system that has the ability to react to the user's interests and select relevant information on the user's behalf accordingly.

One of the most promising widely implemented and familiar technologies to understand user's interest is *collaborative filtering* [9,10,11,12]. Collaborative filtering based approaches utilize the correlations (commonalities) between users on the basis of their ratings to predict and recommend items which have the highest correlations to the user's rated items that together represent the user's interest. Here, we show how to use collaborative filtering to create a personalized music delivery system in a peer-to-peer environment.

3. THE *WI-FI* WALKMAN

The prototype of the *Wi-Fi* walkman running on a Sharp Zaurus PDA is shown in Figure 1.

The *Wi-Fi* walkman allows to exchange music files (MP3 formatted) in a mobile network in a personalized way. The music files are stored on the local storage device (e.g. hard disk, or fresh memory) of a *Wi-Fi* walkman and can be accessed through the *Wi-Fi* mobile network. The key issue in the *Wi-Fi* walkman is how to locate music files that will be interested in the user. To this regard, music recommendation is implemented as a user oriented music file filter to help the user to find relevant or desired music files according to the current situated context and learned user interest.

Fig. 1 The *Wi-Fi* walkman prototype running on a Sharp Zaurus PDA.

3.1 Peers and play-lists

In this section, we will define our research problems. Consider the case that users share music content in a peer-to-peer network. Each peer represents a *Wi-Fi* walkman used by a particular user. Let's define the set of peers as:

$$P_i, i = \{1, ..., M\} \tag{1}$$

where M is the number of the peers currently online in the peer-to-peer network. That means they can be located and accessed with the sufficient bandwidth. Since the peers (*Wi-Fi* walkman) and users exist in pairs, we will use the term *peer* and *user* interchangeablely.

The music content in the network is defined as a set of items, denoted by the set I. Each item has a specific physical location, i.e.

$$I = \{I^{i,j} \mid i = \{1, ..., M\}; j = \{1, ..., N_i\}\} \tag{2}$$

where N_i is the number of items physically located in the local storage device by the peer P_i. $I^{i,j}$ denotes the *j*th item own by user P_i. The set of items own by peer P_q is denoted as:

$$I_q = \{I^{i,j} \mid i = q; j = \{1, ..., N_q\}\} \tag{3}$$

Users will retrieve music content regarding to their own interests. At a particular time, a user, however, when time passes, will have a particular interest. The interest can be obtained either explicitly or implicitly. For instance, it could be explicitly obtained by asking users to rate items. Alternatively, this can also be implicitly indicated by the music items that the user is playing. In our *Wi-Fi* walkman, we use the user's music play-list to indicate the user's music interest. Formally, we use a vector $V_q = \{v_q^{i,j}\}, i = \{1, ..., M\}; j = \{1, ..., N_i\}$ to represent the play-list of the user P_q, where the element $v_q^{i,j} = 1$, if user P_q played the item $I^{i,j}$, otherwise $v_q^{i,j} = 0$.

We would like to note that generally the interest of the user will change over time. It in fact depends on the current context. Therefore, the play-list (representing the current users' interest) should ideally be dependent on the time also, i.e. $V_q \rightarrow V_q(t)$.

We utilize a sliding time window to forget the old music items users have played, as shown in Fig. 2. By doing so, the system focuses on the current user's interest.

$$\ldots I_i^{10}(t-T-1)\,,\, I_i^{11}\underbrace{(t-T),\ldots,I_i^{12}(t-2)\,,\,I_i^{13}(t-1)}_{\text{Time window}}\,,\,I_i^{14}(t) \qquad \text{Play sequence}$$

Fig. 2 Time window for forgetting.

The current recommender system is implemented by using the collaborative filtering technique. Collaborative filtering utilizes the correlations (commonalities) between users on the basis of their ratings (in this case, the play-lists of the users) to predict and recommend music items which have the highest correlations to the user's preference.

The accuracy of the collaborative filtering directly relies on the number of users, who provide their ratings. In mobile networks, the density of peers may vary strongly depending on the local situation. For instance, on the bus, there are only a dozen of people while in the airport there are thousands of peoples. Depending on the current density of peers, we perform recommendation by two different approaches, namely the flooding model and the client/server model.

3.2 Flooding Model

When the density of peers is large (i.e. thousands of users) and the play-lists from those users are enough to obtain a good recommendation, we use the flooding approach to find the correlations between users.

By using the correlation [12], the similarity between the play-lists V_q and V_p of two users is calculated as follows:

$$Sim(V_q,V_p)=\frac{\sum_{i,j}^{M,N_i}(v_q^{i,j}-\bar{v}_q)(v_p^{i,j}-\bar{v}_p)}{\sqrt{\sum_{i,j}^{M,N_i}(v_q^{i,j}-\bar{v}_q)\sum_{i,j}^{M,N_i}(v_p^{i,j}-\bar{v}_p)}}\,, \tag{4}$$

where $\bar{v}_q$ and $\bar{v}_p$ are the mean rating of the user P_q and P_p that are used for removing the bias, respectively.

$$\bar{v}_q=\frac{1}{\sum_i^M N_i}\sum_i^M\sum_j^{N_i}v_q^{i,j}\,,\quad \bar{v}_p=\frac{1}{\sum_i^M N_i}\sum_i^M\sum_j^{N_i}v_p^{i,j} \tag{5}$$

The distance measurement between a music item $I^{i,j}$, not known to user P_q and the play-list from user P_q can be calculated as the weighted average rating [12,13], as follows:

$$d(I^{i,j},V_q)=\bar{v}_q+k\sum_{\{V_p|V_p\in N_q,sim(V_q,V_p)>T\}}sim(V_q,V_p)(v_p^{i,j}-\bar{v}_p) \tag{6}$$

where k is a normalization constant. In the flooding model, the play-list V_q of the user P_q is broadcasted to all its neighbors P_p in order to determine the recommendation for that user. The neighboring peers check the similarity (using in Eq.(5)) between the received play-list and their own play-list. They decrease the TTL (Time to Live) field of the broadcasted play-list and then pass it to their neighboring peers again until the TTL count reaches 0. We use set N_q to denote all the neighboring peers that the querying play-list V_q can reach. If one of the neighboring peers has a play-list that has a similarity to the broadcasted play-list that is higher than T, then the items in the play-list of the neighbor P_p (including the locations) are sent back to the peer P_q that posed the query V_q. We use I_q^* to denote the set of these returned items. Finally all items I_q^* received by the querying peer are ranked according to the distance measurement (Eq.(6)) and consequently the top-N ranked items are recommended to the user (Eq.(7)).

$$\mathrm{Rec}_q^N = TopN\{\ rank\{d(I^{i,j}, V_q) \mid I^{i,j} \in I_q^*, I^{i,j} \notin I_q\}\ \} \tag{7}$$

3.3 Client/Server Model

When the density of the peers is small and consequently the play-lists (rating) from those users are not enough to obtain a good recommendation, we have to access a predefined rating database and use the database to calculate the recommendation. In this model, we assume the peer has a chance to access a server which has a rating database. The rating database stores the play-lists of all the users in the networks.

Fig.3 illustrates the procedure of obtaining the recommended play-list. In order to reduce the computational complexity, we apply the item-based recommendation algorithm proposed in [14] to calculate the recommendations.

In item-based recommendation, each music item can be represented by who has played it. More formally, each item $I^{i,j}$ can be represented by a vector $U^{i,j}$, where its element $u_q^{i,j} = 1$ if the item $I^{i,j}$ has been played by the peer P_q and zero otherwise.

Item-based recommendation is then performed by exploring the correlations between the items rather than the correlations between users. Recommendations are created by finding items that are similar to other items that the user according to:

$$sim(I^{i,j}, I^{i',j'}) = \frac{Freq(I^{i,j}, I^{i',j'})}{Freq(I^{i,j}) \times Freq(I^{i',j'})} \tag{8}$$

where $Freq(I^{i,j})$ is the number of times that item $I^{i,j}$ is in any of the play-lists. $Freq(I^{i,j}, I^{i',j'})$ is the number of times that item $I^{i,j}$ and $I^{i',j'}$ are in the same play-list.

Due to the fact that the item-to-item matrix is relatively static, it is possible to compute this matrix offline, which extremely reduces the computational demands. That it, by applying Eq. (8), for each item $I^{i,j}$, its topN similar items can be obtained offline and it is denoted as I_q^{TopN}.

When the play-list V_q of user P_q send to the server, the recommendation then is calculated according to the following equation:

$$\mathrm{Rec}_q^N = TopN\{\ rank\{sim(I^{i,j}, I^{i',j'}) \mid I^{i',j'} \in I_{TopN}^{i,j}, I^{i,j} \notin I_q; I^{i,j} = 1 \cap I^{i,j} \in V_q\}\ \} \tag{9}$$

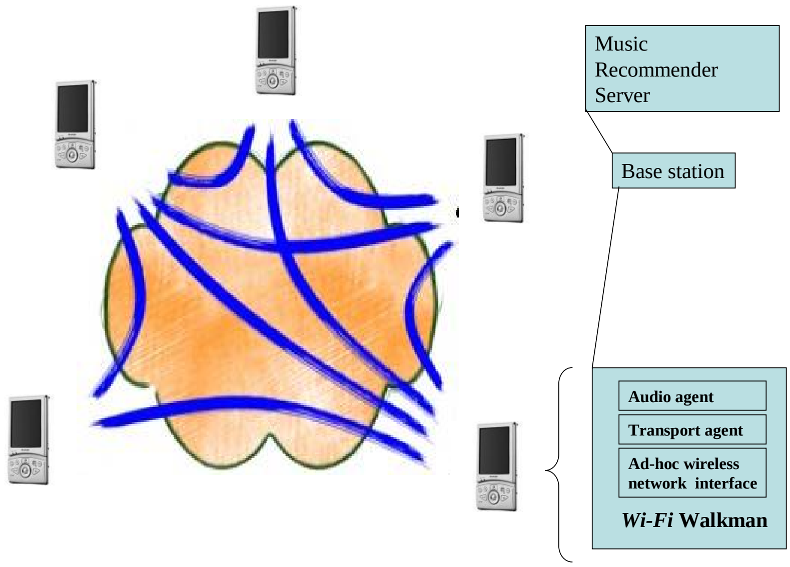

Fig. 3 Illustration of the *Wi-Fi* Walkman in client/server model.

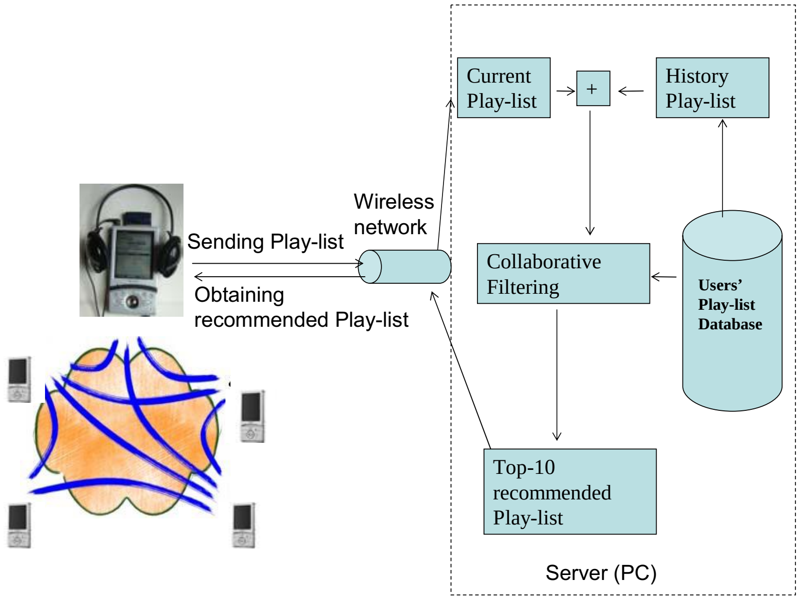

Fig. 4 Recommendation in the client/server model.

3.4 Implementation details

The *Wi-Fi* walkman is implemented on the Sharp Zaurus PDA, see Fig. 1, by using C++. It is running on an ad-hoc wireless network. It features audio playback, audio storage, audio recommendation, and ad-hoc wireless connectivity for audio exchange.

The *Wi-Fi* walkman itself contains an audio agent, a transport agent, and a wireless interface shown in Fig. 3. The audio agent is responsible for the communication with the recommendation services, manages the MP3 files on the storage devices (e.g. a fresh card), and selects which MP3 to play. The transport agent uses the wireless ad-hoc network to communicate with other transport agents and enables the sharing of the music files. Due to the dynamic nature of an ad-hoc network, the transport agents must keep track of the other walkmans around them. The enhanced ad-hoc wireless interface also informs the transport agent of new walkmans and walkmans that can no longer be reached.

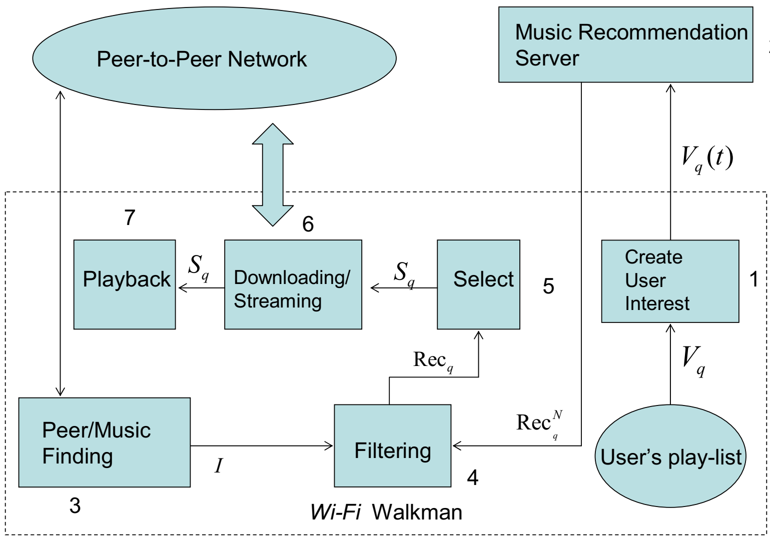

Fig. 5 System Diagram of the *Wi-Fi* Walkman application.

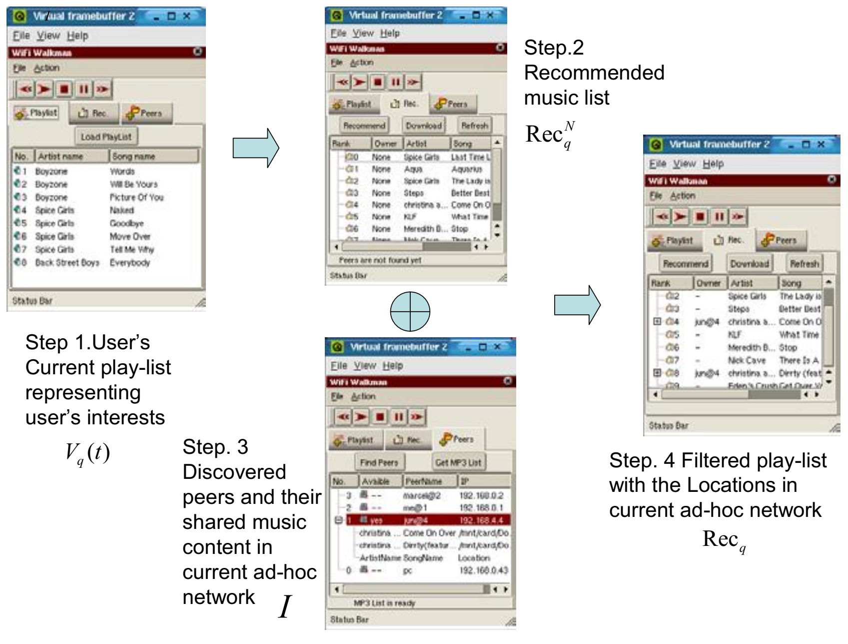

Fig. 6 Snap-shots of the *Wi-Fi* Walkman prototype.

The recommendation is implemented in the server part. We utilize a dataset of the *AudioScrobbler* community [15] as our play-list dataset. Currently this dataset has 857.020 tracks and 4.175.146 playback actions. The interaction between each peer and the server is illustrated in Fig. 4.

Snap-shots of the *Wi-Fi* walkman application are shown in Fig. 6. The procedure to obtain the music files that fit the user's interest is illustrated in Fig. 5 and each step is described as follows:

Wi-Fi_Walkman()
Begin

1. Create $V_q(t)$ to represent the user's current interest from the play-list by utilizing a time window.
2. Get recommendation Rec_q^N from server
3. Find online peers and obtain the music item list *I* (resources) from those peers.
4. Filter the music list *I* to get the recommended list Rec_q by the top*N* recommended items Rec_q^N.

   $\mathrm{Rec}_q = I \cap \mathrm{Rec}_q^N$
5. Select the downloading/streaming items by users through GUIs.

   $\mathrm{S_q} \subset \mathrm{Rec}_q$
6. Locate the recommended items $\mathrm{S_q}$ and download/stream them
7. Playback the obtained items $\mathrm{S_q}$

End.

4. CONCLUSIONS

In this paper, we introduce a new wireless application called *Wi-Fi* walkman. Without bothering users for any annoying keywords input, the *Wi-Fi* walkman can steer user's music interest and recommend appropriate music in the peer-to-peer networks.

In our framework, user's interest is inferred by the play-list of a user. Based on collaborative filtering methods, system recommends music to users both in the flooding model and the client/server model depending on the local density of the peers.

REFERENCES

[1] M. Östergren. "Sound Pryer Field Trials: Learning About Adding Value to Driving", In *Proceedings of the workshop Designing for ubicomp in the wild: Methods for exploring the design of mobile and ubiquitous services(MUM'2003)*, 2003.

[2] A. Bassoli, C. Cullinan, J. Moore, and S. Agamanolis. "TunA : a mobile music experience to foster local interactions(poster)", In *Proceedings of the Fifth International Conference on Ubiquitous Computing (UbiComp 2003)*, Seattle, 12-15 October 2003.

[3] FreeNet. http://freenet.sourceforge.net

[4] Gnutella. http://www.gnutella.com

[5] O. D. Gnawali. "A keyword set search system for peer-to-peer networks", Master's thesis, Massachusetts Institute of Technology, June 2002.

[6] J. Li, B. Loo, J. Hellerstein, F. Kaashoek, D. Karger, and R. Morris. "On the feasibility of peer-to-peer web indexing and search", In *Proceedings of the 2nd Int. Workshop on Peer-to-Peer Systems*, 2003.

[7] B. Cooper and H. Garcia-Molina. "Studying search networks with SIL", In *Proc. of IPTPS*, 2003.

[8] B. Bhattacharjee, S. Chawathe, V. Gopalakrishnan, P. Keleher, and B. Silaghi. "Efficient peer-to-peer searches using result-catching", In *Proceedings of the 2nd Int. Workshop on Peer-to-Peer Systems*, 2003.

[9] U. Shardanand and P. Maes, 1995. “Social Information Filtering: Algorithms for Automating ‘Word of Mouth’ ”, In *Proceedings of the Conference on Human Factors in Computing Systems (CHI95),* 210-217, Denver, Co, ACM Press.
[10] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. “Item-based collaborative filtering recommendation algorithms”, In *Proceedings of the WWW10 Conference*, pages 285-- 295, Hong Kong, 2001.
[11] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, and J. Riedl. “GroupLens: Applying Collaborative Filtering to Usenet News”, *Communications of the ACM*, 40(3), pp. 77-87, 1997.
[12] J. S. Breese, D. Heckerman, and C. Kadie. “Empirical analysis of predictive algorithms for collaborative filtering”. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98).* G. F. Cooper, and S. Moral, Eds. Morgan-Kaufmann, San Francisco, Calif., 43-52. 1998.
[13] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. “GroupLens: applying collaborative filtering to usenet news”. *Communications of ACM,* 40. 77-87. 1997.
[14] K. George. “Evaluation of item-based top-N recommendation algorithms”, Technical Report #00-046, Dept. of C.S., Univ. Of Minnesota, 1999.
[15] Audioscrobbler Community. http://www.audioscrobbler.com/