

Collaborative Filtering with Topic and Social Latent Factors Incorporating Implicit Feedback

GUANG-NENG HU, XIN-YU DAI, and FENG-YU QIU, Nanjing University

RUI XIA, Nanjing University of Science and Technology

TAO LI, Florida International University

SHU-JIAN HUANG and JIA-JUN CHEN, Nanjing University

Recommender systems (RSs) provide an effective way of alleviating the information overload problem by selecting personalized items for different users. Latent factors-based collaborative filtering (CF) has become the popular approaches for RSs due to its accuracy and scalability. Recently, online social networks and user-generated content provide diverse sources for recommendation beyond ratings. Although *social matrix factorization* (Social MF) and *topic matrix factorization* (Topic MF) successfully exploit social relations and item reviews, respectively; both of them ignore some useful information. In this article, we investigate the effective data fusion by combining the aforementioned approaches. First, we propose a novel model *MR3* to jointly model three sources of information (i.e., ratings, item reviews, and social relations) effectively for rating prediction by aligning the latent factors and hidden topics. Second, we incorporate the implicit feedback from ratings into the proposed model to enhance its capability and to demonstrate its flexibility. We achieve more accurate rating prediction on real-life datasets over various state-of-the-art methods. Furthermore, we measure the contribution from each of the three data sources and the impact of implicit feedback from ratings, followed by the sensitivity analysis of hyperparameters. Empirical studies demonstrate the effectiveness and efficacy of our proposed model and its extension.

CCS Concepts: • **Information systems** → **Recommender systems**; • **Human-centered computing** → **Collaborative filtering**;

Additional Key Words and Phrases: Recommendation systems, collaborative filtering, implicit feedback, hidden topics, latent social factors

ACM Reference format:

Guang-Neng Hu, Xin-Yu Dai, Feng-Yu Qiu, Rui Xia, Tao Li, Shu-Jian Huang, and Jia-Jun Chen. 2018. Collaborative Filtering with Topic and Social Latent Factors Incorporating Implicit Feedback. *ACM Trans. Knowl. Discov. Data.* 12, 2, Article 23 (January 2018), 30 pages.

<https://doi.org/10.1145/3127873>

This work is supported by the 863 program (2015AA015406), the National Science Foundation of China (61472183, 61333014, 61672288, 91646116) and the Natural Science Foundation of Jiangsu Province for Excellent Young Scholar (BK 20160085). The authors' thank Jiliang Tang for providing the datasets and thank Chaochao Chen for providing their code. The authors' would also like to thank the reviewers for their profound comments and valuable suggestions to improve the quality of this article.

Authors' addresses: G.-N. Hu, X.-Y. Dai (corresponding author), F.-Y. Qiu, S.-J. Huang, and J.-J. Chen, National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China; emails: njuhgn@gmail.com, {daixinyu, chenjj}@nju.edu.cn, {qiufy, huangsj}@nlp.nju.edu.cn; R. Xia, School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210023, China; email: rxia@njust.edu.cn; T. Li, School of Computing and Information Sciences, Florida International University, Miami, FL 33199; email: taoli@cs.fiu.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 ACM 1556-4681/2018/01-ART23 \$15.00

<https://doi.org/10.1145/3127873>

1 INTRODUCTION

For all the benefits of the information abundance and communication technology, the “information overload” is one of the digital-age dilemmas we are confronted with. Recommender systems (RSs) are instrumental in tackling this problem (Sarwar et al. 2001; Linden et al. 2003; Adomavicius and Tuzhilin 2005; Mnih and Salakhutdinov 2007; Koren et al. 2009). They help offer potential interesting information to individual consumers and allow online users to quickly find the personalized information that fits their needs. RSs are nowadays ubiquitous in various domains and e-commerce platforms. They are used to recommend books at Amazon.com (Linden et al. 2003), music at Last.fm and Pandora.com, movies at Netflix.com (Bennett and Lanning 2007), videos at YouTube.com, and resources in scientific community (Fang and Si 2011).

Collaborative filtering (CF) approaches play a central role in traditional RSs, which are extensively investigated in research community and widely used in industry (Sarwar et al. 2001; Linden et al. 2003; Adomavicius and Tuzhilin 2005; Ekstrand et al. 2011). They are based on the simple intuition that if users rated items similarly in the past, then they will be likely to rate other items similarly in the future. Latent factors CF, which learns a latent vector of preferences for each user and a latent vector of attributes for each item, gains popularity and becomes the standard model for recommender due to its accuracy and scalability (Mnih and Salakhutdinov 2007; Koren 2008; Koren et al. 2009). CF models, however, suffer from data sparsity and the imbalance of ratings. They perform poorly on cold users and cold items for which there are no or few data.

To overcome these weaknesses, additional sources of information are integrated into RSs. Social networking and knowledge sharing sites like Twitter.com, Facebook.com, Epinions.com, and Ciao.com are popular platforms for users to connect to each other, to participate in online activities, and to generate shared content or opinions (Li et al. 2015). Social relations (Ma et al. 2008; Jamali and Ester 2011; Bao et al. 2014a; Tang et al. 2013b; Guo et al. 2015), and item reviews (Jakob et al. 2009; Ganu et al. 2009; Wang and Blei 2011; McAuley and Leskovec 2013; Bao et al. 2014b) provide independent and diverse sources for recommendation beyond the explicit rating information source, which present both opportunities and challenges for traditional RSs. One research thread, which we call *topic matrix factorization* (Topic MF), is to integrate ratings with item contents or reviews text (Jakob et al. 2009; Ganu et al. 2009; Wang and Blei 2011; McAuley and Leskovec 2013; Bao et al. 2014b). Reviews justify the rating behavior of a user; ratings are associated with item attributes that can be revealed from reviews. Topic MF methods combine latent factors in ratings with latent topics in item reviews. Another research thread, which we call *social matrix factorization* (Social MF), is to combine ratings with social relations (Ma et al. 2008, 2009; Jamali and Ester 2011; Bao et al. 2014a; Tang et al. 2013b; Guo et al. 2015). Extensive studies have found higher likelihood of establishing social ties among people having similar characteristics, namely the theory of homophily, and homophilous ties become effective means of social influence (Tang et al. 2013a). Social MF methods factorize rating matrix and social matrix simultaneously assuming that both matrices share the common latent user space. Nevertheless, both Social MF and Topic MF ignore some useful information, either item reviews or social relations.

Contrast to integrating more data sources into RSs, another way is to mine the limited data information more deeply. For example, the SVD++ (Koren 2008) model factorizes only the rating matrix and exploits the implicit feedback from it. Item ratings tell us *how* a user rated an item, i.e., an explicit rating score (e.g., from 1 to 5) to indicate her preference degree. Moreover, implicit feedback is associated with these explicit rating scores, telling us *which* items the user rated. Users chose to indicate their preferences implicitly by voting a rating, leaving the rating high or low alone. In another way, users who have rated the same/similar set of items are more likely to have similar preferences than those who have not, in an *a priori* sense. The TrustSVD (Guo et al. 2015)

model employs rating and social relation data sources and exploits the implicit feedback from both of them. These methods mines the limited data sources deeply and achieves good results comparing to the shallow mining.

A recent work (Chen et al. 2014) adopts CTR (*collaborative topic regression*, one of the Topic MF methods) (Wang and Blei 2011) to exploit ratings and reviews, and adopts SoReg (*social regularization*, one of the Social MF methods) (Ma et al. 2011) to exploit ratings and social relations. Experimental results show performance improvements compared to the two individual components. Similar methods are also proposed for tag recommendation (Wang et al. 2013) and article recommendation (Purushotham et al. 2012). These methods, however, have two limitations. First, the two components they used are not effective. For example, some improved components were proposed like HFT (*hidden factors and topics*, one of the Topic MF methods) (McAuley and Leskovec 2013) to exploit item reviews and LOCABAL (*local and global*, one of the Social MF methods) (Tang et al. 2013b) to exploit social relations. Second, they did not mine the data sources more deeply like incorporating implicit feedback from ratings, though RSs can indeed benefit from implicit information; and this benefit has already been demonstrated like in the SVD++ (Koren 2008; Koren et al. 2009) and the TrustSVD (Guo et al. 2015) models. In this article, we attempt to overcome these two drawbacks.

Contributions. First, we investigate the effectiveness of fusing social relations and reviews to rating prediction in a novel way, inspired by the complementarity of the two independent sources for recommendation. The core idea is the alignment between latent factors found by Social MF and hidden topics found by Topic MF to form a unified model. Through latent factors and hidden topics we can learn the representations of user entity and item entity from heterogenous data sources; and the connections among data sources can be reflected in the dependencies among these representations. In this way, we can gain the maximum benefits from all of the information effectively.

Second, we mine the data more deeply by incorporating implicit feedback from ratings into the proposed model to enhance its capability and to demonstrate its flexibility. The core idea is to learn an extra implicit feature matrix to consider the influence of rated items. Due to the sparseness of data, users who have few data will have latent features close to the average, leading to their predicted ratings close to the items' average. Through the implicit features, the users' rated items will have an *a priori* impact on their ratings with respect to unseen items.

Our main contributions are outlined as follows:

- Proposing a novel model *MR3* to jointly model user-item ratings, social network structure, and item reviews for rating prediction and along with an extended Social MF method, which exploits the ratings and social relations more tightly by capturing the graph structure of neighbors; the *MR3* model integrates two effective components. (Section 4)
- Extending the proposed model to obtain a new model *MR3++* by incorporating implicit feedback from ratings to enhance its capability and to demonstrate its flexibility; the extension model mines the limited information more deeply by introducing implicit features that captures the influence of rated items. (Section 5)
- Adapting an alternating optimization algorithm to learn the proposed models that contain different kinds of parameters. (Section 6)
- Evaluating the proposed models extensively on two real-world datasets to demonstrate their performance and to understand their working. (Section 7)

A preliminary version of the work, i.e., *MR3*, has been published in Hu et al. (2015). In this journal submission, we have extended our previous conference paper from the following aspects. First, we propose a new model *MR3++* by incorporating implicit feedback from ratings to mine

the limited information more deeply (Section 5). Second, we refine the preliminaries (Section 3) and abstract the learning processing (Section 6) which are suitable for both MR3 and MR3++. Third, we evaluate the newly proposed model extensively on two real-world datasets, including: (i) prediction performance (Section 7.3.2), (ii) impact of implicit feedback (Section 7.4.3) and its comparison with contribution of more data sources (Section 7.4.4), and (iii) the hyperparameters analysis (Section 7.5.2). Finally, we give further analysis of contributions from auxiliary sources of data (Section 7.4.2).

The organization of this article is as follows. We first review related works in Section 2. We introduce the notation and preliminaries in Section 3. We present details of the proposed model in Section 4, and its extension to incorporate implicit feedback in Section 5. Their learning processes are described in Section 6. In Section 7, we demonstrate our methods empirically on two real-life datasets. We give concluding remarks and discussion of some future work in Section 8.

2 RELATED WORKS

We review some related works on CF-based RSs, which are divided into four categories according to information sources they exploit among ratings, reviews, and social relations. Implicit feedback from ratings is also reviewed in the corresponding category.

Collaborative filtering. CF recommender approaches have two types: memory-based CF and model-based CF (Adomavicius and Tuzhilin 2005; Ekstrand et al. 2011). They both assume that if users rated items similarly in the past, then they will be likely to rate other items similarly in the future. Memory-based CF methods are grouped into user-based CF and item-based CF. The former predicts the rating of an active user based on the ratings of other similar users on the item (Breese et al. 1998), while the latter based on the ratings of other similar items given by the same user (Sarwar et al. 2001; Linden et al. 2003). And the two kinds of memory-based CF can be unified by similarity fusion in a generative probabilistic model (Wang et al. 2006). The similarity between two users or two items can be measured by Pearson correlation or cosine similarity from past rating history. Model-based CF learns a model from past ratings and then uses the learned model to predict unseen ratings. Latent semantics models (Hofmann 2004) are some early representative model-based CF. And latent factors CF or MF-based CF, which learns a latent vector of preferences for each user and a latent vector of attributes for each item, gains popularity and becomes the standard model for recommender due to its accuracy and scalability (Koren et al. 2009; Mnih and Salakhutdinov 2007). These MF techniques include maximum margin MF (Rennie and Srebro 2005), nonnegative MF (Gu et al. 2010), and collective MF (Singh and Gordon 2008). A relaxation on the assumption that interprets ratings as numerical values is to interpret them as ordinal ones (Koren and Sill 2011).

Implicit feedback from ratings can readily be added into the basic MF-based CF models (Koren 2008; Mnih and Salakhutdinov 2007; Rendle 2010). The intuition behind the implicit information from ratings is that users who have rated the same/similar items are more likely to have similar preferences than those who have not, in an *a priori* sense. Another way to exploit implicit feedback from ratings is to deduce the preference-confidence pairs from the raw observed ratings (Hu et al. 2008). The observed rating data is treated as an indication of positive and negative preferences associated with varying confidence levels. The resulting objective function sums over all the full matrix entries rather than over the only observed ones (Liu et al. 2010). CF models, however, suffer from data sparsity and the imbalance of ratings; and they perform poorly on cold users and cold items for which there are no or few data. Currently, there are mainly two threads to alleviate these problems: topic MF integrating reviews text information and social MF integrating social network information.

Topic matrix factorization. One research thread, which we call Topic MF, is to integrate ratings with item contents or reviews text (Wang and Blei 2011; Gopalan et al. 2014; McAuley and Leskovec 2013; Ling et al. 2014; Bao et al. 2014b). Early works (Jakob et al. 2009; Ganu et al. 2009) extract the fine-grained ratings of item aspects from online reviews and then use them as content-based features for CF. Aspects of a movie have actors, genres, and visual effects; aspects of a restaurant have price, cleanness, and service; and aspects of a hotel have location, cleanliness, and room view. The extraction of aspects needs domain knowledge and some amount of manual interaction. Besides item reviews, topic modeling approaches have been incorporated into RSs to find the hidden topics of documents like scientific articles (Blei et al. 2003; Wang and Blei 2011; Gopalan et al. 2014). Such methods are belonging to One-Class CF (Pan et al. 2008), where the dimensions they discover are not correlated with ratings.

In recent works, some authors adopt *latent Dirichlet allocation (LDA)* (McAuley and Leskovec 2013) or *nonnegative matrix factorization (NMF)* (Bao et al. 2014b) to learn latent topic factors from item reviews and meanwhile adopt a MF model to exploit the ratings. To bridge the topic-specific factors and rating-specific factors, softmax transformations/exponential functions are proposed to link the two. These methods assume that the dimensionality discovered in ratings is the same as that found in reviews. Ling et al. (2014) replaced the MF model with a mixture of Gaussian to avoid the difficult choice of the transformations; Xu et al. (2014) used the co-clustering of user community and item group to generate the rating distributions and topic distributions, allowing the different dimensionality of user factors and item factors; and Diao et al. (2014) introduced the aspect-based model to link the interest distribution of users and the content distribution for movies, allowing the dimensionality of the two to be different. In general, Topic MF methods combine latent factors in ratings with latent topics in item reviews. Nevertheless, Topic MF ignores some useful information, e.g., social relations.

Social matrix factorization. Another research thread, which we call Social MF, is to combine ratings with social relations (Ma et al. 2008, 2011; Jamali and Ester 2011; Tang et al. 2013b; Bao et al. 2014a; Guo et al. 2015). Early work (Massa and Avesani 2007) computed the trust value between users from the trust network to replace the similarity weight in user-based CF. Empirical results show that RSs incorporating trust information are effective in terms of accuracy while alleviating the cold-user problem. It's a memory-based CF or trust-related neighborhood model and does not study the rating matrix and the trust network systematically.

In recent works, the authors (Ma et al. 2008; Chaney et al. 2015; Tang et al. 2013b; Yang et al. 2013; Bao et al. 2014a; Guo et al. 2015) factorize the rating matrix and the social matrix simultaneously assuming that they share the common latent user space. They exploit the social relations from multiple views: (1) local and global (LOCABAL) view (Tang et al. 2013b) where *local* perspective reveals the correlations between the user and her local neighbors, while *global* perspective indicates the reputation of users in the global network; (2) trustee and truster (TrustMF) view (Yang et al. 2013) where the *trustee* model captures how others follow the rating of a user while the *truster* model captures how other users affect the rating of a user; and (3) decomposed trust view (Bao et al. 2014a) where four trust aspects (benevolence, integrity, competence, and predictability) are formulated to predict the trust values and the predicted trust is combined with the similarity of the latent user feature vectors to get the total trust between two users.

Another way to exploit social network information is to use it as a social regularization (SoReg), which constrains that the latent factors of users should be close to the average of their trusted neighbors (Ma et al. 2011). A similar work to this is also proposed in Jamali and Ester (2011) to allow trust propagation. Nevertheless, Social MF ignores some useful information, e.g., reviews text.



Fig. 1. Three types of recommendation information sources. On the Ciao and Epinions datasets, there are three kinds of data sources, i.e., rating scores, item reviews, and social relations.

Hybrid recommender approaches. There is a tendency towards hybrid approaches. The authors in Fang and Si (2011) propose the matrix co-factorization techniques to exploit the user and the item side information for one-class CF (Pan et al. 2008); their objective is to minimize the reconstruction loss of both user-word and item-word TFIDF weight matrix. A recent work to exploit the three types of information for recommendation (Chen et al. 2014) adopts CTR (*collaborative topic regression*, one of the Topic MF methods) (Wang and Blei 2011) to exploit ratings and reviews, and adopts SoReg (*social regularization*, one of the Social MF methods) (Ma et al. 2011) to exploit ratings and social relations. Experimental results show better performance compared to the two individual components. Similar methods are also proposed for tag recommendation (Wang et al. 2013), celebrity recommendation (Ding et al. 2013), and article recommendation (Purushotham et al. 2012). However, these models have two drawbacks. First, the two components they used are not effective and some better components were proposed like the HFT (*hidden factors and topics*, one of the Topic MF methods) model (McAuley and Leskovec 2013) and the LOCABAL (*local and global*, one of the Social MF methods) model (Tang et al. 2013b). Second, they did not mine the data sources more deeply like exploiting implicit feedback from ratings, though RSs can indeed benefit from implicit information as demonstrated by the SVD++ model and the TrustSVD model. In this article, we attempt to overcome these two drawbacks.

3 PRELIMINARIES

Before proposing our models, we review briefly the representative approaches to exploit the three types of information individually. To this end, we first introduce the notations related to the three data sources shown in Figure 1.

Notations. Suppose there are M users $\mathcal{P} = \{u_1, \dots, u_M\}$ and N items $\mathcal{Q} = \{i_1, \dots, i_N\}$. We reserve u, v, w to index the users, and i, j, k to index the items. Let $R \in \mathbb{R}^{M \times N}$ denote the rating matrix, where the entry $R_{u,i}$ is the rating of user u on item i , and we mark a zero if it is unknown. The task of rating prediction is to predict the unknown/missing ratings from the observed data.

In addition to this explicit rating information, other side information sources may exist. One such source is social relations. Users connect to others in a social network, where a link between two users indicates their friendship or trust relation. We use $T \in \mathbb{R}^{M \times M}$ to indicate the user-user social relations, where the entry $T_{u,v} = 1$ if user u has a relation to user v or zero otherwise. Another side data source is item reviews. Items have affiliated content information, e.g., reviews

Table 1. Notations

Symbol	Meaning	Form
M, N	The number of users, and of items	Scalar
L	The size of the word vocabulary	Scalar
\mathcal{P}, \mathcal{Q}	The set of users, and of items	Set
N_d	The set of words in doc d	Set
N_u	The set of items rated by the user u	Set
T_u	The set of users trusted by the user u	Set
$R_{u,i}, R_{u,i}^b$	Rating of item i by user u , and its implicit binary rating	$R \in \mathbb{R}^{M \times N}, R^b \in \{0, 1\}^{M \times N}$
$T_{u,v}$	Social relation between user u and v	$T \in \mathbb{N}^{M \times M}$
$w_{d,n}$	The n^{th} word in doc d	Doc-term matrix, $w \in \mathbb{N}^{N \times L}$
$\bar{W}_{u,i}$	Weight on the rating of item i given by user u	Pre-computed, $W \in \mathbb{R}^{N \times N}$
$C_{u,v}$	Social strength (trust value) between user u and v	Pre-computed, $C \in \mathbb{R}^{M \times M}$
$S_{u,v}$	Social similarity between user u and v	Pre-computed, $S \in \mathbb{R}^{M \times M}$
F	Dimensionality of latent factors/topics	Hyper-parameter, scalar
P_u	F -dimensional feature vector for user u	Parameters, $P \in \mathbb{R}^{F \times M}$
Q_i	F -dimensional feature vector for item i	Parameters, $Q \in \mathbb{R}^{F \times N}$
Y_j	F -dimensional implicit feature vector for item j	Parameters, $Y \in \mathbb{R}^{F \times N}$
θ_i	F -dimensional topic distribution for item i	Parameters, $\theta \in \Delta^{F \times N}$
ϕ_f, ψ_f	Word distribution for topic f , and the unnormalized one	Parameters, $\phi \in \Delta^{L \times F}$
H	Social correlation matrix	Parameters, $H \in \mathbb{R}^{F \times F}$

commented by users. The observed review data $d_{u,i}$ is a piece of text of item i written by user u , often along with a rating score $R_{u,i}$.¹

Notations used throughout the article are summarized in Table 1.

Rating information. For the information source of ratings, MF-based latent factor models (Mnih and Salakhutdinov 2007; Koren et al. 2009) are mainly to find the latent user-specific feature matrix $P = [P_1, \dots, P_M] \in \mathbb{R}^{F \times M}$ and item-specific feature matrix $Q = [Q_1, \dots, Q_N] \in \mathbb{R}^{F \times N}$ to approximate the observed rating matrix in the least-squares sense (more precisely, regularized least squares or ridge regression), obtained by solving the following problem

$$\min_{P, Q} \sum_{R_{u,i} \neq 0} (R_{u,i} - \hat{R}_{u,i})^2 + \lambda (\|P\|_{Fro}^2 + \|Q\|_{Fro}^2), \quad (1)$$

where λ is the regularization parameter to avoid over-fitting and the predicted ratings, and $\|\cdot\|_{Fro}^2$ denotes the Frobenius norm.

$$\hat{R}_{u,i} = \mu + b_u + b_i + P_u^T Q_i. \quad (2)$$

The parameters μ , b_u , and b_i are the mean of ratings, bias of the user and bias of the item, respectively. The F -dimensional feature vectors P_u and Q_i represent users' preferences and items' characteristics. The dot products capture their interaction or match degree.

¹We organize these reviews as a document-term matrix $w \in \mathbb{N}^{N \times L}$ where the entry $w_{d,n}$ is the occurrence of token n in doc d . For convenience, we also refer the review as $d_{u,i}$.

Social information. For the information source of social relations, social MF methods (Zhu et al. 2007; Tang et al. 2013b) are mainly to find the latent social-specific feature matrix P and social correlation matrix $H \in \mathbb{R}^{F \times F}$ to approximate the observed social similarity matrix $S \in \mathbb{R}^{M \times M}$ in the least-squares sense, by solving the following problem²:

$$\min_{P, H} \sum_{T_{u,v} \neq \emptyset} \left(S_{u,v} - P_u^T H P_v \right)^2 + \lambda \left(\|P\|_{Fro}^2 + \|H\|_{Fro}^2 \right), \quad (3)$$

where $S_{u,v}$ is the social similarity between user u and her trustee v , defined as the cosine similarity between their rating vectors

$$S_{u,v} = \frac{\sum_i R_{u,i} \cdot R_{v,i}}{\sqrt{\sum_i R_{u,i}^2 \cdot \sum_i R_{v,i}^2}}. \quad (4)$$

The assumption to make the above method work is that the latent social-specific matrix is shared with the latent user-specific matrix; both are referred as P here. Namely, users have dual identity: one is involved in rating behavior and the other is involved in social behavior.

Review information. For the information source of reviews, topic modeling approaches are used to find the item properties and hidden topics (Blei et al. 2003; McAuley and Leskovec 2013). The negative log-likelihood (NLL) of the reviews collection is defined as³

$$- \sum_{d=1}^N \sum_{n \in N_d} \left(\log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}} \right), \quad (5)$$

where the parameters θ and ϕ are the topic and word distributions, respectively; $w_{d,n}$ and $z_{d,n}$ are the word and the corresponding topic in doc d . Reviews explain the ratings why the users rate in that way. Intuitively, a certain property of the item is probably discussed by a specific distribution of words that corresponds to a certain topic. The following softmax transformation sharpens this intuition and also bridges the gap between the real-valued parameters $Q_i \in \mathbb{R}^F$ associated with ratings and the corresponding probabilistic ones $\theta_i \in \Delta^F$ associated with reviews.

$$\theta_{i,f} = \frac{\exp(\kappa Q_{i,f})}{\sum_{f=1}^F \exp(\kappa Q_{i,f})}, \quad (6)$$

where the parameter κ controls “peakiness” of the transformation and can be jointly learned with other parameters.

The dependencies among data matrices and parameter matrices in three different recommendation approaches are shown in Figure 2. The left subfigure is latent factors CF approach to exploit the rating information; the middle subfigure is social MF approach to integrate ratings with social relations; and the right subfigure is topic MF approach to combine ratings with item reviews.

4 THE PROPOSED MODEL

In this section, we propose a model to solve the following problem. We call it Problem 1, which requires to model three data sources simultaneously.

²We omit the term about the ratings, which is the same as that in Equation (1), to show clearly how to exploit the social relations.

³We aggregate all reviews of a particular item as a “doc;” so the indices of docs are corresponding to those of items.

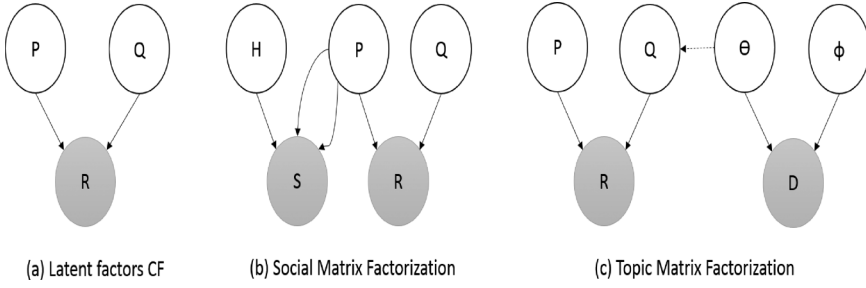


Fig. 2. Illustrations of the dependencies among data matrices and parameter matrices in three different recommendation approaches. (a) Latent factors CF exploits ratings, (b) Social MF integrates ratings with social relations; and (c) Topic MF integrates ratings with reviews text. Shaded nodes are data and others are parameters.

4.1 Problem 1

PROBLEM 1. *Rating Prediction with Social Relations and Reviews.*

Input: (1) a rating matrix R , (2) a social network among users T , (3) a reviews collection along with the ratings D , (4) a user u in the user set \mathcal{P} , and (5) an item i in the item set \mathcal{Q} .

Output: the predicted preference of user u on item i , where $u \in \mathcal{P}$ and $i \in \mathcal{Q}$.

In Problem 1, to predict the unknown ratings, we have three types of information to exploit: ratings, social relations, and item reviews.

4.2 MR3: A Model of Ratings, Item Reviews, and Social Relations

In Section 3, we have reviewed the three kinds of approaches to exploit the three kinds of information sources individually (see Figure 2), i.e., MF-based CF for the information source of ratings (see Equation (1), or Figure 2(a)), social MF for the information source of social relations (see Equation (3), or Figure 2(b)), and the topic MF for the information source of item reviews (see Equation (5), or Figure 2(c)). With these preliminaries in mind, we can present our solution to the Problem 1, exploiting all of data sources simultaneously.

For Problem 1, the main challenge is how to fuse effectively the three heterogeneous data sources to form a unified model. We tackle this challenge by combining the two parts, i.e., Social MF and Topic MF, described in the following. The core idea is, based on the CF, the alignment between latent factors and hidden topics found by the above two parts.

In one part, the LOCABAL (*local and global*, one of the Social MF methods) model (Tang et al. 2013b) exploits the ratings and social relations by incorporating social latent factors into CF. The goals to achieve are modeling ratings accurately and also capturing the social context, by solving the following problem⁴:

$$\min_{P, Q, H} \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i})^2 + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta), \quad (7)$$

where the rating weights $W_{u,i} = 1/(1 + \log r_u)$ are computed from the PageRank scores of users in the social network, representing the global perspective of social context (Tang et al. 2013b), where r_u is the rank of user u in decreasing order, i.e., the top-ranked users having high ranking scores. The regularization parameter λ_{rel} controls the contribution from social relations, parameters $\Theta = \{P, Q, H\}$, and the regularization term is given by $\Omega(\Theta) = \|P\|_{Fro}^2 + \|Q\|_{Fro}^2 + \|H\|_{Fro}^2$.

⁴The global social context $W_{u,i}$ is omitted here for clarity, and given in Equation (9) instead.

In another part, the HFT (*hidden factors and topics*, one of the Topic MF methods) model (McAuley and Leskovec 2013) exploits the ratings and item reviews by incorporating topic latent factors into CF. The goals to achieve are both modeling ratings accurately and generating reviews likely, by solving the following problem:

$$\min_{P, Q, \Phi} \sum_{R_{u,i} \neq 0} (R_{u,i} - \hat{R}_{u,i})^2 - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} (\log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}}), \quad (8)$$

where λ_{rev} controls the contribution from reviews, and parameters $\Phi = \{\theta, \phi\}$. The connection between ratings and reviews is the coupling between parameters Q_i and θ_i as shown in Equation (6).

Based on CF, we can exploit the three data sources simultaneously by combining the above two parts (i.e., Equation (7) and Equation (8)). By aligning latent factors and topics, we connect Social MF and Topic MF through the proposed model *MR3 (model of rating, relation, and review)*, which minimizes the following problem (Hu et al. 2015):

$$\begin{aligned} \mathcal{L}(\Theta, \Phi, z, \kappa) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i})^2 \\ & - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} (\log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}}) \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta), \end{aligned} \quad (9)$$

Beyond the LOCABAL model, we borrow the idea *trust values* from the SoRec method (Ma et al. 2008) to exploit the ratings and social relations more tightly

$$C_{u,v} = \sqrt{d_v^- / (d_u^+ + d_v^-)}, \quad (10)$$

where the outdegree d_u^+ represents the number of users whom u trusts, while the indegree d_v^- denotes the number of users who trust v . The trust values are used to capture the graph structure of neighbors, representing the social influence locality, i.e., user behaviors are mainly influenced by close/direct friends in their ego networks (Zhang et al. 2013),

Before we delve into the learning algorithm, a brief discussion on Equation (9) is in order. On the right hand, the first term is rating squared-error weighted by user reputation in the social network; the second term is the NLL of item reviews corpus; the third term is local social context factorization weighted by trust values among users; and the last term is Frobenius norm penalty of parameters to control over-fitting. The contributions from item reviews and social relations are controlled by the two hyper-parameters λ_{rev} and λ_{rel} . The connection between ratings and social relations is through the shared user latent feature space P , ratings and reviews are linked through the transformation involving Q and θ as shown in Equation (6), where parameter κ controls the peakiness of the transformation, and all of the three data sources are jointly modeling based on CF with topic and social latent factors. The dependencies among the parameter and data matrices in the proposed model are depicted in Figure 3. Note that the dotted line between Q and θ indicates the fixed transformation between them (see Equation (6)) and does not exhibit any distribution dependency; hence the figure does not describe a truly Bayesian generative model on the whole and the purpose is to clearly display the relationship among matrices of parameters and data.

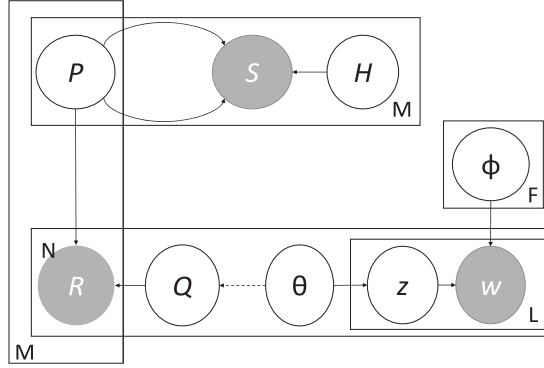


Fig. 3. Relationship among matrices of parameters and data of the proposed model (MR3). Shaded nodes are data (R : rating matrix, S : social similarity matrix, and D (here it is embodied by words w): doc-term matrix of reviews). Others are parameters (P : matrix of latent user factors, Q : matrix of latent item factors, H : social correlation matrix, θ : doc-topic distributions, and ϕ : topic-word distributions). The double connections between P and S are indicated by the term $(S - P^T H P)$ in Equation (3). The dotted line between Q and θ indicates their coupling through the fixed transformation as shown in Equation (6).

eSMF. We separate the following part from the proposed model MR3 and denote it as the *extended Social MF* (eSMF) model (Hu et al. 2015)

$$\begin{aligned} \mathcal{L}(\Theta) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i})^2 \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta). \end{aligned} \quad (11)$$

The eSMF model is the Social MF component of MR3, which extends the LOCABAL model (Tang et al. 2013b) by capturing the graph structure of neighbors via trust values (Ma et al. 2008) representing the social influence locality (Zhang et al. 2013).

5 AN EXTENSION OF THE PROPOSED MODEL

In this section, we incorporate the implicit feedback from ratings into the proposed model to enhance its capability and demonstrate its flexibility, leading to a solution to the following problem. We call it Problem 2, which requires to mine the limited information more deeply.

5.1 Problem 2

For item ratings information, they tell us *how* a user rated an item, i.e., an explicit rating score (e.g., from 1 to 5) to indicate her preference degree. Moreover, implicit feedback is always associated with these explicit rating scores, telling us *which* items the user rated. In detail, a binary matrix R^b can be constructed from the rating matrix, where the entry has one if the corresponding one in the rating matrix is observed and zero if it is unseen. Users chose to indicate their preferences implicitly by voting a rating, leaving the rating high or low alone. In another way, users who have rated the same/similar items are more likely to have similar preferences than those who have not, in an *a priori* sense.

With the above additional consideration, we extend the Problem 1 to reach the following problem.

PROBLEM 2. *Problem 1 with Implicit Feedback from Ratings.*

Input: (1) a rating matrix R and its implicit matrix R^b , (2) a social network among users T , (3) a reviews collection along with the ratings D , (4) a user u in the user set \mathcal{P} , and (5) an item i in the item set \mathcal{Q} .

Output: the predicted preference of user u on item i , where $u \in \mathcal{P}$ and $i \in \mathcal{Q}$.

By Problem 2, we define the meaning of “mine the limited information more deeply” as “exploit the implicit feedback constructed from explicit ratings.” That is, besides learning the latent user factors and item factors from ratings, we also learn an implicit feature vector for each item. As we will see in Equation (12), the implicit feedback from ratings can be used as the prior preferences of users.

5.2 MR3++: An Extension of the Proposed Model Incorporating Implicit Feedback from Ratings

In Section 4.2, we have introduced the solution to the Problem 1. The solution is the model MR3 (see Equation (9)), which exploits the three types of information sources simultaneously by connecting the Social MF approach and the Topic MF approach. In this subsection, we extend the proposed model, leading a solution to Problem 2.

The core idea of mining ratings deeply is to learn an extra implicit feature matrix $Y \in \mathbb{R}^{F \times N}$ to consider the influence of rated items. Due to the sparseness of data, users who have few data will have latent features close to the average, leading to their predicted ratings close to the items’ average. Through the implicit features, the users’ rated items will have an *a priori* impact on their ratings with respect to unseen items. In more detail, the total preferences of user u on item i decompose into two parts: one part indicates some “intrinsic preference” reflecting in her latent feature vector P_u ; and another part shows the “influence of her rated items” captured by their implicit feature vectors Y_j , where the index j denotes the set of the user’s rated items. These ideas are shaped in the SVD++ model (Koren 2008) and the constrained PMF model (Mnih and Salakhutdinov 2007), where the predicted ratings are now computed by (rather than by Equation (2))

$$\hat{R}_{u,i}^* = P_u^T Q_i + \left(|N_u|^{-\frac{1}{\alpha}} \sum_{j \in N_u} Y_j \right)^T Q_i + \mu + b_u + b_i, \quad (12)$$

where N_u is the set of items rated by the user u , i.e., $N_u = \{j : R_{u,j}^b = 1\}$ and Y_j is the implicit feature vector for item j . In the SVD++ model, $\alpha = 2$; in the constrained PMF model, $\alpha = 1$. We can see, for example, if user u has rated the same items as those rated by user v , i.e., $N_u = N_v$, then in an *a priori* sense, these two users are likely to have similar preferences.

The extended model contains two components: one component is to combine three kinds of data sources (ratings, social relations, and reviews), i.e., the MR3 model; and another is to incorporate implicit feedback from ratings and hence mining the rating source more deeply. Hence we call the extended model MR3++, where “++” stands for extending the MR3 model by exploiting implicit feedback from ratings. MR3++ minimizes the following problem:

$$\begin{aligned} \mathcal{L}(\Theta^*, \Phi, z, \kappa) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i}^*)^2 \\ & - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} (\log \theta_{z,d,n} + \log \phi_{z,d,n,w_{d,n}}) \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta^*), \end{aligned} \quad (13)$$

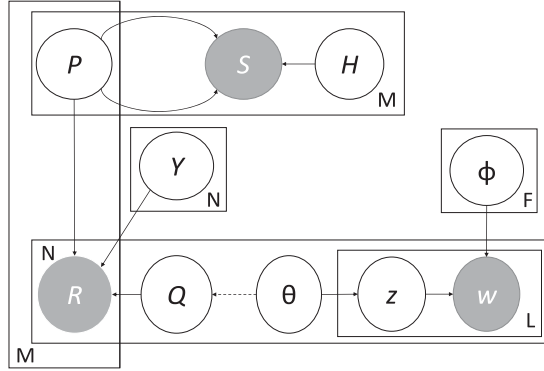


Fig. 4. The Extension of the Proposed Model (MR3++). Shaded nodes are data and others are parameters. Implicit feature matrix $Y \in \mathbb{R}^{F \times N}$ is added to consider the influence of rated items. The total preferences of a user on an item now decompose into two parts: one part indicates “intrinsic preference” reflecting in her latent feature vector; and another part shows the “influence of her rated items” captured by their implicit feature vectors.

where $\hat{R}_{u,i}^*$ is given by Equation (12) (rather than Equation (2)), and $\Omega(\Theta^*) = \|P\|_{Fro}^2 + \|Q\|_{Fro}^2 + \|H\|_{Fro}^2 + \|Y\|_{Fro}^2$.

The dependencies among the parameter and data matrices in the extension model are depicted in Figure 4, where the implicit feature matrix is added compared to the proposed model.

Implicit feature matrix Y. Note that the idea of learning feature matrix Y originally (to the best of our knowledge) comes from the NSVD model (Paterek 2007) to decrease the number of parameters, where Y is used to replace the learning of latent user feature matrix P ; hence the complexity of parameters is from $O(MK + NK)$ to $O(NK)$ and does not depend on the scale of users. Besides capturing implicit feedback, Y has the effect of learning the item-item similarity $sim(i, j) = Y_j^T Q_i$. The predicted rating is computed by $R_{u,i} = \sum_{j \in N_u} sim(i, j) + b_u + b_i$. This method is modified to exclude the known rating information for a given user-item pair when estimating the similarity matrix in the FISM model (Kabbur et al. 2013). The estimated rating is computed as $R_{u,i} = \sum_{j \in N_u \setminus \{i\}} sim(i, j) + b_u + b_i$.

To mine the implicit feedback from ratings, we adopt the SVD++ model idea in our proposed model MR3++: learning both the item-item similarity $sim(i, j) = Y_j^T Q_i$ and the user-item similarity $sim(u, i) = P_u^T Q_i$. And hence the predicted ratings are computed by $R_{u,i} = sim(u, i) + |N_u|^{-\frac{1}{2}} \sum_{j \in N_u \setminus \{i\}} sim(i, j) + \mu + b_u + b_i$, which is just the same with Equation (12). Combined with the FISM model idea, we can also adopt the following rating predictor in our MR3++ model:

$$R_{u,i} = sim(u, i) + (|N_u| - 1)^{-\alpha} \sum_{j \in N_u \setminus \{i\}} sim(i, j) + \mu + b_u + b_i,$$

where α is an adjustable hyper-parameter between 0 and 1.

No matter what kind of implicit predictor is used (NSVD, SVD++, or FISM), the idea of ming the limited data more deeply is worth keeping in mind. And this is a meaningful point for extending the MR3 model to the MR3++ model.

6 MODEL LEARNING

We give the optimization algorithms to learn the models (namely MR3 and MR3++) proposed in the above two sections (MR3 in Section 4 and MR3++ in Section 5). Their learning processes are the same except the minor difference with respect to gradients of two parameters.

6.1 Learning Process

Our objective is to search

$$\arg \min_{\Theta, \Phi, z, \kappa} \mathcal{L}(\Theta, \Phi, z, \kappa). \quad (14)$$

Observe that parameters $\Theta = \{P, Q, H\}$ and $\Phi = \{\theta, \phi\}$ are coupled through the transformation between Q and θ as shown in Equation (6) (the dotted line shown in Figure 3). The former parameters Θ are associated with ratings and social relations, which can be found by gradient descent methods, while the latter parameters Φ are associated with reviews text, which can be found by Gibbs sampling (Griffiths and Steyvers 2004). So, similar to the HFT (hidden factors and topics, one of the Topic MF methods) model [38], we design a procedure alternating between the following two steps:

$$\text{update } \Theta^{\text{new}}, \Phi^{\text{new}}, \kappa^{\text{new}} = \arg \min_{\Theta, \Phi, \kappa} \mathcal{L}(\Theta, \Phi, z^{\text{old}}, \kappa); \quad (15a)$$

$$\text{sample } z_{d,n}^{\text{new}} \text{ with probability } p(z_{d,n}^{\text{new}} = f) = \phi_{f,w_{d,n}}^{\text{new}}. \quad (15b)$$

For the first step as shown in Equation (15a), topic assignments $z_{d,n}$ for each word in reviews corpus are fixed; then we update the terms Θ, Φ , and κ by gradient descent (GD). Recall that θ and Q depend on each other; we fit only Q and then determine θ by Equation (6). This is the same as in the standard gradient-based MF for recommender (Mnih and Salakhutdinov 2007) except that we have to compute more gradients, which will be given later separately.

For the second step as shown in Equation (15b), parameters associated with reviews corpus θ and ϕ are fixed; then we sample topic assignments $z_{d,n}$ by iterating through all docs d and each word within, setting $z_{d,n} = f$ with probability proportion to $\theta_{d,f} \phi_{f,w_{d,n}}$. This is similar to updating z via LDA (Blei et al. 2003) except that topic proportions θ are not sampled from a Dirichlet prior, but instead are determined in the first step through Q .

Finally, the two steps are repeated until a local optimum is reached. In practice, we sample topic assignments every 5 GD epoches and this is called a pass; usually it is enough to run 50 passes to find a local minima. The experimental settings are detailed in the experimental section.

For MR3++, the learning process is the same with that of MR3, except that MR3++ has more gradients to compute and other gradients related to the implicit feature matrix Y have to modify accordingly. The learning algorithm is summarize in Algorithm 1.

6.2 Gradients of the Parameters

We now give gradients used in Equation (15a). (Gradients of biases are omitted; rating mean is not fitted because ratings are centered before training.) Besides the notations given in Table 1, more notations are required (Griffiths and Steyvers 2004) to learn the models.

- For each item i (i.e., doc i , generated from all reviews commented on this item): (1) M_i is an F -dimensional count vector, in which each component is the number of times each topic occurs for it; (2) m_i is the number of words in it; and (3) $z_i = \sum_f \exp(\kappa Q_{if})$ is a normalizer.
- For each word w (in the prescribed word vocabulary): (1) N_w is an F -dimensional count vector, in which each component is the number of times it has been assigned to each topic; (2) n_f is the number of times topic f occurs for it; and (3) $z_f = \sum_w \exp(\psi_{fw})$ is a normalizer.

ALGORITHM 1: Learning Process of the Proposed Models

Input: Ratings R , reviews w , and relations T , number of latent factors F , maximum number of iterations $maxIter$, number of epoches $nEpoch$.

Output: User features P , item features Q , social relation matrix H , implicit features Y , topic proportions θ , and topic distributions ϕ .

Pre-compute W , S , and C by Equation (4.2), (4), and (10);
Initialize P, Q, H, Y randomly from $\mathcal{N}(0, 0.01)$;
 $iter = 1$;
repeat
 for $epoch = 1; epoch < nEpoch; epoch ++$ **do**
 update $\Theta^{new}, \Phi^{new}, \kappa^{new} = \arg \min_{\Theta, \Phi, \kappa} \mathcal{L}(\Theta, \Phi, z^{old}, \kappa)$ by SGD using gradients Equations (16)–(22);
 end
 sample $z_{d,n}^{new}$ with probability $p(z_{d,n}^{new} = f) = \phi_{f,w_{d,n}}^{new}$;
 $iter ++$;
until convergence (e.g., $iter > maxIter$ or $\frac{|\mathcal{L}^{iter} - \mathcal{L}^{iter+1}|}{|\mathcal{L}^{iter}|} < 10^{-6}$);

The user-specific feature matrix P exists in three terms: the first is in the rating prediction, the second is in the local social context, and the third is in the Frobenius norm penalty.

$$\begin{aligned} \frac{1}{2} \frac{\partial \mathcal{L}}{\partial P_u} = & \sum_{i:R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i} - R_{u,i}) Q_i + \lambda P_u \\ & + \lambda_{rel} \sum_{v:T_{u,v} \neq 0} C_{u,v} (P_u^T H P_v - S_{u,v}) H P_v \\ & + \lambda_{rel} \sum_{v:T_{v,u} \neq 0} C_{v,u} (P_v^T H P_u - S_{u,v}) H^T P_v. \end{aligned} \quad (16)$$

The item-specific feature matrix Q exists in three terms: the first is in the rating prediction, the second is in the item reviews, and the third is in the Frobenius norm penalty.

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial Q_i} = & 2 \sum_{u:R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i} - R_{u,i}) P_u \\ & - \lambda_{rev} \kappa \left(M_i - \frac{m_i}{z_i} \exp(\kappa Q_i) \right) + 2\lambda Q_i. \end{aligned} \quad (17)$$

The social correlation matrix H exists in two terms: one is in the local social context, and the other is in the Frobenius norm penalty.

$$\frac{1}{2} \frac{\partial \mathcal{L}}{\partial H} = \lambda_{rel} \sum_{T_{u,v} \neq 0} C_{u,v} (P_u^T H P_v - S_{u,v}) P_u P_v^T + \lambda H. \quad (18)$$

The parameter κ controls the peakiness between Q and θ as shown in Equation (6). The differentiation of \mathcal{L} over κ is followed by the chain rule: \mathcal{L} over θ , and then θ over κ .

$$\frac{\partial \mathcal{L}}{\partial \kappa} = -\lambda_{rev} \sum_{i,f} Q_{if} \left(M_{if} - \frac{m_i}{z_i} \exp(\kappa Q_{if}) \right). \quad (19)$$

Note that ϕ_f is a stochastic vector, so we optimize the corresponding unnormalized vector ψ_f and then get $\phi_{fw} = \exp(\psi_{fw})/z_f$. The unnormalized word distributions exist in the likelihood

term of the reviews corpus.

$$\frac{\partial \mathcal{L}}{\partial \psi_{fw}} = -\lambda_{\text{rev}} \left(N_{fw} - \frac{n_f}{z_f} \exp(\psi_{fw}) \right). \quad (20)$$

For MR3++, the gradients of the Y and Q are given below, and others are the same.

The gradients of the added implicit feature parameters Y_j are computed by

$$\frac{1}{2} \frac{\partial \mathcal{L}}{\partial Y_j} = \sum_{R_{u,i} \neq 0} W_{u,i} |N_u|^{-\frac{1}{2}} (\hat{R}_{u,i}^* - R_{u,i}) Q_i + \lambda Y_j, \quad \forall j \in N_u. \quad (21)$$

The gradients of the original parameters Q_i are now computed by (an extra term with respect to implicit ratings Y is added to P , compared with that in Equation (17))

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial Q_i} = & 2 \sum_{u:R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i}^* - R_{u,i}) \left(P_u + |N_u|^{-\frac{1}{2}} \sum_{j \in N_u} Y_j \right) \\ & - \lambda_{\text{rev}} \kappa \left(M_i - \frac{m_i}{z_i} \exp(\kappa Q_i) \right) + 2\lambda Q_i. \end{aligned} \quad (22)$$

7 EXPERIMENTS

In the above Section 4.2 and Section 5.2, we have introduced the solutions to Problem 1 and to Problem 2, respectively. The solution to Problem 1 is the result of the proposed model MR3 (See Equation (9)), which exploits all three types of information simultaneously. The solution to Problem 2 is the model MR3++ (see Equation (13)), which extends the MR3 model by incorporating implicit feedback from ratings.

In this section, we first compare our proposed eSMF model with a state-of-the-art Social MF method to show the benefit of exploiting the graph structure of neighbors via trust values that capture the social influence locality. Second, we demonstrate the effectiveness of the proposed model MR3 and the improvement of its extension MR3++ over various different recommender approaches. Further, we design experiments to see the contribution of each data source to the proposed model and the impact of the implicit feedback, followed by sensitivity analysis of our models to three meta parameters.

7.1 Dataset and Metric

In this section, we first introduce the two datasets used to evaluate the recommendation performance, including the simple preprocessing and basic statistics. And then we describe the performance metric and the evaluation protocol.

7.1.1 Datasets and Statistics. We evaluate our models on two datasets: Epinions and Ciao.⁵ They are both knowledge sharing and review sites, in which users can rate items, connect to others, and give reviews on products (see Figure 1). We remove stop words⁶ and then select top $L = 8,000$ frequent words as vocabulary; to avoid noise we reserve users who rated more than three times and remove items that were rated by only once or twice. The items indexed in rating matrix are aligned to documents in doc-term matrix, that is, we aggregate all reviews of a particular item as a “doc.” Statistics of datasets are given in Table 2. We see that the rating matrices of both datasets are very sparse, and the average length of documents is short on Epinions.

Note that the number of average words per item on Ciao is 42 times longer than that on Epinions, and the social density on Ciao is 12 times denser than that on Epinions. So Ciao contains richer and

⁵<http://www.public.asu.edu/~jtang20/>.

⁶<http://www.ranks.nl/stopwords>.

Table 2. Statistics of the Datasets

Statistics	Epinions	Ciao	Total
# of Users	49,454	7,340	56,794
# of Items	74,154	22,472	96,626
# of Ratings/Reviews	790,940	183,974	974,914
# of Social Relations	434,680	112,942	547,622
# of Words	2,246,837	28,874,000	31,120,837
Rating density	0.00022	0.0011	–
Social density	0.00018	0.0021	–
Average words per item	30.3	1284.9	–
Average relations per user	8.78	15.38	–

higher quality information in social relations and reviews. Hence we expect that social relations and reviews contribute much more to the proposed model on the Ciao dataset. We will see the quantitative results later (Section 7.4.4) that are consistent with this observation.

7.1.2 Evaluation Protocol and Metric. There are three kinds of parameters to be set. For optimization-related parameters, we use mini-batch gradient descent method with momentum to optimize the objective functions. Reference to empirical rules (e.g., Mnih and Salakhutdinov (2007)), we set momentum = 0.8, batchSize = #training / #numOfBatches, and learning rate = 0.0007; we randomly shuffle the training data prior to each epoch. For regularization-related parameters, we use norm regularization to avoid over-fitting. According to related bibliographies (e.g., Guo et al. (2015)), we set norm penalty $\lambda = 0.5$. For models-related parameters, these are F that is the number of latent factors, λ_{rev} which controls the contribution from reviews, and λ_{rel} that controls the contribution from social relations; see Section 7.5 in detail.

We randomly select $x\%$ as the training set and report the prediction performance on the remaining $1-x\%$ testing set, where x usually varies in $\{20, 50, 80, 90\}\%$. The reported results are the average values over five times independent random selection, being similar to 5-fold cross validation. All comparing methods abide by the same evaluation protocol.

The metric *root-mean-square error* (RMSE) for rating prediction task is defined as

$$RMSE_{\mathcal{T}} = \sqrt{\sum_{(u,i) \in \mathcal{T}} (R_{u,i} - \hat{R}_{u,i})^2 / |\mathcal{T}|}, \quad (23)$$

where \mathcal{T} is the test set. Compared with the metric *mean absolute error* (MAE) (Herlocker et al. 1999), RMSE puts more emphasis on large deviation than MAE (Herlocker et al. 2004). For instance, an error deviation of 2 points increases the total sum of error by 4 under the metric RMSE, while by 2 under the metric MAE. A smaller RMSE or MAE means a better prediction performance. A small improvement regarding RMSE could have a significant impact on the quality of recommendation and 10% RMSE improvement led to \$1M Grand Prize (Bennett and Lanning 2007; Koren 2008).

There are some other metrics used for evaluation in recommendation systems. For example, in real-world systems we may care about a few top-items for each user and hence we use the metric recall@ K to evaluate the performance of recommending the top K items to the target user. In this article, We think RMSE is a better metric for our task of rating prediction. In rating prediction, we not only care about the positive preference of ratings above 4, which means the users like the item so much, but also care about the negative preference of ratings under 1, which means that the users do not like the item so much. However, recall@ K only care about the positive preference

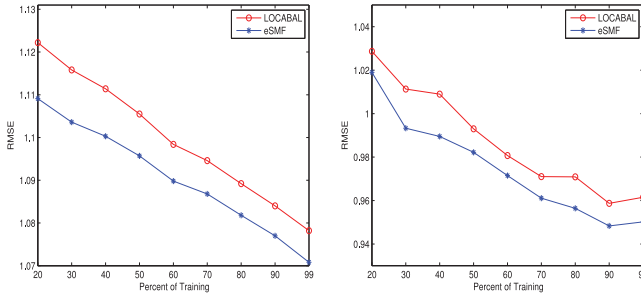


Fig. 5. Comparisons of eSMF with a Social MF method on two datasets. Left: Epinions; Right: Ciao. The figures are copied from Hu et al. (2015).

for real recommendation. We focus on the RMSE results and also report the recall results briefly in Section 7.6.

For each user’s ratings, we randomly reserve one into the validation set, and randomly select 1 or 5 into the training set, and put the rest into the test set. The former setting is called sparse setting and the latter is dense. We report the test result where the corresponding validation result is the best.

Note that, for the MR3++ model extended from the proposed model MR3, the predicted ratings $\hat{R}_{u,i}$ (see Equation (2)) should be replaced by $\hat{R}_{u,i}^*$ accordingly (see Equation (12)).

7.2 Comparing Social MF Methods

In this section, we first compare our proposed eSMF model with LOCABAL Tang et al. (2013b), a state-of-the-art Social MF method, to show the benefit of exploiting the graph structure of neighbors. The motivation for the comparison is two-fold: (1) to demonstrate that exploiting ratings and social relations more tightly can further improve the performance of social RSs; (2) to form an effective component of the proposed model MR3, which we evaluate in the following section (Section 7.3.1).

We use grid search to determine λ_{rel} that controls the contribution from social relations and report the best RMSE on the testing set over 50 passes. The eventually reported results are the average values over five times independent random selection of training set, being similar to five-fold cross validation. For both LOCABAL and eSMF, we get the best RMSE when $\lambda_{rel} = 0.1$. Parameters $\Theta = \{P, Q, H\}$ are randomly initialized from the normal distribution $\mathcal{N}(0, 0.01)$.

The results are demonstrated in Figure 5, with varying percentage of the training set = {20, 30, 40, 50, 60, 70, 80, 90, 99} and we have the following observation:

- Exploiting ratings and social relations tightly (i.e., exploiting the graph structure of neighbors via trust values capturing social influence locality) can further improve recommender performance in terms of RMSE on both datasets. For example, eSMF obtains 1.18%, 0.89%, and 0.72% relative improvement compared with a state-of-the-art Social MF method LOCABAL on Epinions with 20%, 50%, and 70% as the training set respectively under the RMSE metric.

7.3 Comparing Different Recommender Systems

In this section, we first demonstrate the effectiveness of the proposed model MR3 with different recommendation approaches to show the benefit of modeling three types of data sources simultaneously. And then we demonstrate the improvement of its extension MR3++ to show the benefit of incorporating implicit feedback from ratings.

Table 3. RMSE Comparisons of the Proposed Model MR3 with Different Methods ($F = 10$)

Datasets	Training (%)	Methods					Improvement of MR3 vs.		
		Mean	PMF	HFT	LOCABAL	MR3	PMF (%)	HFT (%)	LOCABAL (%)
Epinions	20	1.2265	1.2001	1.1857	1.1222	1.1051	8.60	7.29	1.55
	50	1.2239	1.1604	1.1323	1.1055	1.0809	7.35	4.76	2.28
	80	1.2225	1.1502	1.0960	1.0892	1.0648	8.02	2.93	2.29
	90	1.2187	1.1484	1.0867	1.0840	1.0634	7.99	2.19	1.94
Ciao	20	1.1095	1.0877	1.0439	1.0287	1.0142	7.25	2.93	1.43
	50	1.0964	1.0536	1.0379	0.9930	0.9740	8.17	6.56	1.95
	80	1.0899	1.0418	0.9958	0.9709	0.9521	9.42	4.59	1.97
	90	1.0841	1.0391	0.9644	0.9587	0.9451	9.95	2.04	1.44
Average							8.34	4.16	1.86

This table is copied from Hu et al. (2015).

7.3.1 Comparing the Proposed Model or MR3 with Different Recommender Systems. We first compare the proposed model MR3 introduced in Section 4.2 (see Equation (9) or Figure 3) with the following different types of recommendation approaches to show the benefit of modeling three data sources simultaneously:

Mean. This method predicts the rating always using the average, i.e., μ in Equation (2), across all training ratings. This is the best constant predictor in terms of RMSE.

PMF. This method performs MF on rating matrix as shown in Equation (1) (Mnih and Salakhutdinov 2007). It is the representative of latent factors CF (see Figure 2(a)). It only uses the rating source.

LOCABAL. This method is based on MF and exploits local and global social context as shown in Equation (7) (Tang et al. 2013b). It is the representative of Social MF (see Figure 2(b))⁷. It only uses ratings and relations.

HFT. This method combines latent factors in ratings with hidden topics in reviews as shown in Equation (8) (McAuley and Leskovec 2013). It is the representative of Topic MF (see Figure 2(c)). It only uses ratings and reviews.

HFT+LOCABAL. This is the simple hybrid method of HFT and LOCABAL, which linear weights their results: $\hat{R}_{u,i} = \alpha \hat{R}_{u,i}^{HFT} + (1 - \alpha) \hat{R}_{u,i}^{LOCABAL}$, where $\alpha \in (0, 1)$.

We use the source code PMF⁸ and HFT⁹ provided by their authors. For the hyperparameter λ_{rev} that controls the contribution from item reviews, it is determined by grid search. For HFT, $\lambda_{rev} = 0.1$; and for MR3, $\lambda_{rel} = 0.001$ and $\lambda_{rev} = 0.05$. More details about the sensitivity to meta parameters of MR3 will be discussed later (Section 7.5.1).

In our experiments, we leave out the comparison with HFT+LOCABAL. Because the core idea of MR3 is the alignment between latent factors found by LOCABAL and hidden topics found by HFT to form a unified model. Theoretically, MR3 is more elegant than linear combination which is out of our motivations. And practically, we do not intent to show the superior of our MR3 over the linear combination. We just show the unified model MR3 is better than its individual components.

The results of the comparison are summarized in Table 3, with varying percentage of the training set = {20, 50, 80, 90} and we have the following observations:

⁷Since we want to compare the performance of MR3 with the state of the art recommenders, so we compare it with LOCABAL and not with our proposed eSMF in Table 3. Instead, the comparison between eSMF and MR3 is conducted in Section 7.4.1.

⁸<http://www.cs.toronto.edu/~rsalakhu/>.

⁹<http://cseweb.ucsd.edu/~jmcauley/>.

- Exploiting social relations and reviews beyond ratings can both significantly improve recommender performance in terms of RMSE on the two datasets. For example, HFT and LO-CABAL obtain 4.95% and 5.60% relative improvement compared with PMF on Epinions, respectively, with 80% as the training set.
- Our proposed model MR3 always achieves the best results. Compared with HFT and LO-CABAL, MR3 averagely gains 0.0466 and 0.0217 absolute RMSE improvement on Epinions and 0.0392 and 0.0165 on Ciao, respectively. The main reason is that MR3 jointly models all three types of information effectively. The contribution from each component of data source to MR3 is discussed in the later subsection (Section 7.4.1).

7.3.2 Comparing the Extension Model or MR3++ with Different Recommender Systems. We then compare the extension of the proposed model MR3++ introduced in Section 5.2 (see Equation (13)) to show the benefit of incorporating implicit feedback from ratings.

PMF. This method is investigated in the above subsection (Section 7.3.1) and we list it here to see the impact of implicit feedback from ratings by comparing it with the following SVD++ method.

SVD++. This method exploits implicit feedback from ratings besides performing MF on rating matrix as shown in Equation (12) (Koren 2008). We can know the impact of implicit feedback from ratings by comparing it with the PMF method.

MR3. This method is investigated in the above subsection (Section 7.3.1) and we list it here to see the performance of its extension more clearly. It does not mine the data source more deeply, i.e., without incorporating implicit feedback from ratings.

MR3++. This method extends the proposed model MR3 by incorporating implicit feedback from ratings.

We adopt the source code of SVD++ provided in the LibRec.net Java RS Library. For the hyperparameter λ_{rev} that controls the contribution from item reviews and λ_{rel} that controls the contribution from social relations, they are determined by grid search. For MR3++, $\lambda_{\text{rel}} = 0.001$ and $\lambda_{\text{rev}} = 0.005$. More details about the sensitivity to meta parameters of MR3++ will be discussed later.

The results¹⁰ of the comparison are summarized in Table 4 and we have the following observations:

- Integrating three data sources and exploiting implicit feedback can both improve the accuracy of rating prediction in terms of RMSE on two datasets. For example, SVD++ and MR3 obtain 7.94% and 8.02% relative improvement compared with PMF on Epinions with 80% as the training set, respectively.
- The extension model MR3++ almost achieves the slightly better results. Compared with SVD++ and MR3, MR3++ averagely gains 5.94% and 0.31% relative RMSE improvement on Ciao. The main reason is that MR3++ can jointly models all three types of information and meanwhile can mine the rating source deeply. The contributions from the two components of MR3++ is discussed in the following subsection.

7.4 Contribution of Data Sources and Impact of Implicit Feedback

In this section, we first measure the contribution from item reviews and social relations in the proposed model MR3, and then measure the impact of implicit feedback from ratings in the extension MR3++ model.

¹⁰The standard deviations are all less than 10^{-5} . Although, the results in Table 3 and Table 4 can be merged, we split the whole results into the present two tables to show the contributions of auxiliary sources (i.e., the MR3 model) and the impact of implicit feedback (i.e., the MR3++ model) more clearly.

Table 4. RMSE Comparisons of the Extended Model MR3++ with Different Methods ($F = 10$)

Datasets	Training (%)	Methods			
		PMF	SVD++	MR3	MR3++
Epinions	20	1.2001	1.1159	1.1051	1.1026
	50	1.1604	1.0816	1.0809	1.0785
	80	1.1502	1.0655	1.0648	1.0641
	90	1.1484	1.0601*	1.0634	1.0618
Ciao	20	1.0877	1.0555	1.0142	1.0132
	50	1.0536	1.0276	0.9740	0.9711
	80	1.0418	1.0139	0.9521	0.9464
	90	1.0391	1.0055	0.9451	0.9425

The columns of PMF and MR3 are copied from Table 3 to show the contributions of auxiliary sources and the impact of implicit feedback more clearly. Refer to Section 7.4.4 for the explanation of the star entry.

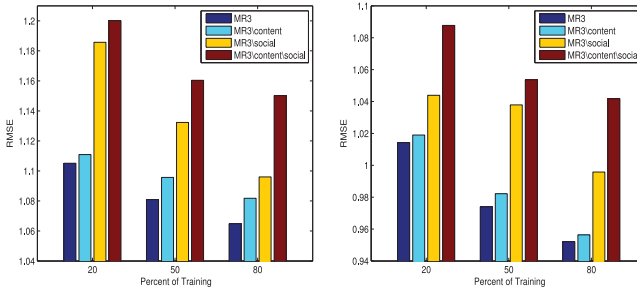


Fig. 6. Predictive performance of MR3 compared with its three components. Left: Epinions; Right: Ciao. The figures are copied from Hu et al. (2015).

7.4.1 Contribution of Data Sources from Reviews and Social Relations. We have shown the effectiveness of integrating ratings with social relations and reviews in our proposed model MR3. We now investigate the contribution of each data source to the proposed model by eliminating the contribution of social relations and reviews from MR3 respectively:

MR3\content: Eliminating the impact of reviews by setting $\lambda_{\text{rev}} = 0$ in Equation (9), which is equivalent to eSMF as shown in Equation (11).

MR3\social: Eliminating the impact of social relations by setting $\lambda_{\text{rel}} = 0$ in Equation (9), which is equivalent to HFT as shown in Equation (8).

MR3\content\social: Eliminating the impact of both reviews and social relations by setting $\lambda_{\text{rev}} = 0$ and $\lambda_{\text{rel}} = 0$ in Equation (9), which is equivalent to PMF as shown in Equation (1).

The predictive results of MR3 and its three components on Epinions dataset are shown in Figure 6. The performance degrades when either social relations or reviews are eliminated. In detail, *MR3\content*, *MR3\social*, and *MR3\content\social* averagely reduce 1.19%, 4.29%, and 7.99% relative RMSE performance on Epinions, respectively, suggesting that both reviews and social relations contain essential information for recommender.

7.4.2 Further Analysis of Contribution from Auxiliary Sources. We have shown that $MR3 \setminus content \setminus social$ degrades 7.99% performance of total relative RMSE; and intuitively we want to know how the additional sources of information help MR3 improve recommendation. We address this issue from the richness perspective of social relations and reviews via contrastive analysis. In detail, we first collect the $(user, item)$ pairs such that the PMF method gets worst prediction, while the MR3 method gets the best; we set the difference threshold between PMF and MR3 to 1. For example, given that the prediction error of PMF on (u, i) is $e_1 = |\hat{R}_{u,i}^{PMF} - R_{u,i}|$ and MR3 is $e_2 = |\hat{R}_{u,i}^{MR3} - R_{u,i}|$, if $(e_1 - e_2) \geq 1$ then we reserve this pair. We then calculate the average social relations among these users and the average review words among these items to measure the quality and richness of the additional information.

With 80% training on the Epinions dataset, we collect total 4,382 such pairs where the number of users is 3,627 and the number of items is 2,875; they represent 2.80%, 7.33%, and 3.88% on total ratings, total users, and total items, respectively. The average social relations among these users are 14.18 and the average review words among these items are 1129.23. Note that these two measures are much larger than the corresponding total average values which are 8.78 and 30.3, respectively (see Table 2). These results intuitively confirm the significant contributions from additional data sources to some extent.

7.4.3 Impact of Implicit Feedback from Ratings. We have investigated the contribution of each data source to the proposed model by eliminating the impact of social relations and reviews from MR3, respectively, showing that extra data sources (item reviews and social relations) are both useful for improving the recommendation performance in terms of RMSE metric. We now investigate the impact of implicit feedback from ratings in the extension model $MR3++$. The $MR3++$ model contains two kinds of components: one is to integrate three data sources (ratings, social relations and item reviews) and the other is to mine a single data source deeply (incorporating implicit feedback from ratings). In detail, we investigate the performance of $MR3++$ by eliminating the impact of data sources and implicit feedback from it in turn:

$MR3++ \setminus sources$: Eliminating the impact of more data sources (i.e., remove social relations and item reviews) by setting $\lambda_{rev} = 0$ and $\lambda_{rel} = 0$ in Equation (13), which is equivalent to the SVD++ model as shown in Equation (12).

$MR3++ \setminus implicit$: Eliminating the impact of implicit feedback from ratings by setting $Y_j = 0$ in Equation (13), which is equivalent to the MR3 model as shown in Equation (9).

$MR3++ \setminus sources \setminus implicit$: Eliminating the impact of both more data sources and implicit feedback by setting $\lambda_{rev} = 0$, $\lambda_{rel} = 0$ and $Y_j = 0$ in Equation (13), which is equivalent to the PMF model as shown in Equation (1).

The predictive results of $MR3++$ and its two components on Epinions and Ciao datasets are shown in Figure 7. The histograms show that the performance degrades somewhat when implicit feedback are eliminated on both datasets. For example, $MR3++ \setminus implicit$ reduces 0.23% and 0.60% relative RMSE performance on Epinions and Ciao datasets with 20% and 80% as the training set, respectively.

7.4.4 Integrating More Data Sources vs. Mining Limited Data Deeply. Revisit Table 4 and Figure 7; we can see clearly that integrating more data sources can improve a bigger margin than only mining a single data source deeply in some cases (here, on the Ciao dataset, in terms of average relative RMSE, the margin greater than 5.61%), but not always (here, on the Epinions dataset, in terms of average relative RMSE, the margin less than 0.23%; and when the training percent is 90, i.e., the star entry in Table 4, mining the rating source deeply even outperforms integrating social relation and review data sources).

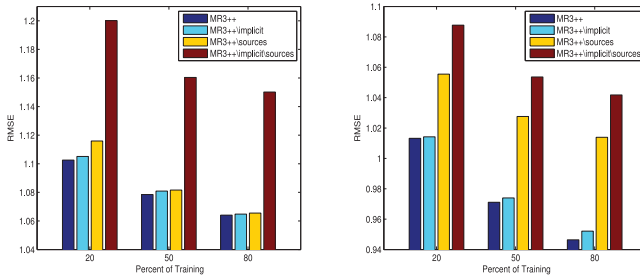


Fig. 7. Predictive performance of MR3++ compared with its two components. Left: Epinions. Right: Ciao. The figures are copied from Hu et al. (2015).

Consider the quality of social relations and reviews. As we mentioned previously in Section 7.1.1 (see Table 2), the number of average words per item on Ciao is 42 times longer than that on Epinions, and the social density on Ciao is 12 times denser than that on Epinions. So Ciao contains richer and higher quality information in social relations and reviews. Intuitively and empirically, more data sources may improve performance better than mining single data source deeply when extra data sources contain richer information.

The results on the Epinions dataset in Figure 7 are worth further thinking. The removing of either implicit feedback component or auxiliary sources component has little effect on the rating prediction performance, but the removing of both components has big impact. This observation is different from the results in Figure 6 and even from the results on the Ciao dataset in Figure 7. One possible explanation for these findings is that the contributions from the two component are not linear-additive. Different datasets show diverse effect between implicit feedback and auxiliary sources.

7.5 Sensitivity to Meta Parameters: F , λ_{rel} , and λ_{rev}

We analyze the sensitivity of the proposed model and its extension model to the three important hyperparameters: one controls the contribution from social relations, one controls the contribution from item reviews, and another determines the dimensionality of latent representations.

7.5.1 Sensitivity Analysis of the Proposed Model or MR3. The model MR3 has three important parameters: (1) the number of latent factors F ; (2) the hyperparameter λ_{rev} that controls the contribution from reviews; and (3) the hyperparameter λ_{rel} that controls the contribution from social relations. We investigate the sensitivity of MR3 to these parameters by varying one of them while fixing the other two.

First, we fix $F = 10$ and study how the reviews associated hyperparameter λ_{rev} and the social relations associated one λ_{rel} affect the whole performance of MR3. As shown in Figure 8, we have some observations: (1) the prediction performance degrades when either $\lambda_{rel} = 0$ or $\lambda_{rev} = 0$; (RMSE is 1.1502 when both are zero.) (2) MR3 is relatively stable and not sensitive to λ_{rel} and λ_{rev} when they are small (e.g., from 0.0001 to 0.1), so we choose the reasonable values 0.001 and 0.05 for them, respectively.

Next, we fix $\lambda_{rel} = 0.001$ and $\lambda_{rev} = 0.05$, and vary the number of latent factors $F = \{5, 10, 15, 20, 30, 50, 70, 100\}$ with 20%, 50%, 80% as the training set, respectively. As shown in Figure 9, MR3 is relatively stable and not sensitive to F , so we choose the reasonable value 10 as default.

7.5.2 Sensitivity Analysis of the Extension Model or MR3++. We repeat the above process to investigate the sensitivity of MR3++ to these three meta parameters.

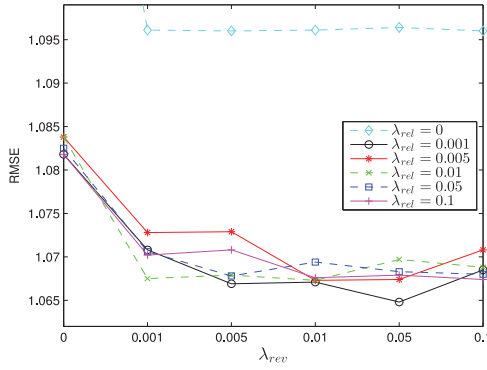


Fig. 8. Predictive performance of MR3 by varying λ_{rel} and λ_{rev} . Both vary in $\{0, 0.001, 0.005, 0.01, 0.05, 0.1\}$. RMSE is 1.1502 when both are zero. Fixing $F = 10$. Percent of training set = 80. Dataset: Epinions. The figure is copied from Hu et al. (2015).

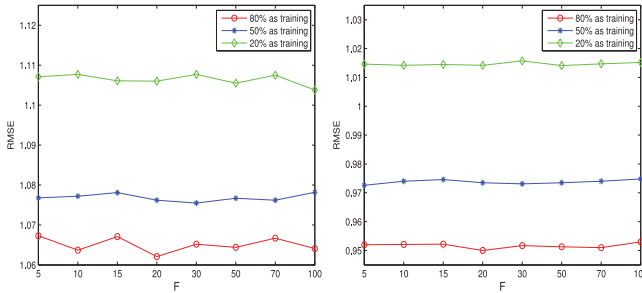


Fig. 9. Predictive performance of MR3 by varying the number of latent factors F . Fixing $\lambda_{rel} = 0.001$ and $\lambda_{rev} = 0.05$. Left: Epinions; Right: Ciao. The figures are copied from Hu et al. (2015).

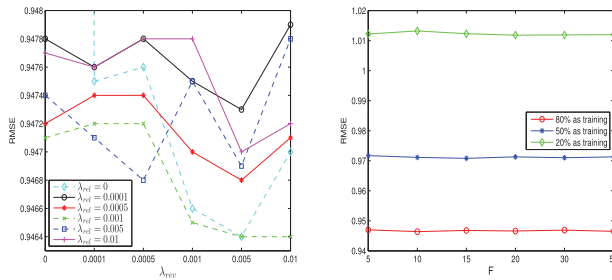


Fig. 10. Predictive performance of MR3++ by varying hyperparameters. Left: varying λ_{rel} and λ_{rev} by fixing $F = 10$; RMSE is 1.0139 when both are zero; Percent of Training is 80%. Right: varying F by fixing $\lambda_{rel} = 0.001$ and $\lambda_{rev} = 0.005$. Dataset: Ciao.

First, we fix $F = 10$ and study how the reviews associated parameter λ_{rev} and the social relations associated one λ_{rel} affect the whole performance of MR3++. As shown in Figure 10 (on the right), we have some observations: (1) the prediction performance degrades when either $\lambda_{rel} = 0$ or $\lambda_{rev} = 0$; (RMSE is 1.0139 when both are zero.) (2) MR3++ is relatively stable and not sensitive to λ_{rel} and λ_{rev} when they are small (e.g., from 0.0001 to 0.01), so we choose the reasonable values 0.001 and 0.005 for them respectively. Note that the length of RMSE range interval is within 0.002.

Table 5. Recall Comparisons of Model MR3/MR3++ with Different Methods

Datasets	Ratings	Methods			
		HFT (%)	LOCABAL (%)	MR3 (%)	MR3++ (%)
Epinions	5	74.7127	73.9329	74.6411	74.6455
	1	59.8015	59.2960	60.1507	60.1232
Ciao	5	73.4361	73.9042	73.8051	74.0804
	1	52.3657	53.1330	53.1330	53.6445

Next, we fix $\lambda_{\text{rel}} = 0.001$ and $\lambda_{\text{rev}} = 0.005$, and vary the number of latent factors $F = \{5, 10, 15, 20, 30, 50\}$ with 20%, 50%, 80% as the training set, respectively. As shown in Figure 10 (on the left), MR3++ is relatively stable and not sensitive to F , so we choose the reasonable value 10 as default.

7.6 Comparing Different Methods with Recall Metric

In this article, we choose RMSE as the main metric for evaluation because we focus on the rating prediction. However, recall is a more practical metric for real top- N RS. Recall cares more about the positive preferences.

Inspired by top- N recommendations (Kabbur et al. 2013), we define the set of all 5 ratings of a specific user u as positive instances, denoted as G_u for the ground truth and $|G_u|$ is the number of his exact 5 ratings. For each user, we select top variant $K_u = |G_u|$ predicted ratings, denoted as G'_u for evaluation. Thus, the recall, or average hit rate metric is defined as

$$\text{recall} = \frac{\sum_u |G_u \cap G'_u|}{\sum_u |G_u|}. \quad (24)$$

Hit rate demonstrates the performance of recalling the test items in size $|G_u|$ recommendations list of user u . Inversely, if we take the ratings of 1 as negative instances, we can see the performance of predicting dislikes of users.

The results of the comparison using recall metric is summarized in Table 5. The setting of meta parameters is the same with Section 7.3.1 and Section 7.3.2. MR3++ achieves a little improvement over HFT and LOCABAL on two datasets, hitting on either rating 5 or rating 1. Overall, MR3 and MR3++ models got the competitive results. We think it is reasonable that our proposed methods did not achieve the significant improvement with the recall metric. HFT, LOCABAL, and our methods focus on getting closer predicted ratings, rather than the best ranking of items. In other words, we define the problem as a prediction regression task, leading us naturally to the standard RMSE metric. The recall metric cares more about the exact match of top- N recommendations where the models are investigated to pick out favorite things of users. These models always put more weight on higher ratings in training set, aiming at recalling the top preferred items. MR3 and MR3++, including HFT, are not designed for this. The emphasis is closer and precise predictions all over 1–5 ratings instead.

7.7 Running Time Analysis

Practically, the running time of models is a great concern in real-world RS. We conduct several rounds of experiments with the same settings, that is, the number of latent factors $F = 10$, training percent = 80% and testing the remaining 20%. We set learning rate = 0.0001 for all models and

Table 6. Empirical Running Time (seconds) Comparison of the Proposed MR3/MR3++ with Different Methods

Datasets	Steps	Methods			
		HFT	LOCABAL	MR3	MR3++
Epinions	Average Epoch	1.4168	1.1251	1.549368	18.714360
	Total (best valid)	77.8941	1746.1273	212.9282	2298.6840
	Predicting	0.2483	0.2500	0.274319	2.117205
Ciao	Average Epoch	0.3490	0.2656	0.416025	1.622635
	Total (best valid)	431.0403	152.2076	601.7270	913.3129
	Predicting	0.0469	0.0531	0.062847	0.098469

We measure the time cost on a PC machine with Intel(R) Core(TM) i5-4590 CPU @ 3.30GHz, 8G RAM, GCC 5.3.0 installed. Time cost of SVD++ is not listed for we adopt the source code from LibRec.net Java Recommender System Library, and the implementation is quite different from HFT, LOCABAL, MR3 and MR3++.

$nEpoch = 5$ in Algorithm 1, meaning we sample and update $z_{d,n}$ every 5 epoches of updating Θ, Φ, κ , which is not required in LOCABAL model.

We compare the training time and the predicting time, respectively, as shown in Table 6. We would like to explain the comparing results as follows:

- As for the time of one training epoch, MR3 is close to HFT and LOCABAL, while MR3 combines more sources of information. MR3++ will include many large-scale implicit feedbacks, so it takes much longer to train through one epoch, especially on the dataset of Epinions which has more users and complicated social networks. We also compare the total time cost for training with different methods. We adopt the early stop on the best valid result to avoid overfitting and we count that time as total. The total time cost of MR3++ is about ten times of that of MR3, because of the around ten times cost for each epoch. The total time of LOCABAL is extremely long as it converges after many more iterations than other methods especially on the larger Epinions dataset.
- As for the predicting time, it is almost the same comparing MR3 with HFT and LOCABAL. MR3++ costs a little more for predicting because of the extra implicit feedbacks included.

In a word, from Table 6, we can conclude that the empirical running time of our proposed models is acceptable compared to other methods.

Figure 11 demonstrates the training process by iterations and RMSE on the valid set. In most cases, these methods become convergent within about 150 rounds of iterations while the valid RMSE of LOCABAL method can still decrease after that. More specifically, about 500 iterations are required on Ciao and even more on Epinions with the conservatively small learning rate. From Figure 11(a) and 11(c), MR3 and MR3++ only take few more iterations until convergence yet yield better performance. Interestingly, by Figure 11(b) and 11(d), MR3++ continues to converge on a lower valid RMSE value than MR3, for MR3++ takes many implicit feedbacks into account.

8 CONCLUSIONS AND FUTURE WORK

Heterogenous recommendation information sources beyond explicit ratings including social relations and item reviews present both opportunities and challenges for conventional RSs. We

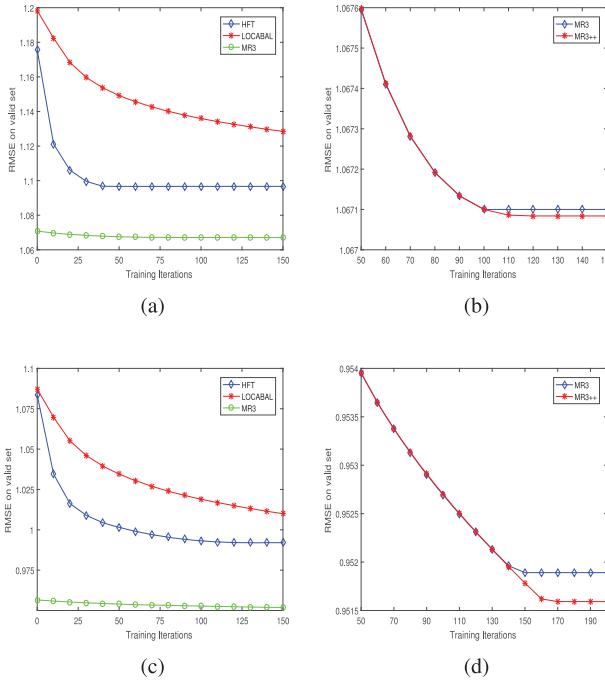


Fig. 11. Training process of all mentioned methods by iterations. Figure (a) and (b) on dataset Epinions, (c) and (d) on dataset Ciao. (a) and (c) compare HFT, LOCABAL and MR3 on two datasets while (b) and (d) compare MR3 with MR3++, respectively.

investigated how to fuse these three kinds of information tightly and effectively for recommendation by aligning the topic and social latent factors. Furthermore, we mine the limited data source more deeply by incorporating implicit feedback from ratings.

We first proposed a novel model MR3 which jointly models ratings, social relations, and item reviews by aligning latent factors and hidden topics to perform social MF and topic MF simultaneously for effective rating prediction. Moreover, an extended Social MF method eSMF was obtained by capturing the graph structure of neighbors via trust values to exploit the ratings and social relations more tightly. We then enhanced the proposed model by incorporating implicit feedback from ratings, resulting in the extension model MR3++, to demonstrate its capability and flexibility. The core idea of mining ratings deeply is to learn an extra implicit feature matrix to consider the influence of rated items.

Empirical results on two real-world datasets demonstrated that our proposed models lead to improved predictive performance compared with various different kinds of recommendation approaches. Furthermore, we designed experiments to understand the contribution of each data source and the impact of the implicit feedback from ratings; we also compare the contribution of integrating more data sources with the impact of mining the limited data source deeply from the perspective of the richness of auxiliary information. We finally analyzed the sensitive analysis of our models to the three meta parameters.

We focused on the rating prediction task in this article and hence we evaluated the performance of RSs under the metric RMSE. However, more metrics should be explored in the future work (McNee et al. 2006).

REFERENCES

- Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17, 6 (2005), 734–749.
- Y. Bao, H. Fang, and J. Zhang. 2014a. Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI'14)*. 350.
- Yang Bao, Hui Fang, and Jie Zhang. 2014b. TopicMF: Simultaneously exploiting ratings and reviews for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 2–8.
- James Bennett and Stan Lanning. 2007. The netflix prize. In *Proceedings of the KDD Cup and Workshop*. 35.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research* 3 (2003), 993–1022.
- John S. Breese, David Heckerman, and Carl Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 43–52.
- Allison J. B. Chaney, David M. Blei, and Tina Eliassi-Rad. 2015. A probabilistic model for using social networks in personalized item recommendation. In *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 43–50.
- Chaochao Chen, Xiaolin Zheng, Yan Wang, Fuxing Hong, and Zhen Lin. 2014. Context-aware collaborative topic regression with social matrix factorization for recommender systems. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*.
- Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J. Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 193–202.
- Xuetao Ding, Xiaoming Jin, Yujia Li, and Lianghao Li. 2013. Celebrity recommendation with collaborative social topic regression. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'13)*.
- Michael D. Ekstrand, John T. Riedl, and Joseph A. Konstan. 2011. Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction* 4, 2 (2011), 81–173.
- Yi Fang and Luo Si. 2011. Matrix co-factorization for recommendation with rich side information and implicit feedback. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. ACM, 65–69.
- Gayatree Ganu, Noemie Elhadad, and Amélie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *WebDB*, Vol. 9. Citeseer, 1–6.
- Prem K. Gopalan, Laurent Charlin, and David Blei. 2014. Content-based recommendations with poisson factorization. In *Proceedings of the Advances in Neural Information Processing Systems*. 3176–3184.
- Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences* 101, suppl 1 (2004), 5228–5235.
- Quanquan Gu, Jie Zhou, and Chris H. Q. Ding. 2010. Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In *Proceedings of the SIAM International Conference on Data Mining*. SIAM, 199–210.
- Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2015. TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 123–129.
- Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. 1999. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 230–237.
- Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* 22, 1 (2004), 5–53.
- Thomas Hofmann. 2004. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems* 22, 1 (2004), 89–115.
- Guang-Neng Hu, Xin-Yu Dai, Yunya Song, Shu-Jian Huang, and Jia-Jun Chen. 2015. A synthetic approach for recommendation: Combining ratings, social relations, and reviews. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*.
- Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM'08)*. IEEE, 263–272.
- Niklas Jakob, Stefan Hagen Weber, Mark Christoph Müller, and Iryna Gurevych. 2009. Beyond the stars: Exploiting free-text user reviews to improve the accuracy of movie recommendations. In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*. ACM, 57–64.
- Mohsen Jamali and Martin Ester. 2011. A transitivity aware matrix factorization model for recommendation in social networks. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*.
- Santosh Kabbur, Xia Ning, and George Karypis. 2013. Fism: Factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 659–667.

- Yehuda Koren. 2008. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 426–434.
- Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.
- Yehuda Koren and Joe Sill. 2011. OrdRec: An ordinal model for predicting personalized item rating distributions. In *Proceedings of the 5th ACM Conference on Recommender Systems*. ACM, 117–124.
- Lei Li, Wei Peng, Saurabh Kataria, Tong Sun, and Tao Li. 2015. Recommending users and communities in social media. *ACM Transactions on Knowledge Discovery from Data* 10, 2 (2015), 17.
- Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE* 7, 1 (2003), 76–80.
- Guang Ling, Michael R. Lyu, and Irwin King. 2014. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender Systems*. ACM, 105–112.
- Nathan N. Liu, Evan W. Xiang, Min Zhao, and Qiang Yang. 2010. Unifying explicit and implicit feedback for collaborative filtering. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. ACM, 1445–1448.
- Hao Ma, Irwin King, and Michael R. Lyu. 2009. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 203–210.
- Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King. 2008. Sorec: Social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. ACM, 931–940.
- Hao Ma, Dengyong Zhou, Chao Liu, Michael R. Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*. ACM, 287–296.
- Paolo Massa and Paolo Avesani. 2007. Trust-aware recommender systems. In *Proceedings of the 2007 ACM Conference on Recommender Systems*. ACM, 17–24.
- Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*. ACM, 165–172.
- Sean M. McNee, John Riedl, and Joseph A. Konstan. 2006. Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *Proceedings of the CHI'06 Extended Abstracts on Human Factors in Computing Systems*. ACM, 1097–1101.
- Andriy Mnih and Ruslan Salakhutdinov. 2007. Probabilistic matrix factorization. In *Proceedings of the Advances in Neural Information Processing Systems*. 1257–1264.
- Rong Pan, Yunhong Zhou, Bin Cao, Nathan N. Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM'08)*. IEEE, 502–511.
- Arkadiusz Paterek. 2007. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of the KDD Cup and Workshop*, vol. 2007. 5–8.
- Sanjay Purushotham, Yan Liu, and C.-C. Jay Kuo. 2012. Collaborative topic regression with social matrix factorization for recommendation systems. In *Proceedings of the 29th International Conference on Machine Learning (ICML'12)*. 759–766.
- Steffen Rendle. 2010. Factorization machines. In *Proceedings of the 10th International Conference on Data Mining (ICDM'10)*. IEEE, 995–1000.
- Jasson D. M. Rennie and Nathan Srebro. 2005. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning*. ACM, 713–719.
- Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*. ACM, 285–295.
- Ajit P. Singh and Geoffrey J. Gordon. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 650–658.
- Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu. 2013a. Exploiting homophily effect for trust prediction. In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*. ACM, 53–62.
- Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. 2013b. Exploiting local and global social context for recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence*. 264–269.
- Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 448–456.
- Hao Wang, Binyi Chen, and Wu-Jun Li. 2013. Collaborative topic regression with social regularization for tag recommendation. In *Proceedings of the International Joint Conference on Artificial Intelligence*.
- Jun Wang, Arjen P. De Vries, and Marcel J. T. Reinders. 2006. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 501–508.
- Yinqing Xu, Wai Lam, and Tianyi Lin. 2014. Collaborative filtering incorporating review text and co-clusters of hidden user communities and item groups. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 251–260.

- Bo Yang, Yu Lei, Dayou Liu, and Jiming Liu. 2013. Social collaborative filtering by trust. (2013), 2747–2753.
- Jing Zhang, Biao Liu, Jie Tang, Ting Chen, and Juanzi Li. 2013. Social influence locality for modeling retweeting behaviors. In *Proceedings of the International Joint Conference on Artificial Intelligence*, vol. 13. 2761–2767.
- Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. 2007. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 487–494.

Received May 2016; revised June 2017; accepted July 2017