Learning and Transferring Social and Item Visibilities for Personalized Recommendation

Lin Xiao
Tsinghua University
Institute of Interdisciplinary Information Sciences,
jackielinxiao@gmail.com

Zhang Min*
Tsinghua University
Department of Computer Science and Technology,
z-m@tsinghua.edu.cn

Zhang Yongfeng
University of Massachusetts Amherst
College of Information and Computer Science,
yongfeng@cs.umass.edu

Liu Yiqun
Tsinghua University
Department of Computer Science and Technology,
yiqunliu@tsinghua.edu.cn

Ma Shaoping
Tsinghua University
Department of Computer Science and Technology,
msp@tsinghua.edu.cn

ABSTRACT

User feedback in the form of movie-watching history, item ratings, or product consumption is very helpful in training recommender systems. However, relatively few interactions between items and users can be observed. Instances of missing user–item entries are caused by the user not seeing the item (although the actual preference to the item could still be positive) or the user seeing the item but not liking it. Separating these two cases enables missing interactions to be modeled with finer granularity, and thus reflects user preferences more accurately. However, most previous studies on the modeling of missing instances have not fully considered the case where the user has not seen the item. Social connections are known to be helpful for modeling users' potential preferences more extensively, although a similar visibility problem exists in accurately identifying social relationships. That is, when two users are unaware of each other's existence, they have no opportunity to connect. In this paper, we propose a novel user preference model for recommender systems that considers the visibility of both items and social relationships. Furthermore, the two kinds of information are coordinated in a unified model inspired by the idea of transfer learning. Extensive experiments have been conducted on three real-world datasets in comparison with five state-of-the-art approaches. The encouraging performance of the proposed system verifies the effectiveness of social knowledge transfer and the modeling of both item and social visibilities.

CCS CONCEPTS

•**Information systems** → **Collaborative filtering; Social networks;** *Social networking sites;*

KEYWORDS

Recommender System, Implicit Feedback, Social Network

1 INTRODUCTION

Recommender systems attempt to provide items or information of interest to users based on their preferences. Generally, user preferences are learnt from historical records, such as ratings, reviews, clicks, and consumption. However, the interactions between users and items that can be observed are rather limited. These missing interactions are referred to as **missing feedback** in recommender systems.

Previous studies have dealt with this problem in two ways: treat all missing feedback as negative feedback [11, 12, 26] , or randomly sample negative feedback from the missing feedback [30] . As a result, all the missing feedback or the sampled missing feedback is considered to be negative feedback, and then recommendations are made based on this augmented information.

However, one important fact should not be overlooked when we revisit the problem of missing interactions—if an item was visible to a user but the user did not consume it, the missing interaction represents a negative preference for the item; nevertheless, if an item was never visible to the user, then it could not possibly have been consumed, and the missing interaction does not imply either a positive or negative preference. Most previous approaches have not separated these two cases, and are therefore biased in terms of the accurate modeling of user preferences.

As shown in previous studies on social-aware recommender systems, the social behavior of users and their interactions with items are positively correlated [21, 31, 35, 40]. However, social information is also sparse, and the social connections between users also face the problem of missing entries. Similarly, a connection between two users may be missing

[*] Corresponding Author

CIKM'17 , Singapore, Singapore
© 2017 ACM. 978-1-4503-4918-5/17/11. . . $15.00
DOI: 10.1145/3132847.3132911

because of a lack of affinity or invisibility (for example, user A does not know that user B is on the social network), and this latter case has not previously been studied.

Two random variables (**item visibility** and **social visibility**) are introduced in our work, indicating whether the items and other users are visible to a specific user. We argue that modeling both **item visibility** and **social visibility** will help to enrich the accurate modeling of user preferences. Although the enrichment is achieved from two different aspects, they can be jointly modeled in recommender systems. The idea of incorporating social information into recommender systems is not new [4, 42], but there are two important aspects that should be considered: first, the model needs to fully utilize the correlation between the social and rating behavior of users and capture this effect in user preference modeling; second, the social and rating behavior comes from different domains, and this heterogeneity needs to be captured in the model. Transfer learning [27] is a suitable choice for the coordination of social information and rating behavior. The key concept behind transfer learning is that knowledge is stored while solving one problem before being applied to a different but related problem. In our work, we learn about the social domain using a latent factor model and apply the learnt user social latent factors to model user preferences. We selectively transfer the latent factors to capture the correlation between two domains while retaining the heterogeneity.

The focus of our work is **to make recommendations to users**. However, there are insufficient user–item interactions to capture the user preferences accurately. Both social and rating domains reflect user preferences, and transferring common knowledge between the domains enables better user profiling. In this paper, we propose a probabilistic generative model for item recommendation that coordinates both user–item and user–user interactions based on transfer learning. We then model both item and social visibilities to handle implicit feedback in recommender systems. The advantage of transfer learning is that it compensates for the shortage of information in the target domain by transferring knowledge from the source domain. As social connections reflect the personalities of users, knowledge from the source (social) domain can be leveraged to enrich the preferences of users that are difficult to learn from the target (rating) domain. More specifically, we adopt a latent factor model to depict the generation of both user–item interactions and user–user social connections. The transfer of latent factors is used to correlate the two different domains. An Expectation-Maximization (EM) algorithm is designed to determine the parameter values. Extensive experiments are conducted on three real-world datasets in comparison with five state-of-the-art approaches. The results show that our method outperforms all of the state-of-the-art approaches.

To the best of our knowledge, only one previous study has considered the visibility of items [19], using the concept of exposure to model whether an item has been observed by a user. The differences between our model and this previous work are twofold: first, the exposure concept only reflects the interaction between users and items, whereas we use visibility to model user–item and user–user interactions simultaneously; second, we adopt a transfer learning strategy to coordinate social and rating information in a unified model, enabling the user preferences to be learned more effectively.

The remainder of this paper is organized as follows. The next section introduces some related work. Section 3 presents a detailed description of the modeling of both item and social visibility in recommender systems. We present the results of extensive experiments with real-world datasets in Section 4. In Section 5, we discuss transfer learning in our model through a comparison with another model. Finally, we conclude the paper in Section 6.

2 RELATED WORK

2.1 Recommendations with Implicit Feedback

User feedback is frequently observed in real-life scenarios in the form of ratings, reviews, clicks, and the consumption of items. Handling user feedback is a key issue in recommender systems, and considerable efforts have been made to deal with missing feedback. Two basic approaches are adopted in previous studies: marking all the missing feedback as negative instances, such as in Weighted Matrix Factorization (WMF) [12] and Sparse Linear Method (SLIM) [25], [37], or sampling from the missing feedback as negative instances, such as in Bayesian Personalized Ranking (BPR) [30].

However, neither approach gives a good interpretation for the selection of negative instances, and assigning artificial weights for all the missing feedback lacks the accuracy needed for modeling. In the proposed method, we model the missing feedback by introducing visibilities and utilize a Bayesian approach to estimate whether the missing feedback is the result of user preferences or invisibility.

Various negative feedback sampling approaches have been considered [26, 29, 30]. Although the sampling procedure differs, the negative feedback is typically selected from the missing entries. [10] and [1] are recent studies on implicit feedbacks in recommendation. In [10], the negative samples are generated from unobserved instances and a K-separable principle is proposed in [1] to allow efficient optimization of implicit feedback problems with Coordinate Descent.

As discussed in the Introduction, only one study has considered item visibility for missing interactions [19]. Our work differs from this in two aspects: first, we model both social and item visibilities in recommender systems, rather than only considering item visibility; second, we adopt transfer learning to coordinate the two kinds of interactions into a unified model.

2.2 Social Recommendations

Social information is known to be helpful in recommendation systems [9, 13, 22, 23, 36, 41]. Most studies assume that some social homophily effect exists, causing users to behave consistently with others in social connections [9, 22]. However, most previous studies on social-aware recommender systems focus on rating prediction tasks, which is claimed to be less efficient in item recommendation tasks [5, 33].

[14] extends the approach in [13] by combining random walks with collaborative filtering for item recommendation. The Multi-Relational Bayesian Personalized Ranking (MR-BPR) model [17], which combines multi-relational matrix factorization with the BPR framework, predicts both user feedback on items and on social relationships. In [2], a topic model is proposed to detect subtopics from microblogs by utilizing the correlations among different media types.

There are two state-of-the-art algorithms that utilize social information for item recommendation tasks [4, 42]. In [42], the authors assume that users are more likely to have seen items consumed by their friends, and use this effect to sample negative feedback in BPR. In contrast, [4] utilizes Poisson factorization to incorporate social information into a matrix factorization scheme. None of the aforementioned studies considers the modeling of missing feedback in social relationships. In the present study, we use some of the most popular social-aware recommendation algorithms as benchmarks. Unlike previous studies, we capture the visibility phenomenon in social relationships and coordinate it into a generative model with transfer learning procedures.

A recent study [38] considers the strength of social ties and its application in social recommendations. As this is an extension of BPR, the missing feedback is randomly selected as negative feedback. In [28], the Collaborative Ranking (CofiRank) model [39] is extended to include social connections. As CofiRank considers observations of both positive and negative feedback, the issue of missing feedback is not investigated. The difference between our work and that described in [38] and [28] lies in two aspects: first, we concentrate on the issue of handling missing feedback from the perspective of visibility, whereas these previous studies either sample the negative feedback from missing feedback [38] or concentrate on observed (both positive and negative) feedback [28]; second, we accommodate user–item interactions and social connections with transfer learning principles, whereas the previous studies extend existing ranking models (CofiRank [39] and BPR [30]) with social connections.

2.3 Transfer Learning

Transfer learning has been adopted in various systems for cross-domain data mining tasks [18, 27, 43, 45]. The key idea of transfer learning is that the knowledge from the source domain is transferred into the model of the target domain. As previously noted [44], the social media which contains multi-domain information, provides a bridge in transfer learning. [15, 32] utilize transfer learning to deal with multi-relational data representation in social networks, but they do not specifically focus on recommendation tasks. Code book transfer [27] and coordinate system transfer [18] are two important applications of transfer learning on recommender systems.

Selective transfer learning [6] has been employed for cross-domain recommendations [20]. As pointed out in [20], parts of the source domain data are inconsistent with the observations in the target domain, which may affect the construction of a model in the target domain. In this paper, we assume that the source (social) domain and target (rating) domain can be represented by latent factors. We propose a selective latent factor transfer model to better capture the consistency and heterogeneity across domains. The approach described in [3] considers the heterogeneity of the source and target domains, functioning in transfer-all and transfer-none mode depending on the distance between the two domains. Our work differs from [3], as the proposed model allows any portion of the latent factors to be transferred.

Most previous studies on recommendations using transfer learning focus on rating prediction tasks (as mentioned before, rating predictions are less effective for item recommendations). In this paper, we use transfer learning to identify the Top-K recommendations and overcome the issues of missing feedback.

3 MODELING SOCIAL AND ITEM VISIBILITIES

3.1 Social and Item Visibilities

As discussed in the Introduction, there are two possible reasons for missing observations, i.e., where no interaction between user u and item i is observed. The first reason is that user u has never had the opportunity to see item i, and therefore cannot possibly have consumed or clicked on the item. The second reason is that user u may have seen item i, but did not like it. Therefore, we introduce the **item visibility** as a random variable to depict whether item i is visible to user u (denoted as a_{ui}).

Moreover, a considerable number of social interactions between users may be missing. Similarly, there are two possible reasons for a missing social relationship between users u and k: the first is that user u does not know user k at all, and the second reason is that user u indeed knows user k, but does not want to be friends with (or trust) user k. Therefore, the **social visibility** (denoted as b_{uk}) is introduced to capture whether user u knows of user k.

In our model, $a_{ui} = 1$ implies that item i is visible to user u; otherwise, $a_{ui} = 0$. Similarly, $b_{uk} = 1$ implies that user k (referral) is visible to user u (referrer); otherwise, $b_{uk} = 0$. As the visibilities are closely related to the missing interactions, a proper model of the visibilities should distinguish whether missing or unobserved interactions are caused by the mismatch of user preferences. Therefore, the observed interactions are the co-product of user preferences and visibilities, and modeling both visibilities captures the user preferences more accurately. Previous studies on social networks [24] reveal that users tend to behave consistently with their friends, which verifies the correlation between the social and rating behavior of users. Therefore, we jointly consider both social and item visibilities in our model, and further adopt the idea of transfer learning to model the user preferences toward items and social connections.

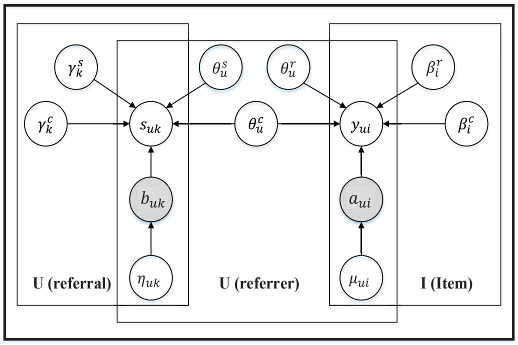

Figure 1: Transferring Social and Item Visibility (TranSIV), two visibility variables are shaded

3.2 Transfer Model with Social and Item Visibilities (TranSIV)

3.2.1 Model Description. Recall that two types of interactions are considered in our work, i.e., user–item interactions and user–user social interactions. These interactions are denoted as $y_{ui}, \forall u \in U, i \in I$ and $s_{uk}, \forall u, k \in U$, respectively. If there exists an observed interaction between user u and item i, $y_{ui} = 1$; otherwise, $y_{ui} = 0$; if there exists a user–user interaction between user u and user k, $s_{uk} = 1$; otherwise, $s_{uk} = 0$. As user–user interactions can be unilateral (such as one person following another on Twitter) or bilateral (such as friendship on Facebook), we denote user u as the referrer and user k as the referral for s_{uk}.

We adopt the idea of transfer learning to coordinate the social and rating information and propose the model shown in Fig. 1. The idea is to share a portion of the latent factors from user–user interactions to identify user–item interactions. As the two types of behavior are not entirely the same in nature, the knowledge learnt from user–user social behavior may only be partially related to user–item interactions, whereas the number of shared factors indicates how the two types of behavior are correlated. The variables of the proposed model are listed in Table 1. In the context of transfer learning, we name the target domain as the "rating domain" (although the user–item interactions can be ratings or binary values, we use the term rating for simplicity) and call the source domain the "social domain."

There are five latent factor vectors corresponding to a pair of users: θ_u^c, θ_u^r, and θ_u^s for user u as the referrer, and γ_k^c and γ_k^s for user k as the referral. For user u, θ_u^c represents the latent factors shared between social and rating behavior; θ_u^r and θ_u^s represent user latent factors corresponding to rating and social behavior, respectively. The item latent factor vectors and the social latent factor vectors for the referral user are denoted in a similar manner (see Table 1 for details).

A user–item interaction is the product of item visibility, the user latent vectors, and the item latent vectors: $y_{ui} = a_{ui}(\theta_u^c \beta_i^c + \theta_u^r \beta_i^r)$. A user–user interaction is the product of social visibility, the referrer's latent vectors, and the referral's latent vectors: $s_{uk} = b_{uk}(\theta_u^c \gamma_k^c + \theta_u^s \gamma_k^s)$. The visibilities are random variables with certain probabilities of being visible: $P(a_{ui} = 1) = \mu_{ui}$ and $P(b_{uk} = 1) = \eta_{uk}$.

Table 1: Variables and Notation

Variables	Meaning
U	The set of users
I	The set of items
y_{ui}	User–item interaction: i.e., whether user u has clicked/consumed item i
s_{uk}	User–user interaction: i.e., whether user u (as referrer) trusts (or is a friend of) user k (as referral)
a_{ui}	Item visibility: i.e., whether user u has seen item i
b_{uk}	User visibility: i.e., whether user u (as referrer) knows user k (as referral)
μ_{ui}	Parameter of the Bernoulli distribution for a_{ui}
η_{uk}	Parameter of the Bernoulli distribution for b_{uk}
(α_1, α_2)	Parameter of the Beta distribution for μ_{ui}
$(\alpha_1^{'}, \alpha_2^{'})$	Parameter of the Beta distribution for η_{uk}
θ_u^c	Common latent factor vector of user u as referrer
θ_u^r	Preference-specific latent factor vector of user u
θ_u^s	Social-specific latent factor vector of user u (as referrer)
β_i^c	Common latent factor vector of item i
β_i^r	Preference-specific latent factor vector of item i
γ_k^c	Common latent factor vector of user k (as referral)
γ_k^s	Social-specific latent factor vector of user k (as referral)

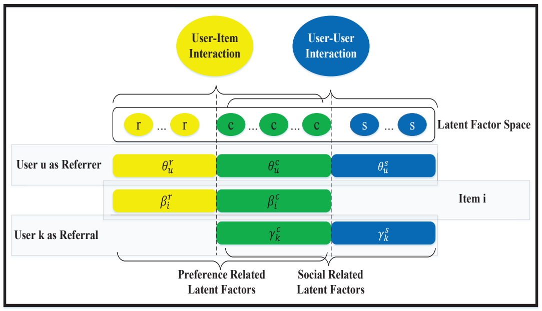

Figure 2: The Latent Factor Space of Two Interactions in TranSIV

A more intuitive explanation is presented in Fig. 2. The ellipses are scalars and the rectangles are vectors; The top ellipses are the binary variables indicating whether user–item (in yellow)/user–user (in blue) interactions are observed; the smaller ellipses are the latent factors in the latent factor space. The green ones represent the common latent factors of both rating and social domains; the yellow and blue ones represent the latent factors of rating and social domain specifically; The seven rectangles stand for the latent vectors of user u as referrer, item i and user k as referral respectively. The details of these vectors can be found in Table. 1

3.2.2 Generative Process of TranSIV. The generative process of the TranSIV model is as follows:

- for each user u as the referrer, draw the latent factor vectors from the Gaussian distributions: $\theta_u^c \sim N(0, \lambda_\theta^{-1} I^c)$, $\theta_u^r \sim N(0, \lambda_\theta^{-1} I^r)$, $\theta_u^s \sim N(0, \lambda_\theta^{-1} I^s)$;
- for each user k as the referral, draw the latent factor vectors from the Gaussian distributions: $\gamma_k^c \sim N(0, \lambda_\gamma^{-1} I^c)$; $\gamma_k^s \sim N(0, \lambda_\gamma^{-1} I^s)$;
- for each item i, draw the latent factor vectors from the Gaussian distributions: $\beta_i^c \sim N(0, \lambda_\beta^{-1} I^c)$, $\beta_i^r \sim N(0, \lambda_\beta^{-1} I^r)$;
- for each item i, draw the item visibility probability from the Beta distribution: $\mu_{ui} \sim Beta(\alpha_1, \alpha_2)$;

- for each user k as the referral, draw the social visibility probability from the Beta distribution: $\eta_{uk} \sim Beta(\alpha_1^{'}, \alpha_2^{'})$
- for each user–item pair u, i, draw the item visibility a_{ui} from the Bernoulli distribution: $a_{ui} \sim Bernoulli(\mu_{ui})$;
- for each user–user pair u, k, draw the social visibility b_{uk} from the Bernoulli distribution: $b_{uk} \sim Bernoulli(\eta_{uk})$
- for each user–item pair with $a_{ui} = 1$, draw the interaction y_{ui} from the Gaussian distribution: $y_{ui}|a_{ui} = 1 \sim N(\theta_u^{cT}\beta_i^c + \theta_u^{rT}\beta_i^r, \lambda_y^{-1})$
- for each user–user pair with $b_{uk} = 1$, draw the interaction s_{uk} from the Gaussian distribution: $s_{uk}|b_{uk} = 1 \sim N(\theta_u^{cT}\gamma_k^c + \theta_u^{sT}\gamma_k^s, \lambda_s^{-1})$

where I^c, I^s, and I^r are identity matrices with the same dimensions as θ_u^c, θ_u^s, and θ_u^r; $\lambda_\theta I^c$, $\lambda_\beta I^r$, $\lambda_\gamma I^s$, λ_y and λ_s are the precisions for the Gaussian distributions.

3.2.3 Model Inference. Given the generative model with hidden visibility variables, we use an EM (Expectation Maximization) algorithm to infer the parameters. In the E step, we estimate the hidden variables by taking expectations while keeping the other variables fixed; in the M step, we use the estimated hidden variables to infer the other variables.

E step: Items that have interactions with users are apparently visible to the corresponding users:

$$\begin{aligned} &E(a_{ui}|y_{ui} = 1) = 1, \forall u \in U, i \in I; \\ &E(b_{uk}|s_{uk} = 1) = 1, \forall u \in U, k \in U \end{aligned} \tag{1}$$

Considering the user–item and user–user pairs that constitute the missing interactions, the expectation of the hidden visibility variables (a_{ui} and b_{uk}) can be derived from the Bayes' Theorem:

$$\begin{aligned} &E(a_{ui}|y_{ui} = 0) \\ &= \frac{P(a_{ui} = 1, y_{ui} = 0)}{P(y_{ui} = 0|a_{ui} = 1)P(a_{ui} = 1) + P(a_{ui} = 0)} \\ &= \frac{\mu_{ui} \cdot N(0|\theta_u^{cT}\beta_i^c + \theta_u^{rT}\beta_i^r, \lambda_y^{-1})}{\mu_{ui} \cdot N(0|\theta_u^{cT}\beta_i^c + \theta_u^{rT}\beta_i^r, \lambda_y^{-1}) + 1 - \mu_{ui}} \\ &E(b_{uk}|s_{uk} = 0) \\ &= \frac{P(b_{uk} = 1, s_{uk} = 0)}{P(s_{uk} = 0|b_{uk} = 1)P(b_{uk} = 1) + P(b_{uk} = 0)} \\ &= \frac{\eta_{uk} \cdot N(0|\theta_u^{cT}\gamma_k^c + \theta_u^{sT}\gamma_k^s, \lambda_s^{-1})}{\eta_{uk} \cdot N(0|\theta_u^{cT}\gamma_k^c + \theta_u^{sT}\gamma_k^s, \lambda_s^{-1}) + 1 - \eta_{uk}} \end{aligned} \tag{2}$$

We denote $E(a_{ui})$ as p_{ui} and $E(b_{uk})$ as q_{uk}; given p_{ui} and q_{uk}, we can estimate the other parameters in the following M step by fixing the hidden variables with their expectations.

M step: First, we update the item and social visibility probabilities of the Bernoulli distributions. As the Beta distributions are conjugate to the Bernoulli distributions, the parameters of the Bernoulli distributions can be updated as follows:

$$\begin{aligned} \mu_{ui} &= \frac{\alpha_1 + \sum_u p_{ui} - 1}{\alpha_1 + \alpha_2 + |U| - 2}, \forall i; \\ \eta_{uk} &= \frac{\alpha_1' + \sum_u q_{uk} - 1}{\alpha_1' + \alpha_2' + |U| - 2}, \forall k \end{aligned} \tag{3}$$

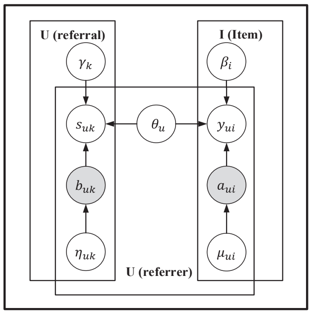

Figure 3: Full Transfer Model of Social and Item Visibilities (FTranSIV).

Then, we update the preference latent factor vectors of users and items as well as the social latent factor vectors. The update process employs Alternating Least-Squares optimization [16].

$$\begin{aligned} &\theta_u^c \leftarrow (\lambda_y \sum_i p_{ui}\beta_i^c\beta_i^{cT} + \lambda_s \sum_k q_{uk}\gamma_k^c\gamma_k^{cT} + \lambda_\theta I^c)^{-1} \\ &(\sum_i \lambda_y p_{ui}(y_{ui} - \theta_u^{rT}\beta_i^r)\beta_i^c + \sum_k \lambda_s q_{uk}(s_{uk} - \theta_u^{sT}\gamma_k^s)\gamma_k^c) \\ &\theta_u^r \leftarrow (\lambda_y \sum_i p_{ui}\beta_i^r\beta_i^{rT} + \lambda_\theta I^r)^{-1}(\sum_i \lambda_y p_{ui}(y_{ui} - \theta_u^{cT}\beta_u^c)\beta_i^r) \\ &\theta_u^s \leftarrow (\lambda_s \sum_k q_{uk}\gamma_k^s\gamma_k^{sT} + \lambda_\theta I^s)^{-1}(\sum_k \lambda_s q_{uk}(s_{uk} - \theta_u^{cT}\gamma_u^c)\gamma_k^s) \\ &\beta_i^c \leftarrow (\lambda_y \sum_u p_{ui}\theta_u^c\theta_u^{cT} + \lambda_\beta I^c)^{-1}(\sum_u \lambda_y p_{ui}(y_{ui} - \theta_u^{rT}\beta_i^r)\theta_u^c) \\ &\beta_i^r \leftarrow (\lambda_y \sum_u p_{ui}\theta_u^r\theta_u^{rT} + \lambda_\beta I^r)^{-1}(\sum_u \lambda_y p_{ui}(y_{ui} - \theta_u^{cT}\beta_i^c)\theta_u^r) \\ &\gamma_k^c \leftarrow (\lambda_s \sum_u q_{uk}\theta_u^c\theta_u^{cT} + \lambda_\gamma I^c)^{-1}(\sum_u \lambda_s q_{uk}(s_{uk} - \theta_u^{sT}\gamma_k^s)\theta_u^c) \\ &\gamma_k^s \leftarrow (\lambda_s \sum_u q_{uk}\theta_u^s\theta_u^{sT} + \lambda_\gamma I^s)^{-1}(\sum_u \lambda_s q_{uk}(s_{uk} - \theta_u^{cT}\gamma_k^c)\theta_u^s) \end{aligned} \tag{4}$$

To make recommendations, we consider both the visibility and user preference with respect to the items, i.e., a click or consumption behavior can only happen when the user actually sees the item and then makes a purchase. Therefore, for each user $u \in U$, the expectation of the missing feedback w.r.t item i is:

$$E(y_{ui}|\theta_u^c, \beta_i^c, \theta_u^r, \beta_i^r, \mu_{ui}) = \mu_{ui}(\theta_u^{cT}\beta_i^c + \theta_u^{rT}\beta_i^r) \tag{5}$$

We present the model inference and recommendation process in Algorithm 1.

3.3 Simplified Model: Full Transfer Model with Social and Item Visibilities (FTranSIV)

As TranSIV transfers a portion of the latent factors from the social domain to the rating domain, the model contains a large number of latent factor vectors. To simplify the model, we can set the latent factors to be the same in both domains, so that the knowledge of the social domain is fully transferred. In this special case, the model is the Full Transfer model of Social and Item Visibility (FTranSIV). This model is illustrated in Fig. 3.

Algorithm 1 Recommendation Algorithm with TranSIV

Input: The observed rating matrix $\mathbf{Y} : \{y_{ui}, \forall u \in U, i \in I\}$, the observed social matrix $\mathbf{S} : \{s_{uk}, \forall u, k \in U\}$
Output: Top-K recommendation lists for users $\mathbf{U}$.

```
Initialize the latent factor vectors $\theta_u^c, \theta_u^s, \theta_u^r, \forall u \in \mathbf{U}$,
    $\beta_i^c, \beta_i^r, \forall i \in \mathbf{I}$, $\gamma_k^c, \gamma_k^s, \forall k \in \mathbf{U}$, and the visibility probabilities
    $\mu_{ui}, \forall i \in \mathbf{I}$, $\eta_{uk}, \forall k \in \mathbf{U}$;
while Not convergent and iter $\leq$ MaxIter do
    E step:
    for each user $u$ and item $i$ for which $y_{ui} = 0$ do
        Compute $E(a_{ui}|y_{ui} = 0)$ following Eq. 2;
    end for
    for each user pair $u$ and user $k$ for which $s_{uk} = 0$ do
        Compute $E(b_{uk}|s_{uk} = 0)$ following Eq. 2;
    end for
    M step:
    Update user and item factor vectors: $\theta_u^c, \theta_u^s, \theta_u^r, \forall u \in \mathbf{U}$,
        $\beta_i^c, \beta_i^r, \forall i \in \mathbf{I}$, $\gamma_k^c, \gamma_k^s, \forall k \in \mathbf{U}$ following Eq. 4;
    Update the visibility probabilities:
        $\mu_{ui}, \forall i \in \mathbf{I}$, $\eta_{uk}, \forall k \in \mathbf{U}$ following Eq. 3;
end while
Make recommendations for each user $u \in \mathbf{U}$ following Eq. 5
    by selecting K items $i \in \mathbf{I}$ with highest $E(y_{ui})$ that have no
    observed interactions with $u$;
```

Note that all user latent factors are shared between the social domain and the rating domain. Therefore, $\theta_u^r = 0$, $\theta_u^s = 0$, $\beta_i^r = 0$, $\gamma_k^s = 0$. Additionally, θ_u^c, β_i^c, and γ_k^c are denoted as θ_u, β_i, and γ_k, respectively.

As a result, the inference process is simplified, the related parameters can be updated in a similar manner as before, and the only difference is the replacement of θ_u^c, β_i^c, and γ_k^c with θ_u, β_i, and γ_k, respectively. Moreover, the other latent factor vectors are set to 0 in the inference process. Experiments comparing the performance of FTranSIV and TranSIV indicate that selective transfer achieves superior performance, but the simplified modeling does not significantly affect the recommendation accuracy.

4 EXPERIMENTS

In this section, we present the results of experimental evaluations conducted on real-world datasets, including **Ciao**[1], **Epinions**[2], and **Flixter**[3]. All datasets contain rating information given by users and the in-site social connections between users. The datasets were preprocessed so that all items have at least five ratings. The social connections in the first two datasets are unilateral trust relationships, which means that when user A trusts user B, user B does not necessarily trust user A. Flixter has a friendship mechanism that is bilateral for both users. The statistical details of these datasets are presented in Table 2.

As long as there exists some user–user or user–item interaction, the corresponding rating is assigned a value of 1 (as implicit feedback), which is the procedure adopted in state-of-the-art methods [11, 19, 30]. We compare our approach with five state-of-the-art algorithms for recommendation based on implicit feedback, namely WMF [12], BPR [30], EXPOMF [19], SBPR [42], and SPF [4]. The first two algorithms are commonly used in item recommendation tasks and exhibit excellent performance, while the latter two are representative social-aware recommendation algorithms that deal with implicit feedback. The characteristics of the comparative approaches are listed in Table 3.

Table 2: Statistical details of the datasets

Datasets	#Users	#Items	#Ratings	#Links	Link Type
Ciao	7,267	11,211	147,995	111,781	Unilateral
Epinions	38,089	23,585	488,917	433,416	Unilateral
Flixter	147,229	17,318	8,093,735	2,420,211	Bilateral

- WMF [12]: Weighted Matrix Factorization using a pointwise optimization strategy for implicit user–item feedback.
- BPR [30]: a classic method that compares observed and missing feedback in a pairwise manner, coupled with matrix factorization for item scoring.
- EXPOMF [19]: a probabilistic model of item exposure in recommender systems; uses matrix factorization for pointwise scoring.
- SBPR [42]: a ranking model that considers social relationships in the learning process, assuming that users tend to assign higher ranks to items that their friends prefer. The negative instances are sampled in a similar manner with BPR.
- SPF [4]: a probabilistic model that performs social Poisson factorization. SPF incorporates user latent preferences for items with latent friend influences, thus matching user preferences with social friends when generating the top-N recommendations.

Table 3: Comparison of the Approaches

Characteristics	WMF	BPR	EXPOMF	SBPR	SPF	TranSIV
Item Interaction	√	√	√	√	√	√
Social Interaction	\	\	\	√	√	√
Item Visibility	\	\	√	\	\	√
Social Visibility	\	\	\	\	\	√

To evaluate the performance of all algorithms, we calculated the **Recall@K**, **MAP@K**, and **NDCG@K**, where $rel_i = 1/0$ indicates whether the item at rank i in the Top-K recommendation list is in the test set. For each user, these metrics can be computed as follows (each metric is the average for all users, and the mean average precision (**MAP**) is the average of all user **AP**s). The notion **IDCG** means the maximum possible **DCG** through ideal ranking. y_u^{test} denotes the items rated by user u in the testing set.

$$\mathbf{Recall@K} = \frac{\sum_{i=1}^{K} rel_i}{\min(K, |y_u^{test}|)};$$

$$\mathbf{AP@K} = \sum_{n=1}^{K} \frac{\frac{\sum_{i=1}^{n} rel_i}{n} \times rel_n}{\min(K, |y_u^{test}|)};$$

$$\mathbf{DCG@K} = \sum_{i=1}^{K} \frac{2^{rel_i} - 1}{log_2(i+1)}; \mathbf{NDCG@K} = \frac{\mathbf{DCG@K}}{\mathbf{IDCG@K}}$$

To evaluate different recommendation lengths, we experimented using $K = 10$, 50, and 100. The parameters for the

[1] http://www.jiliang.xyz/trust.html
[2] https://alchemy.cs.washington.edu/data/epinions/
[3] http://www.cs.ubc.ca/jamalim/datasets/

Table 4: Performance comparison on three datasets (Ciao, Epinions, Flixter). “” denotes that the result is better than all baselines with a significance test of $p < 0.01$**

Ciao	Recall@10	Recall@50	Recall@100	NDCG@10	NDCG@50	NDCG@100	MAP@10	MAP@50	MAP@100
WMF	0.0753	0.1611	0.2149	0.0631	0.0905	0.1048	0.0334	0.0368	0.0384
BPR	0.0616	0.1520	0.2080	0.0503	0.0845	0.0948	0.0253	0.0328	0.0320
SBPR	0.0620	0.1557	0.2124	0.0520	0.0849	0.0997	0.0262	0.0334	0.0350
SPF	0.0599	0.1147	0.1536	0.0534	0.0696	0.0796	0.0295	0.0299	0.0308
EXPOMF	0.0751	0.1612	0.2149	0.0626	0.0899	0.1041	0.0331	0.0365	0.0380
TranSIV	0.0800**	0.1710**	0.2309**	0.0663**	0.0948**	0.1104**	0.0354**	0.0387**	0.0404**
Epinions	Recall@10	Recall@50	Recall@100	NDCG@10	NDCG@50	NDCG@100	MAP@10	MAP@50	MAP@100
WMF	0.0617	0.1386	0.1891	0.0475	0.0710	0.0836	0.0256	0.0290	0.0304
BPR	0.0510	0.1353	0.1934	0.0374	0.0643	0.0785	0.0196	0.0248	0.0262
SBPR	0.0556	0.1429	0.1963	0.0420	0.0696	0.0841	0.0224	0.0275	0.0290
SPF	0.0298	0.0761	0.1135	0.0245	0.0382	0.0472	0.0127	0.0142	0.0149
EXPOMF	0.0613	0.1400	0.1919	0.0474	0.0716	0.0845	0.0256	0.0292	0.0305
TranSIV	0.0649**	0.1514**	0.2095**	0.0495**	0.0759**	0.0900**	0.0268**	0.0308**	0.0323**
Flixter	Recall@10	Recall@50	Recall@100	NDCG@10	NDCG@50	NDCG@100	MAP@10	MAP@50	MAP@100
WMF	0.3610	0.4951	0.5723	0.3030	0.3469	0.3724	0.2245	0.2282	0.2337
BPR	0.1612	0.3337	0.4366	0.1227	0.1765	0.2061	0.0893	0.1005	0.1046
SBPR	0.3145	0.5001	0.5948	0.2641	0.3400	0.3734	0.2018	0.2291	0.2383
SPF	0.1756	0.2837	0.3628	0.1405	0.1719	0.1936	0.0954	0.0938	0.0947
EXPOMF	0.3788	0.4947	0.5684	0.3248	0.3543	0.3760	0.2484	0.2379	0.2391
TranSIV	0.3882**	0.5272**	0.6119**	0.3276**	0.3623**	0.3855**	0.2507**	0.2415**	0.2427**

baseline algorithms were initialized as in the corresponding papers, and were then carefully tuned to achieve optimal performance. The dimensions of the latent factor vectors were set to 20, 100, and 50 for Ciao, Epinions, and Flixter, respectively. For TranSIV, the shared dimension was set to be 80% of the total dimension by default; therefore, the dimensions of $\theta^c, \theta^r, \theta^s$ were (16,4,4), (80,20,20), and (40,10,10) for the three datasets, respectively. The dimensions of β^c and γ^c are the same as that of θ^c, and the dimensions of β^r and γ^s are the same as those of θ^r and θ^s.

4.1 Comparative Analysis on Overall Performances

We conducted a four-fold cross-validation, with three folds used for training and the remaining fold used for testing. The experiment was conducted four times, and the average results are presented in Table 4.

From these results, several observations can be made:

- The methods that model item visibility (EXPOMF and TranSIV) generally outperform those methods that do not consider visibility. The visibility-related methods model missing feedback as negative feedback with corresponding probabilities, and this finer-grained modeling increases the learning accuracy of user preferences.
- Methods utilizing social information usually outperform those without social information. In Table 4, for example, the performance of SBPR is better than that of BPR, and TranSIV performs better than EXPOMF and BPR. This is not surprising, as social information is complementary to ratings. However, social information may also contain noise that is inconsistent with the rating behavior of users. Therefore, a proper model that coordinates social information may perform better. The transfer learning procedure captures this correlation and heterogeneity simultaneously. The effect of transfer learning will be discussed in detail in Section 5.
- As shown in Table 4, TranSIV outperforms SBPR and SPF. Although social information is useful for recommendations, the performance can vary depending on how the social information is utilized. The results show that methods which consider social visibility outperform those that do not. Visibility information allows missing social interactions to be modeled with finer granularity, which is the advantage of TranSIV.
- Considering the performance on each dataset, we find the improvements of TranSIV depend on the sparsity of the dataset. The Flixter dataset is relatively dense in terms of user–item interactions (averaging 54.97 interactions per user, compared with 20.37 and 12.84 for Ciao and Epinions, respectively). The user preferences are more difficult to learn from sparse user–item interactions, but can be enriched by the knowledge learnt from social interactions. Thus, the transfer learning procedure is more useful on sparse datasets. To make further verifications, we conduct experiments on less training data and the results are presented in Section 4.2.

4.2 Performance with Less Training Data

We also conducted experiments using different proportions of the training data to validate the sensitivity of our model in terms of data sufficiency. Each dataset was split into four folds, and one, two, and three folds were successively used for training, while the remainder were used for testing. We used all of the social information in the social-aware algorithms (SBPR, SPF, and TranSIV). The results are similar for all

Table 5: Performance comparison with 25% and 50% data for training on Ciao, **: $p < 0.01$

25%	**Recall@10**	**Recall@50**	**Recall@100**	**NDCG@10**	**NDCG@50**	**NDCG@100**	**MAP@10**	**MAP@50**	**MAP@100**
WMF	0.0566	0.0804	0.1041	0.0583	0.0636	0.0719	0.0284	0.0208	0.0211
BPR	0.0363	0.1075	0.1485	0.0388	0.0750	0.0909	0.0151	0.0250	0.0281
SBPR	0.0366	0.1117	0.1537	0.0399	0.0780	0.0934	0.0157	0.0261	0.0281
SPF	0.0582	0.0750	0.1055	0.0630	0.0624	0.0720	0.0335	0.0218	0.0217
EXPOMF	0.0595	0.0843	0.1095	0.0619	0.0670	0.0756	0.0308	0.0227	0.0230
TranSIV	**0.0834****	**0.1226****	**0.1590****	**0.0863****	**0.0862****	**0.1085****	**0.0448****	**0.0356****	**0.0363****

50%	**Recall@10**	**Recall@50**	**Recall@100**	**NDCG@10**	**NDCG@50**	**NDCG@100**	**MAP@10**	**MAP@50**	**MAP@100**
WMF	0.0689	0.1280	0.1700	0.0658	0.0853	0.0986	0.0323	0.0307	0.0321
BPR	0.0497	0.1357	0.1845	0.0499	0.0863	0.1016	0.0216	0.0304	0.0319
SBPR	0.0514	0.1370	0.1883	0.0513	0.0890	0.1049	0.0226	0.0318	0.0337
SPF	0.0611	0.0968	0.1321	0.0623	0.0706	0.0811	0.0329	0.0269	0.0275
EXPOMF	0.0726	0.1307	0.1726	0.0702	0.0886	0.1019	0.0352	0.0324	0.0338
TranSIV	**0.0847****	**0.1537****	**0.2052****	**0.0822****	**0.1038****	**0.1198****	**0.0423****	**0.0395****	**0.0412****

three datasets; the results using 25% and 50% of the Ciao data for training are presented in Table 5.

It is known that insufficient training data present significant difficulties in personalized recommendations. In real-world scenarios, dealing with scarce feedback is a common problem. As can be seen in Table 5, TranSIV still outperforms the baselines. As less training data implies more missing feedback, our algorithm exhibits higher accuracy (compared with others) when there is more missing feedback. Considering that social and consumption behavior are correlated, the knowledge learnt from social behavior can compensate for the shortage of user feedback on consumption. As a result, the use of social information produces a great improvement when the training data are scarce. This can be verified by comparing the improvements in social-aware algorithms over non-social methods in Tables 4 and 5.

Furthermore, TranSIV generally achieves more of an improvement when the training data are scarce (especially when K is relatively small, say $K = 10$). This observation coincides with the conclusion in [18], which states that transfer learning contributes even more when data in the target domain is sparse.

4.3 Impact of transferred knowledge

As stated in previous sections, the social domain and rating domain are correlated, but heterogeneous in nature. Therefore, the social information may add noise into the recommender system. The portion of latent factors that should be transferred reflects how knowledge from the social domain is passed to the rating domain.

The dimension of θ_u^c determines how many latent factors are shared between user–user and user–item interactions. By altering the dimension of θ_u^c with a stepsize of 20% (when the full dimension is 100 and Ratio=20%, the dimensions of θ_u^c, θ_u^r, and θ_u^s are 20, 80, 80), we compared the sensitivity of our algorithm to the proportion of transferred knowledge. Note that a ratio of 0% corresponds to EXPOMF and a ratio of 100% corresponds to FTranSIV. The results using the Epinions and Ciao datasets are presented in Fig. 4.

As illustrated in the results, a suitable proportion of shared factors is needed to achieve optimal performance. The specific number of shared latent factors that gives the best performance may vary with the dataset (for example, the numbers for Epinions and Ciao are slightly different). However, empirical investigations suggest that 80% of shared factors achieves good performance. On the Epinions dataset, the performance improves as the transfer ratio increases from 0–80%, achieving optimal performance around 60–80%; while sharing all the latent factors (ratio=100%) does not lead to the best performance. This illustrates that the user–user and user–item interactions indeed share similar information that can be leveraged through shared factors, but they also contain unique information that requires independent factors to capture.

When no latent factors are transferred, i.e., the ratio is 0%, the knowledge learnt from the social domain is not adapted into the rating domain. Therefore the model only utilizes item visibility for preference modeling, which leads to suboptimal performance.

Moreover, given a 100% transfer ratio, FTranSIV contains fewer latent factor vectors (three types of latent vectors: $\theta_u, \forall u \in U$; $\beta_i, \forall i \in I$; $\gamma_k, \forall k \in U$) and it degenerates to a simplified version of TranSIV. We find that the performance of FTranSIV worsens, but remains superior to that of the baselines and the case when no latent factors are transferred.

5 DISCUSSION

As transfer learning is applied to coordinate social information with that from the rating domain, it would be instructive to evaluate the performance of transfer learning in comparison with another model that utilizes social information for visibility modeling. To the best of our knowledge, no previous methods applied visibility modeling with social information. Therefore, we adopted the idea of [19] and incorporated auxiliary information to model visibility in recommender systems. More specifically, we constructed the Social Content Visibility Matrix Factorization (SCVMF) model.

First, we constructed a social content vector for each user through the network embedding procedure [8], then used the social embedding vector as the input feature for visibility modeling. Then we applied a similar idea to Content

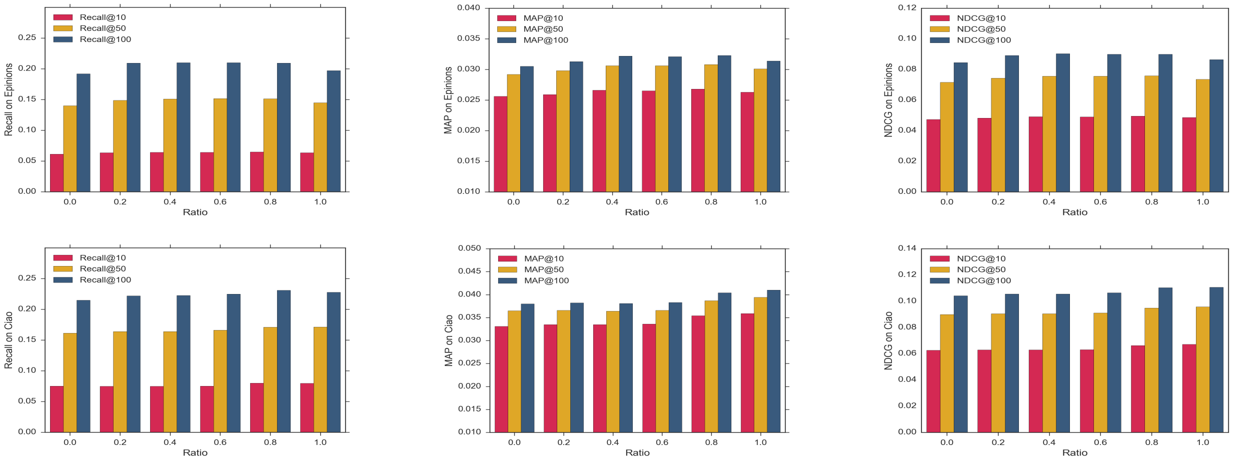

Figure 4: Performance of TranSIV with different dimensions of θ^c on Epinions and Ciao datasets (Ratio refers to the proportion of shared latent factors; Ratio = 1.0 refers to FTranSIV)

ExpoMF [19] and used the logistic regression model to capture social-incorporated visibility. This visibility was then adopted to learn the user preferences.

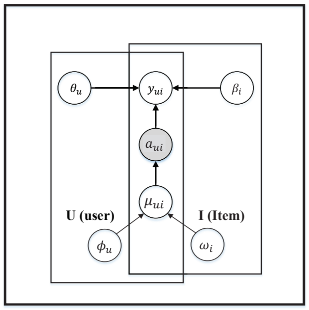

Figure 5: Social Content Visibility Matrix Factorization (SCVMF).

The graphical model of SCVMF is presented in Fig. 5. In this model, θ_u and β_i are the user and item latent factor vectors; a_{ui} is the visibility variable and μ_{ui} is the probability of $a_{ui} = 1$. μ_{ui} is assumed to be a logistic regression outcome of social embedding features (ϕ_u), where ω_i is the item-specific weight:

$$\mu_{ui} = \frac{1}{1 + exp(-\phi_u \omega_i)} \tag{6}$$

The social embedding vector ϕ_u is generated from a state-of-the-art network embedding technique [8], where the social adjacency matrix is used as the network. Then ϕ_u is used as input feature for visibility modeling. The basic idea behind this model is that the item visibility is associated with the social connections of users and the interaction between the items and users is a product of item visibility and the latent factor vectors of the users and items. The model is learnt via the SGD (Stochastic Gradient Descent) method. Note that there can be other approaches for incorporating social information and visibility (although no existing studies have done so); we leave this as one direction for future work.

The transfer learning procedure was evaluated in comparison with the SCVMF model. We conducted experiments on the Ciao dataset with both TranSIV and SCVMF (*in comparison with EXPOMF*), using latent factor vectors of dimensions consistent with the experimental settings in previous sections. The results are presented in Table 6.

Table 6: Performance Comparison between SCVMF and TranSIV on Ciao, **: $p < 0.01$

Ciao	**Recall@10**	**NDCG@10**	**MAP@10**
EXPOMF	0.0751	0.0626	0.0331
SCVMF	0.0754	0.0641	0.0345
TranSIV	**0.0800****	**0.0663****	**0.0354****

As shown in the table, TranSIV outperforms SCVMF with all metrics. As TranSIV and SCVMF both exploit visibility for user preference modeling, transfer learning clearly uses social information more effectively. Comparing SCVMF and EXPOMF, we find that the incorporation of social information into exposure modeling leads to some improvements over EXPOMF, which does not consider social information. Despite its modeling of item visibilities, the social information is under-utilized in this way, whereas transfer learning directly incorporates social information into the modeling of user preferences and visibility. This illustrates the benefits of transfer learning in terms of the recommendation accuracy given by TranSIV.

6 CONCLUSION

In this paper, we examined the problem of using missing user feedback in social-aware recommender systems, which is an important challenge in state-of-the-art personalized recommendations. A novel unified model that adopts social and item visibilities with transfer learning (TranSIV) has been proposed to solve this problem.

We believe that modeling visibility is vital to recommender systems, because a missing feedback does not necessarily mean that a user dislikes an item – it may be simply because that the user had no chance to see the item at all. However, this simple intuition is largely ignored in most of the current recommendation algorithms. In the future, we aim to further study the nature of visibility in personalized recommendation systems. Specifically, we will extend the idea of visibility and transfer learning in the proposed approaches to cross-domain (i.e., different categories) recommendation. Another interesting direction is to consider the recommendation problem in signed social networks, where both likes and dislikes (or trust and distrust) are explicitly labeled [7, 34].

7 ACKNOWLEDGEMENT

This work is supported by Natural Science Foundation of China (Grant No. 61532011, 61672311) and National Key Basic Research Program (2015CB358700). Part of the work is sponsored by the Center for Intelligent Information Retrieval.

REFERENCES

[1] Immanuel Bayer, Xiangnan He, Bhargav Kanagal, and Steffen Rendle. 2017. A Generic Coordinate Descent Framework for Learning from Implicit Feedback. In *WWW*.

[2] Jingwen Bian, Yang Yang, and Tat-Seng Chua. 2013. Multimedia Summarization for Trending Topics in Microblogs. In *CIKM*.

[3] Bin Cao, Sinno Jialin Pan, Yu Zhang, Dit Yan Yeung, and Qiang Yang. 2010. Adaptive Transfer Learning. In *AAAI*.

[4] Allison JB Chaney, David M Blei, and Tina Eliassi-Rad. 2015. A probabilistic model for using social networks in personalized item recommendation. In *Recsys*.

[5] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Recsys*.

[6] Eric Eaton and Marie desJardins. 2011. Selective Transfer Between Learning Tasks Using Task-Based Boosting.. In *AAAI*.

[7] Rana Forsati, Mehrdad Mahdavi, Mehrnoush Shamsfard, and Mohamed Sarwat. 2014. Matrix factorization with explicit trust and distrust side information for improved social recommendation. *Transactions on Information Systems (TOIS)* (2014).

[8] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *KDD*.

[9] 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 *AAAI*.

[10] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *WWW*.

[11] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast Matrix Factorization for Online Recommendation with Implicit Feedback. In *SIGIR*.

[12] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *ICDM*.

[13] Mohsen Jamali and Martin Ester. 2009. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In *KDD*.

[14] Mohsen Jamali and Martin Ester. 2009. Using a trust network to improve top-N recommendation. In *Recsys*.

[15] Meng Jiang, Peng Cui, Fei Wang, Qiang Yang, Wenwu Zhu, and Shiqiang Yang. 2012. Social recommendation across multiple relational domains. In *CIKM*.

[16] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* (2009).

[17] Artus Krohn-Grimberghe, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2012. Multi-relational matrix factorization using bayesian personalized ranking for social network data. In *WSDM*.

[18] Bin Li, Qiang Yang, and Xiangyang Xue. 2009. Transfer learning for collaborative filtering via a rating-matrix generative model. In *ICML*.

[19] Dawen Liang, Laurent Charlin, James McInerney, and David M Blei. 2016. Modeling user exposure in recommendation. In *WWW*.

[20] Zhongqi Lu, Erheng Zhong, Lili Zhao, Wei Xiang, Weike Pan, and Qiang Yang. 2012. Selective Transfer Learning for Cross Domain Recommendation. *Computer Science* (2012).

[21] Hao Ma. 2014. On measuring social friend interest similarities in recommender systems. In *SIGIR*.

[22] Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. 2008. Sorec: social recommendation using probabilistic matrix factorization. In *CIKM*.

[23] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *WSDM*.

[24] Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* (2001).

[25] Xia Ning and George Karypis. 2011. Slim: Sparse linear methods for top-n recommender systems. In *ICDM*.

[26] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. 2008. One-class collaborative filtering. In *ICDM*.

[27] Weike Pan, Evan Wei Xiang, Nathan Nan Liu, and Qiang Yang. 2010. Transfer Learning in Collaborative Filtering for Sparsity Reduction.. In *AAAI*.

[28] Dimitrios Rafailidis and Fabio Crestani. 2016. Joint Collaborative Ranking with Social Relationships in Top-N Recommendation. In *CIKM*.

[29] Steffen Rendle and Christoph Freudenthaler. 2014. Improving pairwise learning for item recommendation from implicit feedback. In *WSDM*.

[30] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*.

[31] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. *Introduction to recommender systems handbook*. Springer.

[32] Suman Deb Roy, Tao Mei, Wenjun Zeng, and Shipeng Li. 2013. Towards cross-domain learning for social video popularity prediction. *IEEE Transactions on multimedia* (2013).

[33] Harald Steck. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Recsys*.

[34] Jiliang Tang, Charu Aggarwal, and Huan Liu. 2016. Recommendations in signed social networks. In *WWW*.

[35] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. 2013. Exploiting Local and Global Social Context for Recommendation.. In *IJCAI*.

[36] Jiliang Tang, Suhang Wang, Xia Hu, Dawei Yin, Yingzhou Bi, Yi Chang, and Huan Liu. 2016. Recommendation with Social Dimensions. In *AAAI*.

[37] Maksims Volkovs and Guang Wei Yu. 2015. Effective Latent Models for Binary Feedback in Recommender Systems. In *SIGIR*.

[38] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social Recommendation with Strong and Weak Ties. In *CIKM*.

[39] Markus Weimer, Alexandros Karatzoglou, Quoc V. Le, and Alex J. Smola. 2008. COFI RANK - Maximum Margin Matrix Factorization for Collaborative Ranking. In *NIPS*.

[40] Bo Yang, Yu Lei, Jiming Liu, and Wenjie Li. 2016. Social collaborative filtering by trust. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2016).

[41] Mao Ye, Xingjie Liu, and Wang-Chien Lee. 2012. Exploring social influence for recommendation: a generative model approach. In *SIGIR*.

[42] Tong Zhao, Julian McAuley, and Irwin King. 2014. Leveraging social connections to improve personalized ranking for collaborative filtering. In *CIKM*.

[43] Zhou Zhao, James Cheng, Furu Wei, Ming Zhou, Wilfred Ng, and Yingjun Wu. 2014. SocialTransfer: Transferring Social Knowledge for Cold-Start Cowdsourcing. (2014).

[44] Erheng Zhong, Wei Fan, and Qiang Yang. 2014. User behavior learning and transfer in composite social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* (2014).

[45] Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. 2011. Heterogeneous Transfer Learning for Image Classification.. In *AAAI*.