Mining Relational Context-aware Graph for Rater Identification

Yue Shi
Multimedia Information Retrieval Lab
Delft University of Technology
y.shi@tudelft.nl

Martha Larson
Multimedia Information Retrieval Lab
Delft University of Technology
m.a.larson@tudelft.nl

Alan Hanjalic
Multimedia Information Retrieval Lab
Delft University of Technology
a.hanjalic@tudelft.nl

ABSTRACT

This paper studies the rater identification problem in recommender systems. We propose to approach rater identification by fusing influence from various factors that relate to users. The rater likelihood is modeled from a probabilistic point of view as the conditional probability of a user given sources of contextual information, resulting in an aggregation model that fuses all the available information sources pertaining to a particular user. The result is a relational context-aware graph. A random walk with restart is used to calculate the proximity scores over this graph, which are used to identify raters. We compare our approach with several baselines in a set of experiments performed on the CAMRa2011 challenge dataset. The results demonstrate the superiority of our approach in predicting the identity of a rater who rated a particular movie within a given household.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering*

General Terms

Algorithms, Experimentation, Performance

Keywords

context-aware recommendation, graph, random walk with restart, rater identification, recommender systems

1. INTRODUCTION

The conventional purpose of recommender systems is to provide users with personalized recommendations, such as movies and music [1]. However, the availability of contextual information in recommender systems has pushed forward new challenges in the community. Thanks to the rich knowledge of the context about users and products, various objectives can be taken into account and even realized, such as examples from the Context-aware Movie Recommendation Challenge 2010[1] [15], i.e., to recommend movies for a particular time [6][14], to recommend movies for a particular emotional status [16], etc. Recently, more and more scenarios in recommender systems involve recommending items to a group of users rather than an individual user [3, 4, 5, 9]. The key issue in this research is twofold: to discover the common preference among a group of users and to discover the personal preference of individual users. Specifically addressing the two aspects, Context-aware movie recommendation challenge 2011[2] (CAMRa2011) is organized with two tracks, referred to "Group recommendation" track and "Rater identification" track. In the first track, the objective is to improve recommendation quality for individual households (consisting of a few users), while in the second track, the objective is to identify raters from households. Our work in this paper studies research issues in the second track. Rater identification is important for group recommendation, since it could help to distinguish common preference from personal preference, thus, have great potential for improving group recommendation models. In addition, we also envision that rater identification could be useful for targeted advertising in online sites.

We consider that the main difficulty for rater identification is the fact that various factors could exert impact on raters. For example in movie recommendation scenario, we may predict a user (among several candidate users) to be a rater, because she is most likely to be the one in favor of the movie that has been rated, or because she is most likely to be the one to use rating 5 (which has been assigned to the movie), or because she is most likely to be the one who likes to watch movies on Monday (when the movie was rated), etc. It is desired that multiple factors can be all taken into account for rater identification. In this paper, we specifically address this issue from probabilistic point of view. We model the rater likelihood as an aggregation of all the sources of contextual information relating to the user. The approach of random walk with restart is used to mine proximity scores of different factors from a relational context-aware graph. Using the CAMRa2011 challenge dataset, we demonstrate the proposed algorithm achieves a lower error rate for rater identification than other alternatives.

Our contributions in this paper can be summarized as:

- We propose a novel algorithm for rater identification,

[1]http://www.dai-labor.de/camra2010/challenge
[2]http://2011.camrachallenge.com/

modeling rater likelihood by taking into account all the factors under a given context.

- We propose to construct a relational context-aware graph, which is interpretable, and can be used to mine proximity of different factors to a user.
- We experimentally demonstrate that the proposed algorithm could be effective for rater identification, as large improvement is achieved over other baseline approaches.

The paper is organized as follows. After presenting an overview of the "Rater identification" challenge and our understanding of it in Section 2, we discuss the related previous work and its relationship to our own work in Section 3. In Section 4, we present the detail of the proposed algorithm. Our experimental evaluation is reported Section 5, and Section 6 summarizes the key aspects of our work in this paper and briefly discusses the possible directions for future work.

2. PROBLEM STATEMENT

As mentioned before, this paper focuses on the "rater identification" track in CAMRa2011 challenge. The problem of this track can be stated as: *Given the users in a household (the candidate users), and given the rating that was assigned to a given movie, and also given the timestamp in which the rating was made, identify which user (one and only one) from the household is the rater.* In other words, if we know there was a user from a specific household who rated a specific movie with a specific rating at a specific time, how to identify which user in the household is the rater.

We summarize our understanding of this particular problem in two points.

1. We consider that a proper solution for this problem is to address "*which user from the household is more likely to have rated the given movie*" rather than to address "*which user's rating for the given movie could be the given rating*". There might be the case that more than one user from the same household rates a movie with a similar rating. In this case, the two users can not be distinguished, even if a good rating prediction algorithm is available. Note that most rating prediction algorithms would still have more than 10% prediction error, as can be seen from Netflix competition [12]. For this reason, our solution presented in this paper follows the first direction, and no rating prediction algorithms are involved.
2. We consider that the rater can be identified by taking into account multiple factors. For examples, users in the same household may have different preference for a movie, or they may have different preference for the time to watch movies, or they may have different preference to the usage of ratings (some users may never use the highest rating for any movies), etc. This consideration motivates our solution to the challenge, as described in detail in Section 4.

3. RELATED WORK

One of the most widely used techniques in recommender systems is collaborative filtering (CF) [1]. Our work in this paper is closest to the branch of graph-based approaches in CF. Inspired by link prediction problem in social networks [13], researchers in recommender systems have exploited link prediction / graph-based approaches for recommending products to individual users. Random walk and its variants [8, 17] have attracted lots of attention in recommender systems [2, 7, 11, 18]. Taking the user-item rating matrix as a bipartite graph, random walk methods have been applied to rank items (nodes in the graph) for each user [7, 18]. Konstas et al. [11] proposed to place all the metadata (e.g., user friendships, item tags) along side the user-item matrix into a graph, for which random walk with restart (RWR) is applied to recommend items to individual users. Although this work succeeds in integrating multiple information sources into recommendation, the graph was constructed from binary matrices. The problem of integrating matrices with different scales into a graph has not been addressed. Our work in this paper has substantial difference regarding this issue. In the graph that we construct, all the relations among different nodes can be interpreted as interaction frequency, a key innovation in this paper. Recent work has proposed to use rich node information to guild random walk for improved link recommendation in social networks [2]. In a case in which content-based information were available, the approach we present in this paper could build on this previous work and take the information into account. We do not explore the possibility at this time, since content-based information is not included in the CAMRa2011 data set.

4. RELATIONAL CONTEXT-AWARE GRAPH MINING

4.1 Rater Likelihood

According to the problem statement as described in section 2, we formulate that the rater identification can be attained by estimating the conditional probability of a user given the item, the rating, and the time, i.e., $p(u|i,r,t)$, which actually indicates the likelihood of the user u to be the rater. By using Bayes rule and ignoring the irrelevant prior, i.e., $p(i,r,t)$ irrelevant to u, we can obtain:

$$p(u|i,r,t) \propto p(i,r,t|u)p(u) \tag{1}$$

We further make two assumptions in this work: 1) The item, the rating and the time are independent to each other. 2) the user prior follows a uniform distribution. We consider the first assumption is reasonable, since it may be probably the case (similar argumentation in other cases) that what rating a user uses would not relate to when she watches a movie. The second assumption is arguable, since it depends on the prior knowledge about the user distribution. However, our proposed model can be expanded if more priors are known exactly. Following the two assumptions, we finally factorize the rater likelihood as three conditional probabilities, as shown below:

$$p(u|i,r,t) \propto p(u|i)p(u|r)p(u|t) \tag{2}$$

Using this inference, the likelihood of the user u to be the rater depends on the likelihoods of user u being associated with item i, user u being associated with rating r, user u being associated with time t, as shown in Eq. (2). We choose to encode each of the three likelihoods in an exponential function, as below:

$$p(u|i) = \frac{1}{Z_1} e^{\alpha f_i(u)},$$
$$p(u|r) = \frac{1}{Z_2} e^{\beta(1-\alpha) f_r(u)}, \qquad (3)$$
$$p(u|t) = \frac{1}{Z_3} e^{(1-\alpha)(1-\beta) f_t(u)}$$

in which Z_1, Z_2 and Z_3 are normalization coefficients that guarantee the integral of the right side of each equation in Eq.(3) over u is 1. Note that we use $f_i(u)$ to denote a *item-proximity* function, which represents the likelihood of user u to be related to a given item i. Similarly, we refer $f_r(u)$ to *rating-proximity*, and $f_t(u)$ to *time-proximity*. α and β are scaling parameters, whose effect will be discussed later.

Substituting Eq. (3) into Eq. (2), we have:

$$p(u|i,r,t) \propto \frac{1}{Z} e^{\alpha f_i(u) + \beta(1-\alpha) f_r(u) + (1-\alpha)(1-\beta) f_t(u)} \qquad (4)$$

where $Z = Z_1 Z_2 Z_3$ is an aggregated normalization coefficient. By taking the log-likelihood and ignoring the normalization coefficient that is irrelevant to u, we can formulate the estimate of the rater likelihood as $L(u)$:

$$L(u) = \log p(u|i,r,t)$$
$$\propto \alpha f_i(u) + \beta(1-\alpha) f_r(u) + (1-\alpha)(1-\beta) f_t(u) \qquad (5)$$

In this formulation, the rater likelihood turns out to be a fusion of proximity scores based on items, ratings and time. α and β, each of which ranges from 0 to 1, are actually tradeoff parameters that controls the relative contributions from different factors. We emphasize that this formulation is approximate to address the problem of the "Rater identification" challenge as stated in Section 2. For a given item, a given rating, and a given time, we first estimate the rater likelihood for each user (each of the candidate users) in the given household according to the proposed algoritgm in Eq.(5), and then choose the most likely one as the rater. In the following, we will detail an efficient approach to compute $f_i(u)$, $f_r(u)$ and $f_t(u)$ by using a relational context-aware graph.

4.2 Relational Context-aware Graph

In this subsection, we present the construction of the relational context-aware graph (RCG). In this work, RCG consists of four types of nodes, i.e., user, item, rating and time, and the relations between them. An illustration of RCG is shown in Fig. 1. Each of the elements (subgraphs/matrices) in RCG is introduced as below:

	U	I	R	T
U	Ø	UI	UR	UT
I	UI^T	Ø	IR	IT
R	UR^T	IR^T	Ø	RT
T	UT^T	IT^T	RT^T	Ø

Figure 1: The illustration of RCG.

- **UI**: **UI** is a matrix that encodes the relations between users and items. In this work, the entry in **UI** is binary, i.e., $\mathbf{UI_{ij}} = \mathbf{1}$, if the ith user rated the jth item, 0 otherwise.

- **UR**: **UR** is a matrix that encodes the relations between users and ratings. The value of an entry $\mathbf{UR_{ij}}$ denotes the number of times (the frequency) of the ith user using the jth rating.

- **UT**: **UT** is a matrix that encodes the relations between users and time. The value of an entry $\mathbf{UT_{ij}}$ denotes the number (the frequency) of the ith user rating actions in the jth time interval. Note that in this work, we set the time interval as a day, which is available from the challenge dataset.

- **IR**: **IR** is a matrix that encodes the relations between items and ratings. The value of an entry $\mathbf{IR_{ij}}$ denotes the number of times (the frequency) of the ith item rated by the jth rating.

- **IT**: **IT** is a matrix that encodes the relations between items and time. The value of an entry $\mathbf{IT_{ij}}$ denotes the number (the frequency) of ratings that are used by users to the ith item in the jth time interval.

- **RT**: **RT** is a matrix that encodes the relations between ratings and time. The value of an entry $\mathbf{RT_{ij}}$ denotes the number of times (the frequency) the ith rating was used by users in the jth time interval.

We emphasize that all the relations in RCG can be interpreted as interaction frequencies between different nodes. In this sense, RCG avoids the scaling artifacts that could exsit among different types of nodes, as discussed in section 3. The relations that are mined from RCG would also indicate the likelihood that the two nodes have an interaction. We also emphasize that RCG can be used to involve all the available contextual information. Although in this paper RCG only exploits the available contextual information in the challenge dataset, RCG can be easily expanded to take additional information nodes into the graph.

4.3 Random Walk with Restart

As mentioned in Section 3, RWR is one of the most well-known algorithms in ranking nodes in networks/graphs. In this paper, we adopt RWR to compute proximity scores in our proposed algorithm. For notation convenience, we use $\mathbf{S}$ to denote the above RCG in the following. For a given node (supposing the kth node) in $\mathbf{S}$, and supposing that $\mathbf{S}$ is column normalized, RWR actually estimates the probabilities (denoted as a column vector $\mathbf{p}$) of the proximity between the kth node and all the other nodes (e.g., $\mathbf{p_j}$ indicates the proximity between the kth node and the jth node), according to the iteration rule, as shown below:

$$\mathbf{p}^{(n+1)} = (1-\theta)\mathbf{S}\mathbf{p}^{(n)} + \theta\mathbf{q} \qquad (6)$$

$$\mathbf{p_x}^{(0)} = \begin{cases} 0, x \neq k \\ 1, x = k \end{cases} \qquad (7)$$

in which $\mathbf{q}$ denotes the "start status" of the kth node, i.e., a column vector that is equal to the kth column of $\mathbf{S}$. θ is a parameter that controls the influence of the "start status". The vector $\mathbf{p}$ converges to a stable/stationary status during

ALGORITHM 1: Computation of $f_i(u)$ using RWR

Input: $\mathbf{S}$, an item with index i, a household H containing the candidate users with index $u \in H$, restart parameter θ, maximal number of iterations $numiter$, a stop condition ϵ.
Output: $f_i(u), u \in H$.
Column normalizing $\mathbf{S}$;
Find in $\mathbf{S}$ the kth node, which corresponds to the item with index i;
Initialize a column vector $\mathbf{p}$ (of the length as the number of rows in $\mathbf{S}$) with all zeros;
$\mathbf{p_k}^{(0)} = 1$;
$n = 0$;
repeat
  $temp = \mathbf{p}^{(n)}$;
  $\mathbf{p}^{(n+1)} = (1-\theta)\mathbf{S}\mathbf{p}^{(n)} + \theta\mathbf{q}$;
  $f = \|\mathbf{p}^{(n+1)} - temp\|_1$;
  $n = n + 1$;
until $f < \epsilon$ *or* $n > numiter$;
for $u \in H$ **do**
  Find in $\mathbf{S}$ the kth node, which corresponds to the user with index u;
  $f_i(u) = \mathbf{p_k}^{(n)}$;
end

the iterations. Applying a certain convergence condition, we can obtain $\mathbf{p}$ in which $\mathbf{p_x}$ indicates the probability that the node k is related to the node x.

The proximity scores of $f_i(u)$, $f_r(u)$ and $f_t(u)$ in Eq.(5) can be computed in a similar procedure. Here, we present the computation for $f_i(u)$ as an example shown in Algorithm 1.

Finally, the rater is identified as the one who attains the largest rater likelihood as in Eq. (5). In other words, the rater is identified as user $\hat{u}$ for a given household H, according to Eq. (8).

$$\hat{u} = \arg\max_{u \in H} L(u) \tag{8}$$

Note that the complexity of the proposed algorithm is quadratic to the size (the number of rows or columns) of $\mathbf{S}$. However, the computation of each proximity score is independent of all the others, meaning that the approach can be parallelized, a design advantage in online systems. We refer to the work of EigenTrust [10] for the readers who have interest in this issue.

5. EXPERIMENTAL EVALUATION

In this section we present a series of experiments that evaluate the proposed algorithm. We first give a detailed description of the dataset in the challenge, the experimental protocol and the evaluation metric that is used in our experiments. Then, we investigate the impact of tradeoff parameters in the proposed algorithm. Finally, we evaluate and analyze the rater identification performance of the proposed algorithm and compared it with other baselines.

We designed the experiments in order to answer two research questions:

1. Could the proposed algorithm improve the rater identification performance by fusing multiple factors for rater likelihood?
2. Could the proposed algorithm indeed benefit from RCG that encodes the relations among all the available contextual nodes?

Table 1: Statistics of the dataset

#users	∼172K
#movies	∼24K
#ratings	∼4.5M
rating sparseness	∼99.9%
#households	290
max #users in a household	4

5.1 Experimental Setup

The dataset of the challenge is provided by the organizers of CAMRa2011. The dataset contains several subsets, including one subset as the training set, one subset as the validation set together with the groundtruth. In addition, a subset contains all the relationships between users and households. Some statistics of the dataset are summarized in Table 1. The data provided in the training set is under the structure $< user_{id}, item_{id}, rating, timestamp >$, indicating a user's rating (discretely scaled 0-100) to an item and the time of the action. Note that in this work we group the timestamps into time intervals with the unit as a day, in order to construct the graph. As a result, we have 367 different time interval nodes in RCG.

The data provided in the validation set is of the structure $< household_{id}, item_{id}, rating, timestamp >$. Specifically, the task is to identify which user in the household is the rater. Note that the memberships of users to households are given, the candidate users are known for each identification case. The total number of cases to be identified in the validation set is 5450.

The evaluation metric we used to assess the accuracy of different algorithms is classification error rate, as expressed below:

$$Error\ rate = \frac{1}{|Ts|}\sum_{i=1}^{|Ts|}\delta(\hat{u}_i - u_i) \tag{9}$$

in which $\delta(x) = 0$, if $x = 0$, otherwise $\delta(x) = 1$. $|Ts|$ denotes the number of testing cases in the validation set. $\hat{u}_i$ denotes the predicted user (usually represented by the identifier of the user) for the ith case in the validation set, and u_i denotes the ground truth of the user (also represented by the identifier) in the household. Note that the result of rater identification in each case can be either true or false, i.e., a binary result. The value of $Error\ rate$ is within the range of 0 to 1, in which the lower value means the better identification performance. In the case that all the households contain two users/members, the identification $Error\ rate$ by a random selection algorithm would be 0.5.

We also notice that we skipped the discussion of restart parameter θ, since it has been widely studied in related work [2, 11]. The value of θ is empirically tuned to 0.8, which is nearly the optimal value for the identification performance by the proposed algorithm. In addition, the maximal number of iterations in Algorithm 1 is set to 10, which is sufficiently large for the algorithm convergence.

5.2 Impact of Tradeoff Parameters

Our first experiment is to investigate the impact of the tradeoff parameters α and β in Eq. (5) on the proposed algorithm. By setting $\beta = 0$, we vary α in order to observe its impact on the proposed algorithm. As can be seen in

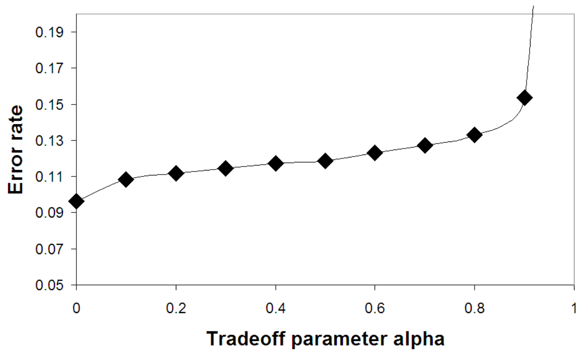

Figure 2: Impact of tradeoff parameter α on the error rate of rater identification.

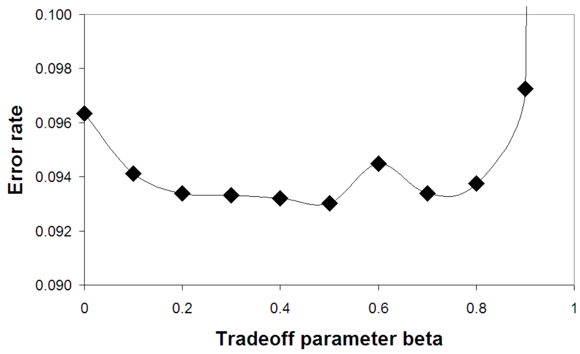

Figure 3: Impact of tradeoff parameter β on the error rate of rater identification.

Eq.(5), in the case of $\beta = 0$, the proposed algorithm actually only fuses proximity scores based on items and time, while the proximity scores based on ratings are not involved. The result is shown in Fig. 2, in which we can observe that the *Error rate* increases as α increases. The optimal performance is achieved at $\alpha = 0$, indicating that the proximity scores based on items would only introduce a negative effect for identifying raters, in the case that proximity scores based on time are used. Note that we cannot conclude that the item-proximity is useless for rater identification in general, since there might be dependence on the dataset and the framework we use in this work. We leave the issue of investigating the usefulness of user-movie interaction relations for rater identification in future work, while continuing to explore the upper limit performance that could be attained from RCG.

By fixing $\alpha = 0$ (in other words, neglecting $f_i(u)$) in Eq.(5), we investigate the impact of β on the proposed algorithm. As can be seen in Fig. 3, the optimal performance is achieved at around $\beta = 0.5$. This result indicates that by fusing time-proximity and rating-proximity, a better identification performance can be attained than only using each of the two. This observation also implies a positive answer to our first research question.

5.3 Performance Comparison

By adopting the optimal tradeoff parameters based on the observation from the previous subsection, i.e., $\alpha = 0$ and $\beta = 0.5$, we compare the identification performance of the proposed algorithm, denoted as "Optimal fusion", with other baseline approaches, which are listed below:

- **Random**: It represents a naive baseline approach that identifes the rater for a given case by randomly selecting a user from the candidate users in the household.
- **Activeness**: It represents another naive baseline that identifies the rater for a given case by selecting the most active user from the candidate users in the household. We define the activeness of the user according to the number of his ratings in the training set. Note that by this approach, the most active user in each household is always identified as the rater in the corresponding case.
- **Item-proximity**: It represents the proposed algorithm where only item-proximity is used, i.e., $\alpha = 1$, and $\beta = 0$ in Eq.(5).
- **Rating-proximity**: It represents the proposed algorithm where only rating-proximity is used, i.e., $\alpha = 0$, and $\beta = 1$ in Eq.(5).
- **Time-proximity**: It represents the proposed algorithm where only time-proximity is used, i.e., $\alpha = 0$, and $\beta = 0$ in Eq.(5).

The experimental results on *Error rate* for all the approaches are shown in Table 2. We summarize our observations into three aspects:

First, Optimal fusion outperforms all the other approaches. Note that improvements achieved by Optimal fusion over all the baselines are statistically significant, according to Wilcoxon signed rank significance test with $p < 0.05$, which was measured across all of the 5450 cases. Compared to the naive baselines, Optimal fusion achieves ca. 400% improvement over Random approach and ca. 300% over Activeness approach. Compared to the variants, each of which only takes into account one factor to estimate the rater likelihood from RCG, Optimal fusion achieves improvements ca. 500% over Item-proximity, ca. 200% over Rating-proximity and 3% over Time-proximity. Note that although the improvement of Optimal fusion over Time-proximity is much smaller compared to all the other cases, this improvement is still statistically significant with $p = 0.033$. As a result, we can confirm a positive answer to our first research question.

Second, we observe that the improvement of Optimal fusion over Time-proximity is much smaller than its improvements over other approaches. Note that the performance of Time-proximity is much better than the other baseline approaches, indicating that time is the most effective factor for identifying the raters. We notice it is critical to examine the performance that can be attained solely by exploiting the time factor, i.e., without other contextual information nodes (items and ratings). For this reason, we estimate the rater likelihood by RWR on a bipartite graph which consists of user nodes and time interval nodes. The result is compared with the proposed algorithm using RCG, as shown in Table 3. We can observe that Time-proximity using RCG outperforms Time-proximity using the bipartite graph by 4%, and Optimal fusion using RCG outperforms Time-proximity using the bipartite graph by 7%. This observation allows us to confirm a positive answer to our second research question, i.e., the proposed algorithm does benefit from RCG

Table 2: Performance comparison between the proposed algorithm "optimal fusion" with other baseline and variant approaches

Approach	Error rate
Random	0.512
Activeness	0.387
Item-proximity	0.601
Rating-proximity	0.322
Time-proximity	0.096
Optimal fusion	**0.093**

Table 3: Error rate comparision of using bipartite graph and using RCG

Time-proximity (bipartite)	Time-proximity (RCG)	Optimal fusion (RCG)
0.100	0.096	0.093

that consists of all the context-aware relations for rater identification.

Finally, we look into the rater identification performance for households with different household sizes, i.e., the number of users (usually family members) in a household. Note that in the challenge dataset, the minimal number of users in a household is 2, and the maximal number is 4. The number of cases for each of the household size in the validation set is 4953 for household size 2, 320 for household size 3, and 177 for household size 4. As can be seen in Table 4, Optimal fusion succeeds in improving over Random, Activeness, Item-proximity and Rating-proximity to a large extent across all the household sizes. Compared to Time-proximity, Optimal fusion achieves improvement in the case of household size 2, the majority of all the cases. The result indicates that the proposed algorithm is robust for identifying raters from different households.

6. CONCLUSION AND FUTURE WORK

We present a novel algorithm for rater identification in this paper. The proposed algorithm is experimentally demonstrated to be effective in fusing various factors for estimating rater likelihood. Our experimental results also reveal that time could be the most critical factor for rater identification, while additional improvement can still be achieved by exploiting all the other contextual information regarding the user.

Our future work on this topic would involve several directions. We are interested in investigating finer time measures for our appraoch. We will also investigate the possibility of improving our approach by exploiting content-based node information as studied in [2]. We are also interested in investigating novel discriminative approaches for rater identification, by taking experience and lessens concerning "rated item identification" from another recent challenge on Yahoo music recommendation[3].

7. ACKNOWLEDGMENTS

The research leading to these results was carried out within the PetaMedia Network of Excellence and has received funding from the European Commission's 7th Framework Program under grant agreement n° 216444.

[3]http://kddcup.yahoo.com/

Table 4: Rater identification performance for households with different household sizes

	size=2 (4953)	size=3 (320)	size=4 (177)
Random	0.494	0.672	0.734
Activeness	0.368	0.509	0.678
Item-proximity	0.584	0.713	0.887
Rating-proximity	0.299	0.494	0.644
Time-proximity	0.097	**0.106**	**0.056**
Optimal fusion	**0.093**	0.109	**0.056**

8. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Kowledge and Data Engineering*, 17(6):734–749, 2005.

[2] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 635–644, New York, NY, USA, 2011. ACM.

[3] L. Baltrunas, T. Makcinskas, and F. Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 119–126, New York, NY, USA, 2010. ACM.

[4] C. Beckmann and T. Gross. Towards a group recommender process model for ad-hoc groups and on-demand recommendations. In *Proceedings of the 16th ACM international conference on Supporting group work*, GROUP '10, pages 329–330, New York, NY, USA, 2010. ACM.

[5] S. Berkovsky and J. Freyne. Group-based recipe recommendations: analysis of data aggregation strategies. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 111–118, New York, NY, USA, 2010. ACM.

[6] Z. Gantner, S. Rendle, and S.-T. Lars. Factorization models for context-/time-aware movie recommendations. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 14–19, New York, NY, USA, 2010. ACM.

[7] M. Gori and A. Pucci. Itemrank: a random-walk based scoring algorithm for recommender engines. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 2766–2771, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[8] T. H. Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, WWW '02, pages 517–526, New York, NY, USA, 2002. ACM.

[9] A. Jameson and B. Smyth. The adaptive web. chapter Recommendation to groups, pages 596–627. Springer-Verlag, Berlin, Heidelberg, 2007.

[10] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pages 640–651, New York, NY, USA, 2003. ACM.

[11] I. Konstas, V. Stathopoulos, and J. M. Jose. On social networks and collaborative recommendation. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 195–202, New York, NY, USA, 2009. ACM.

[12] Y. Koren. Collaborative filtering with temporal dynamics. *Commun. ACM*, 53:89–97, April 2010.

[13] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, pages 556–559, New York, NY, USA, 2003. ACM.

[14] N. N. Liu, B. Cao, M. Zhao, and Q. Yang. Adapting neighborhood and matrix factorization models for context aware recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 7–13, New York, NY, USA, 2010. ACM.

[15] A. Said, S. Berkovsky, and E. W. De Luca. Putting things in context: Challenge on context-aware movie recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 2–6, New York, NY, USA, 2010. ACM.

[16] Y. Shi, M. Larson, and A. Hanjalic. Mining mood-specific movie similarity with matrix factorization for context-aware recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 34–40, New York, NY, USA, 2010. ACM.

[17] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *Proceedings of the Sixth International Conference on Data Mining*, ICDM '06, pages 613–622, Washington, DC, USA, 2006. IEEE Computer Society.

[18] H. Yildirim and M. S. Krishnamoorthy. A random walk method for alleviating the sparsity problem in collaborative filtering. In *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, pages 131–138, New York, NY, USA, 2008. ACM.