2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Exploiting Homophily-based Implicit Social Network to Improve Recommendation Performance

Tong Zhao, Junjie Hu, Pinjia He, Hang Fan, Michael Lyu and Irwin King

***Abstract*—Social information between users has been widely used to improve the traditional Recommender System in many previous works. However, in many websites such as Amazon and eBay, there is no explicit social graph that can be used to improve the recommendation performance. Hence in this work, in order to make it possible to employ social recommendation methods in those non-social information websites, we propose a general framework to construct a *homophily-based implicit social network* by utilizing both the rating and comments of items given by the users. Our scalable framework can be easily extended to enhance the performance of any recommender systems without social network by replacing the homophily-based implicit social relation definition. We propose four methods to extract and analyze the implicit social links between users, and then conduct the experiments on Amazon dataset. Experimental results show that our proposed methods work better than traditional recommendation methods without social information.**

I. Introduction

As the dramatic growth of online stores, information filtering techniques like recommender system are widely used in such websites, like Amazon. To avoid massive information and description of the related products blocking users from quickly reaching their interested products, recommender system utilizes a specific ranking criteria that will suggest a list of potential products to the users so that users can quickly select their interested products. In this way, recommender system has shown great power to boost the sale for the online stores in practice.

Although commercial success in the online websites has convinced the significant role of recommender system, traditional recommender systems suffer from the several weaknesses. Firstly, Data sparsity is the inherent challenge with respect to massive information of huge numbers of products. As reported in recent survey, available ratings from the users are usually less than 1% of all the products. Thus how to suggest related products to the users who rate just a few or no products poses a great challenge. Secondly, traditional recommender system tends to ignore the social information between users with common interest. This contradicts the real cases. In real life, we are easily affected by our friends and some experts with common interest with you. When we want to purchase some kind of products, we tend to ask our friends who have ever bought those products or people who share the common taste with you to those products. Besides, even for some unknown people, we would probably make friends with them if they share almost the same interest with us. Due to this intuition, many social recommender system ([10], [11], [19]) were proposed and outperformed traditional recommender system without social information. However, some online systems, like Amazon, eBay, etc., do not form an explicit community around friends or experts in some fields, hence we have difficulty to use the social information for recommendations in such websites. Does it means that Social Recommendation method can no longer be applied to this kind of websites? How to build an implicit social graph by the available information of users motivates our work.

In this paper, we aims at analyzing the similarity between users and predicting a potential weighted connection between them. In our framework, we make the following assumptions according to real life recommendations from our friends.

- Users highly sharing with specific common interest are tended to form a community.
- Users can be easily affected by their friends in the same community and tend to follow their friends' recommendation.
- Users may simultaneously have several interests which may be different with their friends. Hence the influence from one user's friends would be weighted unequally.

Based on the above assumptions, we propose four strategies to analyze the similarity between the users and conduct experiments using data from Amazon. As for Amazon dataset, since we have few explicit social relationship between users, the contribution to our framework lies in how to dig up the social information from the various data such as review comments to some

Tong Zhao, Junjie Hu, Pinjia He, Hang Fan, Michael Lyu and Irwin King are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong (email: {tzhao, jjhu, pjhe, hfan, lyu, king}@cse.cuhk.edu.hk).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

specific products, rating to products belong to specific topics, etc.

The main contribution of our work can be summarized as follows.

1) We purpose a general and scalable framework to apply social recommendation method to the online websites without explicit social information.
2) We put forward four methods to dig up the similarity between the users and build a homophily-based implicit social graph.
3) We employ both traditional recommendation methods and social recommendation methods to verify the idea.

The rest of the paper is organized as follows. In Section II, we briefly review the related works and then we formally define our problem in Section III. Section IV describes our proposed framework together with four scalable strategies to analyze the links between users. We conduct the experiments using data from Amazon and give our empirical analysis on the results in Section V. Finally we conclude our work in Section VI.

II. Related Work

Much effort has been made for recommender systems and a large number of works based on Collaborative filtering[7], [13] have been done for helping users find out the most valuable information. Recently, several matrix factorization methods [14], [15], [16] are proposed for collaborative filtering. They focus on representing the user-item rating matrix with low dimension latent vectors. However, these methods for recommender systems assume that users are independent and also ignore the social activities between users. In fact, the reality is that recommendations from users' friends are more convincing and many websites, such as Lastfm, Delicious and Slashdot, provide the means for users to build their trust/social relationships. Hence, how to understand trust relationships in social networks and how to make full use of the graph-based trust/social relationships and user-item rating matrix for improving the accuracy of recommendation have been well studied.

There are a few works focusing on trust analysis in social networks. Siegler et al. pointed out that there is a strong and significant correlation between trust and similarity[22]. Lu et al. design a framework for incorporating social context information to improve review quality prediction[9]. Guha et al. proposed a framework for modeling trust propagation[3]. Leskovec et al. utilize the topological feature of a social network to predict the trust and distrust relations among users[6]. Golbeck investigates various properties of trust such as transitivity, composability and asymmetry. Matsuo and Yamamoto study the bidirectional effects between trust relations and product rating[12]. Tang[18] et al. measure the multi-faceted trust strength between users on category level.

Furthermore, several trust-based approaches [1] and influence-based approaches ([2], [5], [8], [20]) are proposed for improving recommendation accuracy. SoRec [10] is proposed as a probabilistic matrix factorization framework which incorporates trust network information into user taste analysis. Ma et al. [11] also propose a matrix factorization framework with social regularization based on the assumption that users' interests should be similar to their friends. Yang et al. [19] further investigate the contribution of social relations to the recommender system. Jiang et al. [5] design a novel matrix factorization framework which exhibits the contribution of two important factors: individual preference and interpersonal influence. In [20], Ye et al. propose a generative model for social recommendation, which captures social influence between friends quantitatively and employ social influence to mine the personal preference of users. Shen et al. [17] also propose a joint personal and social latent factor model for social recommendation.

III. Problem Definition

In this section, we first introduce several notations and definitions used in the paper and then formally define the problem of exploiting homophily-based implicit social network to improve recommendation performance. Here, we use *M* denotes the number of users, *N* is the number of items.

Now we define the concepts that will be used in the paper.

Definition 1: **User-item Matrix**: Let R be an $M \times N$ matrix in which every row corresponds to a user, each column an item; and each element records the rating score, $r_{ij} \in \mathbb{R}$ of item *j* rated by user *i*.

Definition 2: **User-item-review Tuple**: The user-item-review tuple $< u, i, w_{ui} >$ represents the rating score of item *i* given by user *u* and the corresponding review text about this score w_{ui}. Here w_{ui} is a word vector containing all the words user *u* use to comment item *i*.

Definition 3: **Implicit Trust Network**: The implicit trust network can be represented as $G^* = (S, E^*)$, where S is the set of users and $E^* \subset S \times S$ is the set of unobserved links representing the related users own similar interests.

Definition 4: **Social Rating Network**: The social rating network can be described as $\{R, G^*\}$ containing both user's rating to items and the social link

information among users. Recommendation in *Social Rating Network* refers to solve the problem of how to utilize the social network and the user-item matrix to recommend items to the particular users who are interested in them. The key challenge is how to use social information to improve the recommendation accuracy.

Based on the above definitions, now we formally define the problem of Exploiting Homophily-based Implicit Social Network to Improve Recommendation Performance.

Problem 1: **Exploiting Homophily-based Implicit Social Network to Improve Recommendation Performance** Given a user-item matrix R and the corresponding user-item-review tuples $\sum < u, i, w_{ui} >$, the goal is to build an implicit trust network G^* by utilizing the review text and rating scores of $\sum < u, i, w_{ui} >$ and then perform recommendation via the new social rating network $\{R, G^*\}$.

The basic idea of this problem is to find a proper way to transform the tradition recommender systems to social recommender systems by constructing an implicit social network among users.

IV. Homophily-based Social Recommendation Framework

Figure 1 shows the general process of our proposed framework. First, we analyze the different types of data, like comment text, rating score, etc., and extract useful information. Then we utilize the information to build the homophily-based implicit social relations between users based on a specific similarity measurement. Finally, based on the built implicit social network, we can apply several social recommender methods to providing users with more realistic recommendation.

In the following subsections, we detail four methods to build implicit social relations between users and then introduce several social recommendation methods that can be employed in our framework.

A. Method 1:Common Rating

The first method is straightforward and only consider the rating information. Specifically, given two users who present ratings on the same products, it's reasonable that they may have more or less similar tastes on some products or belong to a community that share the same interest in some kind of products. Hence they has high probability to have a connect rather than those without any common ratings on the same products. Therefore, we would like to assign a uniformly weighted link between these users with probability p, where p can be chosen according to different scenarios. In this way, we can build a very dense social network

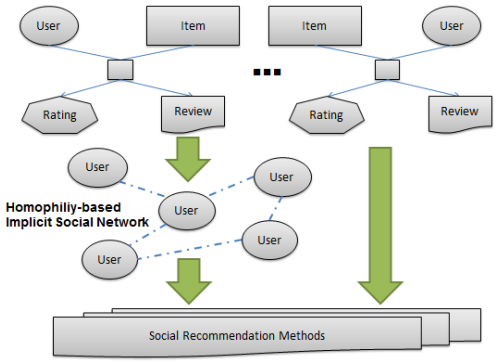

Fig. 1. Framework to perform social recommendation on online websites without explicit social information

by connecting the potential users. In this way, we can generate a *Social Rating Network* $\{R, \hat{G}\}$, which is an implicit social graph by the weighted link between users.

B. Method 2:Pearson product-moment correlation (PCC)

Pearson product-moment correlation coefficient, well known as PCC, is a statistical measurement of linear correlation between two variables. Equation 1 gives the PCC formula of two variable X and Y.

$$PCC(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})}}, \quad (1)$$

Here let X and Y be the rating scores from two users, and $\overline{X}$ and $\overline{Y}$ denote the average ratings by the two users. Therefore, PCC value can capture the rating similarity between the two users. Specifically, by utilizing the *User-item Matrix*, we can find out the candidate users pair who have ever rated the same items. Then we can calculate the PCC value of their rating scores on their common items. By PCC measure, we can filter out some users pairs which has more or less the similar rating scores on the same items. Based on the assumption that users with similar tastes on different types of products have higher probability to form a community and are likely to make friends with each other even if they don't know each other before. For example, researchers who work in the field of machine learning is more likely to appreciate the work from those who are also expert in machine learning rather than those are expert in drawing.

Since we are dealing with huge amount of data, the number of the candidate user pairs can be as much as the square of the number of the available users. However, candidate pair with PCC value near

to zero do less conduce to the optimization of the objective function. Therefore, after filtering out the candidate user pairs with $PCC < threshold$ where the $threshold$ depends on the concrete scenarios, we assume there are some highly implicit social relations between these users and assign the PCC value as the weight for the links between them. In this way, we can generate a *Social Rating Network* $\{R, \hat{G}\}$, which is an implicit social graph by the weighted link between users.

C. Method 3:Topic Similarity

As we know, text comment contains rich information of the users towards some particular products. However, due to the various expression of the same meaning, how to measure the similarity of the users' tastes towards some kind of products remains a great challenge. For example, users may describe the expensive camera using some comments as high price or high cost. To extract the meaning of the review comment, topic model is a tool to statistically discover the abstract topics that may hide in a collection of documents. Intuitively, people sharing with the similar interest would present similar topics in their words. For example, people who are really Apple fans are most likely to talk with their friends about the new products of Apple and be familiar with the quality of some kind of new products. Implicitly, they form a community without their recognition and their review comments are usually very useful to other people who would like to purchase the Apple products. Hence in this case, if we can figure out this group of people, we can do better recommendation to the unknown users who have high probability to join this group. Topic model that uncovers the hidden thematic structure of the users' comments can help to dig up the similarity between users. Lots of works on topic analysis and sentiment analysis have been done to mine the implicit information from the text.

Here we employ Author-Topic model (ATM) as a tool to investigate users' interests revealed by their comments. ATM takes a set of documents and the related authors as input, and it gives the author-topic distribution and topic-word distribution as output. In this method, we summarize all the comments written by a particular user as a document and then use ATM to obtain the user-topic distribution to represent user's interests.

The author-topic distribution θ and topic-word distribution ϕ can be calculated by Gibbs sampling as follows.

$$\theta_{ak} = \frac{N_{ak}^{UK} + \alpha}{\sum_{k'} N_{ak'}^{UK} + K\alpha}, \tag{2}$$

$$\phi_{wk} = \frac{N_{wk}^{VK} + \beta}{\sum_{w'} N_{w'k}^{VK} + V\beta}, \tag{3}$$

where U, V, K represent the number of users, words and topics, respectively, N_{ak}^{UK} represents the number of times that topic k assigned to user a, and α, β are the hyperparameters of the dirichlet distributions, w' and k' represent each word and topic.

Given such user-topic distribution, now the user-interest similarity can be calculated by the similarity between the corresponding user-topic distribution vectors. The candidate pair $(U1, U2)$ with $Sim(U1, U2)$ value near to zero do less conduce to the optimization of the objective function. Therefore, after filtering out the candidate user pairs with $Sim(U1, U2) < threshold$ where the $threshold$ depends on the concrete scenarios, we assume there are some highly implicit social relations between these users and assign the $Sim(U1, U2)$ value as the weight for the links between them. In this way, we can generate a *Social Rating Network* $\{R, \hat{G}\}$, which contains an implicit social graph with the weighted link between users.

D. Method 4:Fine-grained Topic Similarity Analysis

Based on the available Amazon data, we also discover that for different rating score, the review comments for the same user also present diversified topics as well. For example, in Figure 2, for two users with the same rating of 5 on two different movies "Titanic" and "Man of Steel", we may easily find out the different tastes for users having even the same rating. The first user is highly fond of love story and romantic movies, while the second user shows great interest in hero story in the war scenario. Similarly in Figure ẽffig:Figure3, two users give the same one rating score to two different movies. Judging from the review comment of the first user, we discover that he may not be expert in Computer Science or program Engineering. Thus he doesn't show great interest in the movies full of technical terms in Computer Science. The same situation happens in the second users. From his review comment, we find out that he doesn't feel great interest in this kind of ludicrous chase movie and fighting war with zombies. According to this observation, different rating scores can represent the favour extend of the users towards some categories of the products. Therefore, we need to take into account the different influences of the comments with different rating scores separately and have a more reasonable combination method to measure the preference of the users.

Out to this purpose, we perform the topic model method on the users' review comments in different rating score R_i, where $R_i = i$ and $i \in [1, 5]$. Figure 4 shows the general steps to conduct Method 4. First we separate the review comment according to different

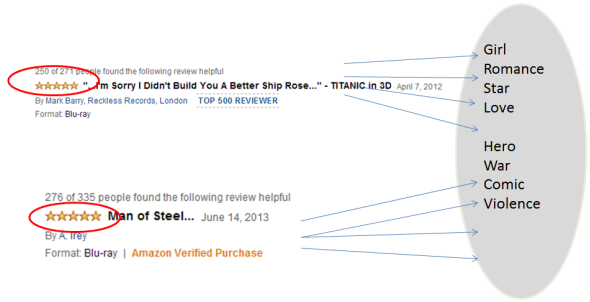

Fig. 2. Illustration to diversified topics for two users with the same rating scores 5 and strong preference towards some topics

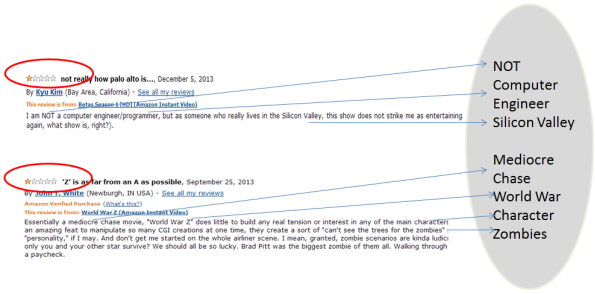

Fig. 3. Illustration to diversified topics for two users with the same rating scores 1 and strong dislike towards some topics

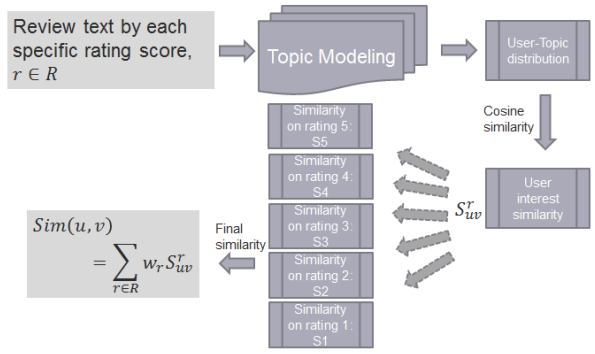

Fig. 4. Framework of method 4

rating scores. For every R_i, we conduct topic model method to generate a vector for each user, where each entity of the vector measures the proportion of the specific topic in all the review comment of this user with rating score R_i. For each user pair $(U1, U2)$, we calculate the topic similarity $Sim_i(U1, U2)$ between user 1 and user 2 who have ever rated some products with the same rating score R_i. Finally we measure the similarity of user 1 and user 2 by a reasonable weighted sum of $Sim_i(U1, U2)$ in Equation 4.

$$Sim(U1, U2) = \sum_{i=1}^{5} w_i Sim_i(U1, U2), \qquad (4)$$

where $Sim(U1, U2)$, $Sim_i(U1, U2)$ and w_i denote the integrated topic similarity between user 1 and user 2, the topic similarity between user 1 and user 2 in rating score R_i and the weight for $Sim(U1, U2)$ respectively. $Sim_i(U1, U2)$ can be calculated in different ways, e.g. cosine similarity, KL-divergence. Here we choose cosine similarity. Since rating score R_5 and R_1 represent the strong preference and dislike respectively of each user while rating score R_3 represents neutral attitude towards some topic, the weight w_i should be proportional to the importance of the rating scores. Hence, we assign larger weight value to the similarity on rating 5 and 1, and smaller value on other ratings. After filtering out the candidate user pairs with $Sim(U1, U2) < threshold$ where the $threshold$ depends on the concrete scenarios, we assume there are some highly implicit social relations between these users and assign the $Sim(U1, U2)$ value as the weight for the links between them.

E. Social Recommendation Methods

In this subsection, we introduce the basic ideas of social recommendation methods and then explain how to employ these methods in our proposed framework. Since the proposed framework is quite general that most of the social recommendation methods can be adapted, here we just take three popular social recommendation methods, *SoReg*, *SoRec* and *SocialMF*, as examples.

Social Spectral Regularization. The first introduced idea of social recommendation methods can be termed as social spectral regularization approaches since they are identical to the objectives used in spectral clustering . One typical example is called *SoReg* with the assumption that users with social relations own similar interest-feature vectors. Given a homophily-based implicit social graph based on the users' similarity, *SoReg* method defines the objective function in Equation 5. We would like to optimize the objective function using the similarity of the candidate user pairs linked by the implicit homophily-based social relations.

$$\begin{aligned}\min_{U,V} \mathcal{L}_2(U, V, R) = &\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2 \\ &+ \frac{\beta}{2}\sum_{i=1}^{m}\sum_{f \in \mathcal{F}^+_{(i)}} Sim(i, f) \parallel U_i - U_f \parallel_F^2, \\ &+ \lambda_1 \parallel U \parallel_F^2 + \lambda_2 \parallel V \parallel_F^2\end{aligned} \qquad (5)$$

where $\mathcal{F}^+_{(i)}$ represents user i's social relations in the implicit social network.

Social Regularization. The second type of social recommendation methods aims to constrain the latent projection of users according to social network information. *SoRec* method is a representative work with the following objective function that model both user's rating and social relations simultaneously.

$$\min_{U,V} \mathcal{L}_2(U,V,R) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2 + \frac{\beta}{2}\sum_{i=1}^{m}\sum_{f\in\mathcal{F}_{(i)}^+} \parallel S_{i,f} - g(< U_i, U_f >) \parallel_F^2 + \lambda_1 \parallel U \parallel_F^2 + \lambda_2 \parallel V \parallel_F^2 \tag{6}$$

where $g(\cdot)$ denotes the logistic function and $< \cdot, \cdot >$ denotes the inner product.

Social Propagation Regularization Social Propagation Regularization allows the propagation of user interest through social relations. *SocialMF* is a typical *Social Propagation Regularization* based approach. Different from other methods, in *SocialMF* the feature vector of each user is modeled based on the feature vectors of his direct neighbors in the trust network. As in the training process, each user feature vector will absorb the value from his friends' feature vectors and then contributes to other users' latent feature vectors, this allows SocialMF to handle the transitivity of trust and trust propagation as follows.

Since we have built an implicit homophily-based social network based on users' rating and comments, now we can easily adopt social recommendation methods to the traditional non-social recommender systems. After identifying the user pairs whose homophily similarity is larger than a threshold, we can directly assign implicit social relations between these users and then make it possible to use social recommendation methods to improve the recommendation performance in the non-social recommender systems. For both *SoReg* and *SoRec*, we utilize stochastic gradient descent approach to approximate the optimal solution.

V. Experiment

A. Data Description

We use a dataset consists of movie reviews from Amazon. The data spans a period of more than 10 years, including all 8 million reviews up to October 2012. Reviews include product and user information, ratings, and a plain text review.

TABLE I. Data Statistics.

Number of reviews	7,911,684
Number of users	889,176
Number of movies	253,059
Users with more than 50 reviews	16,341
Timespan	Aug 1997 - Oct 2012

B. Data Preprocessing

The raw data consists of 889,176 users, 253,059 movies and 7,911,684 ratings between them. To extract useful and helpful information, we have to preprocess the raw data and generate several data files in a clean style. To dig out users who gave common ratings on the same movie, we have to prepare a user-movie-rating file which includes all ratings available in our dataset. Similarly, as mentioned in Section IV, a user-movie-rate text is needed as the input file when considering about Pearson product-moment correlation. Thus, we need to generate a user-movie file first. Besides, we are supposed to collect all reviews for each user in our dataset. Because the input of our topic model are many small plain text files, each of which contains all the reviews posted by a specific user on Amazon's website. Topic model will read all these files, which represent for corresponding users, analyze them and map them to a 10-dimensional vector space. After mapping, we are able to try out our method 3 and 4 on it.

1) Extract User-movie-rating Pairs: For one record in the raw data file, we have 8 separated lines which stand for eight features of that record. In each record, the dataset provide us with product ID, user ID, profile name, helpfulness, rating, time, summary and review text. Among all these features, we should extrct user ID, movie ID and the corresponding ratings.

In our preprocessing step, we find out that some lines in the raw data file can't be read normally using Python 2.7 engine. After detailed analyzing, we locate the problem that some illegal characters exist in the raw data file, and Python 2.7 engine will just regard them as EOF mark and stop reading raw data. To address this problem, we neglect records which contains illegal characters and simply jump to the next one. Because the quantity of these illegal records are so small compared with all records we possess that it can hardly affect our prediction accuracy.

Then we extract user ID, service ID and the corresponding rating from each record and output all of them into a tsv (tab-separated values) file. This file has 7,911,424 lines, which indicates 7,911,424 available user-movie-rating records. Afterwards, for the purpose of making those following data analyzing steps easier, we write a hash function to map users as well as movies into two integer spaces. We also keep their mapping rules for further reference.

2) Collect Reviews for Each user: As we mentioned above, similar users are found mainly through two features: numeric rating and text. To make use of the numeric rating feature, we have generated a user-movie-rating file containing all available numeric feature information that implies similarity between different users. Now we will introduce how we vectorized those plain text features and map them into numeric vector spaces.

To leverage the efficiency of calculating similarity and the power of topic representation, we set the number of topics as 10 after several empirical studies. Topic model is then applied to dig out how much these 10 latent topics impact each user and output a 10-dimensional vector with numeric value for each of them. The input of our topic model program is a bunch of plain text file. Each file contains not only all summaries but also reviews of a movie with the same rating value that a user wrote.

Thus, for each user, we generate five different plain text feature files. At least more than 1 million files under each rating value are constructed after this preprocessing step. Then they will be used as the input of topic model and mapped into a 10-dimensional vector space.

3) Calculate Similarity Based on Text: After the vectorization step, we form five new vectors, which are stored in five files, to identify each user. Then we calculate the similarity between each two user-pair under all five files. (Each file is a 10-dimensional vector which stands for the identity of that user under a specific rating value.) Then similarity between two users are calculated by Equation 4.

C. Methods in Comparison

We employ the following methods to help demonstrate the helpfulness of implicit trust.

- **PMF:** This method is proposed by Salakhutdinov and Minh [16]. It only uses the user-item matrix for recommendations.
- **RPMF:** This method is proposed by Erheng Zhong [21]. It uses the decision tree structure to build a hierarchical matrix factorization framework for recommendation by incorporating context information.
- **SoRec:** This method [10] incorporates social network data into user-item data by extracting a common latent factor, using *Probabilistic Matrix Factorization*.
- **SoReg:** This method [11] designs a matrix factorization framework with social regularization to constraint social recommendation based on the idea that users present similar interests to their friends.
- **SocialMF:** This method [4] utilizes matrix factorization technique to allow interest propagation through social relations.

TABLE II. $RMSE$ OF DIFFERENT METHODS ON AMAZON DATA

Methods	Network Setting	RMSE	MAE
PMF	/	0.2331	0.1760
RPMF	/	0.2321	0.1710
SoReg	Common Item	0.2322	0.1746
SoRec	Common Item	0.2412	0.1788
SocialMF	Common Item	0.2323	0.1721
SoReg	PCC	0.2310	0.1746
SoRec	PCC	0.2344	0.1763
SocialMF	PCC	0.2275	0.1681
SoReg	Topic Analysis	0.2321	0.1747
SoRec	Topic Analysis	0.2354	0.1770
SocialMF	Topic Analysis	0.2315	0.1705
SoReg	Topic Analysis by rate	0.2302	0.1666
SoRec	Topic Analysis by rate	0.2324	0.1683
SocialMF	**Topic Analysis by rate**	**0.2295**	**0.1661**

D. Experiment Performance

Table II shows the $RMSE$ and MAE value of the above methods, which is PMF, RPMF, SoReg, SoRec and SocialMF. In the experiment, we utilize four strategies in Session 4. Root of mean square error($RMSE$) and mean of absolute error(MAE) are often used as evaluation criteria for Recommender System.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(x_t - \overline{x})^2}{n}}, \tag{7}$$

$$MAE = \frac{\sum_{t=1}^{n}|x_t - \overline{x}|}{n}, \tag{8}$$

where $\overline{x}$ is the mean of the sequence x_i.

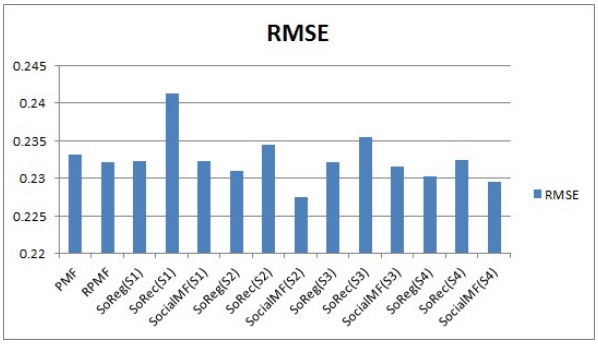

Fig. 5. $RMSE$ performance using different methods

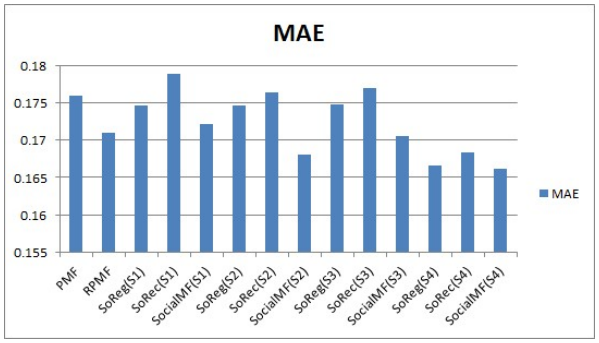

Fig. 6. MAE performance using different methods

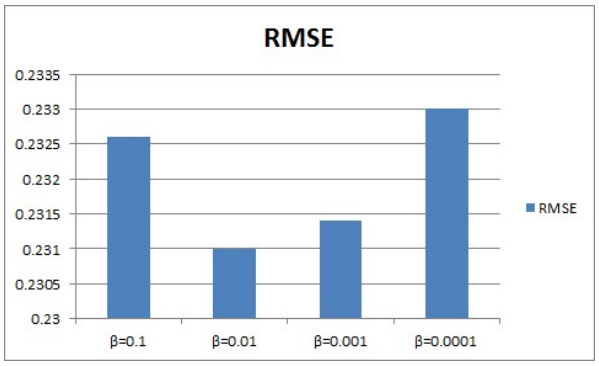

Fig. 7. $RMSE$ performance according to parameter $\beta \in \{0.1, 0.01, 0.001, 0.0001\}$

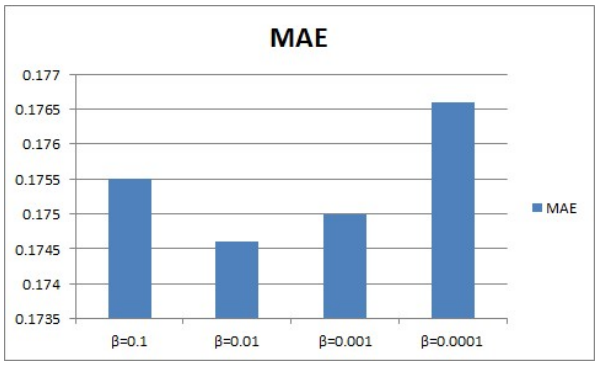

Fig. 8. MAE performance according to parameter $\beta \in \{0.1, 0.01, 0.001, 0.0001\}$

E. Result Analysis

5 and 6 demonstrate the experimental results of different recommendation methods under the homophily-based implicit social networks generated by different strategies. In the figures, (S*) represents the number of methods for generating the social information. We can see that by incorporating both topic information extracted from comments and the rating similarity to set up the implicit social network, all the social-based recommendation methods can be adopted and achieve better performance than those non-social recommendation methods.

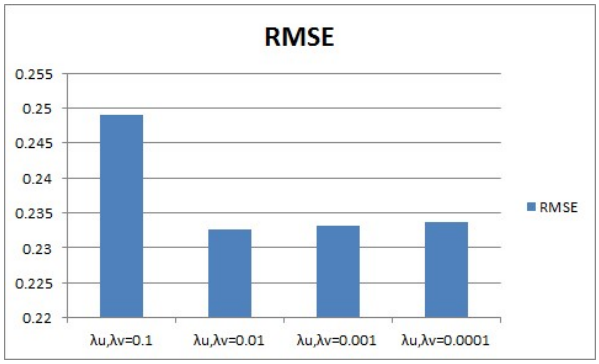

Fig. 9. $RMSE$ performance according to parameter $\lambda_u \lambda_v \in \{0.1, 0.01, 0.001, 0.0001\}$

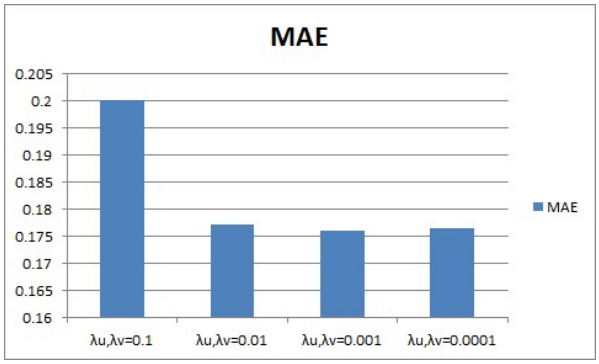

Fig. 10. MAE performance according to parameter $\lambda_u \lambda_v \in \{0.1, 0.01, 0.001, 0.0001\}$

Moreover, extensive experiments are done to observe the impact of parameters in our approach. To figure out which β value provides us with the best performance in our model, we run experiments under different β ($\beta = 0.1$, $\beta = 0.01$, $\beta = 0.001$ and $\beta = 0.0001$). Fig. 7 and Fig. 8 show us the $RMSE$ as well MAE of our model under different β, respectively. From the above mentioned two figures, we can easily find out that when $\beta = 0.01$, our approach performs the best.

Then we use the same method to dig out the most suitable value for parameters λ_u and λ_v. For the purpose of simplicity, we set $\lambda_u = \lambda_v$. Two groups of experiments are done under different parameter settings ($\lambda_u = \lambda_v = 0.1$, $\lambda_u = \lambda_v = 0.01$, $\lambda_u = \lambda_v = 0.001$ and $\lambda_u = \lambda_v = 0.0001$). Fig. 9 and Fig. 10 display $RMSE$ and MAE of $SocialMF$ under different λ_u,λ_u values, respectively. Experiment results show that when we set $\lambda_u = \lambda_v = 0.01$, the *SocialMF* under the generated implicit social network raises the best prediction.

VI. Conclusion

In this work, we proposed a general framework to build a homophily-based implicit social network with the purpose of applying social recommendation to some online websites without explicit social information. Due to this motivation, we proposed four strategies to extract the social relationship between users and perform some classical social recommendation methods on Amazon dataset. Our framework is scalable and can be easily extended to different scenarios by substitutional similarity measurement. Experiments on Amazon dataset show promising improvements of the recommender system by achieving less $RMSE$ and MAE. We find that exploiting both rating and topic analysis to build the homophily-based implicit social network can achieve best improvement. By well selecting the good parameters for social recommendation methods, we analyze the effect of the robustness of our proposed strategies.

VII. Future work

Open directions lie in the following aspects. First, a crucial question is how to evaluate the suitable strategies in this framework so that best method can be easily selected. The second question is how to improve the model to deal with huge number of data when the number of the available users pair is limited. Third, more studies should be put on how to dig up more meaningful similarity from other sources of data, for example the clicks to the *useful comment* button from other users, etc. Finally, we need to further analysis the robustness of our framework against the choose of the parameters.

VIII. Acknowledgement

The work described in this paper was fully supported by the Basic Research Program of Shenzhen (Project No. JCYJ20120619152419087 and JC201104220300A), and the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK 413212 and CUHK 415212).

References

[1] P. Bedi, H. Kaur, and S. Marwaha. Trust based recommender system for semantic web. *In Proceedings of IJCAI'07*, pages 2677–2682, 2007.

[2] P. Cui, F. Wang, S. Liu, M. Ou, S. Yang, and L. Sun. Who should share what? item-level social influence prediction for users and posts ranking. *In Proceedings of SIGIR'11*, pages 185–194, 2011.

[3] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. *In Proceedings of WWW'04.*, 2004.

[4] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. *In RecSys.*, 2010.

[5] M. Jiang, P. Cui, R. Liu, Q. Yang, and F. Wang. Social contextual recommendation. *In Proceedings of CIKM'12*, pages 45–54, 2012.

[6] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. *In Proceedings of WWW'10.*, 2010.

[7] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, pages 76–80, 2003.

[8] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang. Mining topic-level influence in heterogeneous networks. *In Proceedings of CIKM'10*, 2010.

[9] Y. Lu, P. Tsaparas, A. Ntoulas, and L. Polanyi. Exploiting social context for review quality prediction. *In Proceedings of WWW'10.*, 2010.

[10] H. Ma, H. Yang, M. R.Lyu, and I. King. Sorec: Social recommendation using probabilistic matrix factorization. *In Proceedings of CIKM'08*, pages 931–940, October 2008.

[11] H. Ma, D. Zhou, C. Liu, M. R.Lyu, and I. King. Recommender systems with social regularization. *In Proceedings of WSDM'11*, pages 287–296, February 2011.

[12] Y. Matsuo and H. Yamamoto. Community gravity: measuring bidirectional effects by trust and rating on online social networks. *In Proceedings of WWW'09.*, 2009.

[13] P.Resnick, N.Iacovou, M.Suchak, P.Nergstrp, and J.Riedl. Grouplens: An open architecture for collaborative filtering of netnews. *In Proceedings of CSCW'94*, 1994.

[14] J. D. M. Rennie and N.Srebro. Fast maximum margin matrix factorization for collaborative prediction. *In Proceedings of ICML'05*, 2005.

[15] R.Salakhutdinov and A.Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. *In Proceedings of ICML'08*, 2008.

[16] R.Salakhutdinov and A.Mnih. Probabilistic matrix factorization. *In Advances in Neural Information Processing Systems*, 2008.

[17] Y. Shen and R. Jin. Learning personal + social latent factor model for social recommendation. *In Proceedings of SIGKDD*, pages 1303–1311, 2012.

[18] J. Tang, H. Gao, and H. Liu. mtrust: discerning multi-faceted trust in a connected world. *In Proceedings of WSDM'12*, pages 93–102, 2012.

[19] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. *In Proceedings of WWW'11*, 2011.

[20] M. Ye, X. Liu, and W.-C. Lee. Exploring social influence for recommendation: a generative model approach. *In Proceedings of SIGIR*, pages 671–680, 2012.

[21] E. Zhong, W. Fan, and Q. Yang. Contextual collaborative filtering via hierarchical matrix factorization. *In SDM*, 2012.

[22] C. Ziegler and J. Golbeck. Investigating interactions of trust and interest similarity. *Decision Support Systems.*, 2007.