KURAM VE UYGULAMADA EĞİTİM BİLİMLERİ EDUCATIONAL SCIENCES: THEORY & PRACTICE

Received: November 5, 2017
Revision received: March 25, 2018
Accepted: March 27, 2018

www.estp.com.tr
DOI 10.12738/estp.2018.5.121 ⬩ October 2018 ⬩ 18(5) ⬩ 2217-2234

Research Article

A Trust-aware Neural Collaborative Filtering for E-learning Recommendation *

Xiaoyi Deng[1]
Huaqiao University

Hailin Li[2]
Huaqiao University

Feifei Huangfu[3]
Huaqiao University

Abstract

Social networks can provide massive quantities of information for communication among users and e-learning communities, and the trust relationships can been employed to reveal users' preferences for improving the performance of e-learning recommendation that aim to mitigate information overload and provide users with the most attractive and relevant learning resources. However, the data sparsity problem degrades recommending performance significantly. To address this problem, a novel trust-aware neural collaborative filtering model is proposed for exploiting multi-sourced information (resource content, user rating and social trust) to predict ratings in e-learning environment. We first ties deep neural network and collaborative topic regression together, to perform users and resources latent factors learning from resource content information and users rating data. Then, we incorporate social trust into rating prediction in our model, in which users' decisions regarding ratings are affected by their preferences and the favors of their trusted friends. In addition, an approach to calculating the maximum a posteriori estimates (MAP) is proposed to learn model parameters. Empirical experiments using two real-world datasets are conducted to evaluate the performance of our model. The results indicate that the proposed model has better accuracy and robustness than other methods for making recommendations in e-learning environment.

Keywords

Recommender Systems • E-learning • Collaborative Topic Regression • Deep Neural Network • Social Trust

* This work are supported by the National Natural Science Foundation of China (No.71401058 and No. 71771094), the Program for New Century Excellent Talents in Fujian Province University, NCETFJ (No. Z1625110), and the Project of Science and Technology Plan of Fujian Province of China (No.2017H01010065).

[1] **Correspondence to:** Xiaoyi Deng (PhD), Business School, Huaqiao University, Quanzhou 362021, China; Research Center for Applied Statistics and Big Data, Huaqiao University, Xiamen 361021, China. Email: londonbell@hqu.edu.cn

[2] Business School, Huaqiao University, Quanzhou 362021, China; Research Center for Applied Statistics and Big Data, Huaqiao University, Xiamen 361021, China. Email: hailin@hqu.edu.cn

[3] College of Foreign Languages, Huaqiao University, Quanzhou 362021, China. Email: huangfu@hqu.edu.cn

Citation: Deng, X. Y., Li, H. L., & Huangfu, F. F. (2018). A Trust-aware Neural Collaborative Filtering for E-learning Recommendation. *Educational Sciences: Theory & Practice, 18(5),* 2217-2234. http://dx.doi.org/10.12738/estp.2018.5.121

With the rapid advancement of internet and e-learning technologies, a huge amount of e-learning resources has been generated and distributed over learning social networks that can significantly facilitate individual learning (Cela, Sicilia, & Sánchez, 2015). However, vast quantities of information in social networks cause information overload issues (Bobadilla, Ortega, Hernando, & Gutiérrez., 2013), which hinders the ability of users without sufficient background knowledge to locate suitable or useful resources for their learning. The need to solve the information overload issue has caused the prevalence of recommender systems, which receive users information from their regarding resources of interest and recommend resources may fit their needs.

The core of a recommender system usually relies on well-known algorithms, such as content-based filtering (CBF), collaborative filtering (CF) (Su, & Khoshgoftaar, 2009; Mothe, & Rakotonirina, 2018), matrix factorization (MF) (Koren, & Bell, 2009; Najim, 2016), latent features (Koren, & Bell, 2011), neural networks (Wang, Xie, & Yan, 2016; Hu, Tang, Pan, Song, & Wen, 2016; Zhang, Yao, & Sun, 2017) and graph-based methods (Song, Zhang, & Giles, 2011), which have been developed over recent decades and are extensively applied in many recommender systems and internet-related fields, such as Amazon, Google and Taobao. These exsiting applications can increase the adoption andparticipation of new and existing users, which promote the success of social network applications. Compared to standard recommender systems, the e-learning recommender systems introduce new challenges that each e-learner uses their own tools, methods, paths, collaborations and processes (Klašnja-Milićević, Ivanović, & Nanopoulos, 2015), and only a few works have utilized all of these resources. Consequently, the user learning process must personalize to an extreme extent, which leads to utilizing users' profiles, learning activities and social relations to recommend learning resources that meet the characteristics and interests of e-learners. Therefore, using massive amount of information about items (learning resources), profiles of users and users' social correlations to facilitate recommendation has become a controversial topic (Groh, Birnkammerer, & Köllhofer, 2012).

Existing studies primarily focus on exploiting item-specific (resource-specific) or user-specific information to address data sparsity and cold-start problems. However, these studies have employed either item content or user information, without taking advantage of both resources. Consequently, several exploratory studies have been proposed to address this issue. Koren *et al*., (2009) proposed a MF model that maps both users and items to a joint latent factor space and later characterized each entity with a feature vector inferred from the existing ratings, which provided better recommendation results than the neighborhood-based models. Ma *et al*. (2008) proposed a factor analysis approach that is based on probabilistic MF to solve data sparsity and poor prediction accuracy problems by employing users' social network information and rating records. Ye *et al*., (2012) proposed a PLSA-like probabilistic generative model that unifies the ideas of social influence and CF-based and content-based methods for items recommendation. However, MF models for recommendation have two main disadvantages: first, the learned latent space is not easily interpreted; second, MF can only exploit observed ratings, while unrated items or unobserved ratings that may reflect users' interests cannot be exploited. As a result, the predictive performance is significantly impact.

To address the existing problems in the MF approach, Wang *et al*., (2011) developed the collaborative topic regression (CTR) model to recommend articles in an online community. CTR combines the conventional CF model with probabilistic topic modeling and generates recommendations based on both item content and other users' ratings. Based on CTR, CTRSMF (Purushotham, Liu, & Kuo, 2012) integrates CTR with social MF models to build a recommender system. Leveraging MF techniques, CTR can handle both explicit ratings and implicit ratings and produce more accurate recommendation. However, CTRSMF does not reveal the underlying relations among users. Different from CTRSMF, LACTR (Kang, & Lerman, 2013) directly learns the amount of attention that users allocate to other users and leverages this learned influence to make recommendations. LACTR can implicitly present a strong condition that users' social interactions usually follow topically similar contents. However, LACTR is sensitive to different datasets, which may not always be accurate.

Recently, deep neural network (DNN) models show great potential for in image processing, natural language processing and recommender systems (Liu *et al*., 2017). DNN are distinguished from the more commonplace single-hidden-layer neural networks by their depth. In DNN models, features are learned in a supervised, unsupervised or semi-supervised methods. Although they are more appealing than shallow models in that the features can be learned automatically, they are inferior to shallow models, such as CF, in capturing and learning the similarity and implicit relationship between items. Therefore, CF models should be integrated with DNN for higher performance. However, only a few models have been developed for CF, such as using restricted Boltzmann machines (RBM) instead of the conventional MF for rating prediction (Georgiev, & Nakov, 2013), employing multi-layer perceptron (MLP) (He *et al*., 2017), stacked denoising autoencoder (SDAE) (Wang, Wang, & Yeung, 2015), deep belief network (DBN) (Abdel-Zaher, & Eldeib, 2016), convolutional neural network (CNN) (Zheng, Noroozi & Yu, 2017), recurrent neural network (RNN) (Wu *et al*., 2017), or the combination of any arbitrary two of them (Zhang, Yao, & Sun, 2017) for learning features. Although these methods involve both DNN and CF, and boost prediction performance, the DNN and CF parts are loosely coupled without exploiting the interaction between rating data and trust information. Besides, there models are modeling without the noise, which means they will perform poorly when user ratings are mixed with significant noise.

To tackle these problems in CTR and its extensions for e-learning recommendation, this paper presents a trust-aware neural CF model, which couples DNN for both item content and side information, and CTR for both user ratings and social trust. A parameter learning method is proposed to infer latent factors both for users and items in our model. The remainder of this paper is arranged as follows: In section 2, an overview of related studies on recommendation systems and CTR models is provided. In section 3, our model is presented, and the parameters learning process is discussed. Experimental results and discussion are presented in section 4 followed by the study's conclusions and future work in section 5.

Related Studies

CF-based recommendation models

CF has been successfully applied in recommender systems for e-learning, it has been a controversial topic for a decade. Two primary approaches exist in CF methods: the neighborhood-based models (NBM) and latent factor models (LFM). The NBM is dependent on the availability of explicit information (such as user ratings) and product recommendations to a target user based on the relationships among their active neighbors without relying on information regarding the items other than their ratings (Koren, Bell, & Volinsky, 2009). The NBM-based CF has an advantage in situations in which analyzing different aspects of the data (such as music, videos and other digital products) or services is difficult. However, the user-item rating matrix is sparse in most cases, which indicates that most users vote for only a few items. Thus, the problem of data sparsity yields unsatisfactory performance of NBM-based CF.

Conversely, LFM utilizes additional information to alleviate the data sparsity problem in NBM-based CF. LFM transforms both items and users into the same latent factor space and later characterizes each entity with a feature vector inferred from the existing ratings. The most popular LFM approaches focus on taking advantage of social network information, especially social trust, to increase the accuracy of conventional CF. The social trust information between two friends can be established based on their voting or following behaviors. Different researchers have explored social networks and social trust information differently. Ma *et al*., (Ma, Yang, Lyu, & King, 2008; Ma, Zhou, Liu, Lyu, & King, 2009; Ma, Zhou, Liu, Lyu, & King, 2011) have proposed different methods for integrating social information with a MF process: SoRec, social trust ensemble (STE) and social regularization (SR). In SoRec (Ma, Yang, Lyu, & King, 2008), a user-item rating matrix and a user-user social matrix are simultaneously factorized using shared user latent factors. STE (Ma, Zhou, Liu, Lyu, & King, 2009) combines LFM with a global ratings offset and a weighted sum of the predicted ratings from trust friends for rating prediction. The SR model (Ma, Zhou, Liu, Lyu, & King, 2011) addresses the transitivity of trust in social networks and exploits the social circles and users' latent factors to create a term to regularize the MF process (Zhang, Chen, & Yin, 2013). All three models achieve better prediction accuracy than the original MF. Several methods predict user rating by traversing users' neighborhood and querying the item ratings of their direct and indirect friends, such as MoleTrust (Massa, & Avesani, 2007) and TrustWalker (Jamali & Ester 2009).

Collaborative topic regression models

Collaborative Topic Regression (CTR) utilizes item content to enhance CF methods and has achieved promising performance by integrating both user rating and item content. The CTR model combines the merits of both probabilistic MF and topic modeling approaches. This section restates the related approaches by constructing the CTR model.

(1) Probabilistic matrix factorization

In MF, users and items are both represented as latent vectors in the shared latent K-dimensional space $\mathbf{R}^K$, where user i is represented as a latent vector $u_i \in \mathbf{R}^K$ and item j is represented as a latent vector $v_j \in \mathbf{R}^K$. The prediction of whether user i will like item j is given by the inner product between their latent representations; $r_{ij} = u_i^T v_j$. To employ MF for CF methods, the latent representations of the users and items must be learned given an observed ratings matrix. A common method is to minimize the regularized squared error loss with respect to user factors $U=\{u_i|i=1, 2, \ldots, I\}$ and item factors $V=\{v_j|j = 1, 2, \ldots, J\}$, as shown in equation (1).

$$\min \sum \left(r_{ij} - u_i^T v_j \right) + \lambda_u \left\| u_i \right\|^2 + \lambda_v \left\| v_j \right\|^2 \tag{1}$$

where λ_u and λ_v are the regularization parameters. MF can be generalized as a probabilistic model by placing a zero-mean spherical Gaussian prior on both latent factors of users and items, which can be described as the following generative process:

I. For each user i, draw user latent vector

$u_i \sim N(0, \lambda_u^{-1} I_K)$

II. For each item j, draw item latent vector

$v_j \sim N(0, \lambda_v^{-1} I_K)$

III.For each user-item pair (i,j), draw the rating

$r_{ij} \sim N(u_i^T v_j, c_{ij}^{-1})$

where c_{ij} is a confidence parameter for rating r_{ij}. If c_{ij} is large, r_{ij} is trusted. Generally, c_{ij}=a if r_{ij}=1, and c_{ij}=b if r_{ij}=0. a and b are tuning parameters that satisfy a>b>0. Therefore, the probabilistic matrix factorization (PMF) can address unobserved ratings.

(2) Latent Dirichl*et al*location

The topic models provide an interpretable low-dimensional representation of the documents. In this section, we exploit the discovered topic structure by Latent Dirichl*et al*location (LDA) for item recommendation. Assume that the fixed vocabulary W, which is referred to as a set of tags to annotate items in this paper. Assume K topics $\varphi=\varphi_{1:K}$, each of which is a distribution over the set of tags. The generative process of LDA is as follows:

I. For each item j, draw the topic proportions

$\theta_j \sim$ Dirichlet(α);

II. For each word/tag w_{jn},

(i) draw the topic assignment $z_{jn} \sim$ Mult(θ_j);

(ii) draw the word/tag $w_{jn} \sim$ Mult($\varphi_{z_{jn}}$).

For the parameters estimation of LDA, we can choose variational inference or Gibbs sampling. The learned topic proportions θ_j are item-specific, whereas the set of topics φ is shared by all items.

(3) Collaborative topic regression

CTR represents users with topical interests and assumes that items are produced by a topic model, as shown in Figure 1. In addition, CTR includes a latent variable ε_j, which can offset the topic proportions θ_j when modeling the user ratings. The offset variable ε_j can capture the item preference for a particular user considering their ratings.

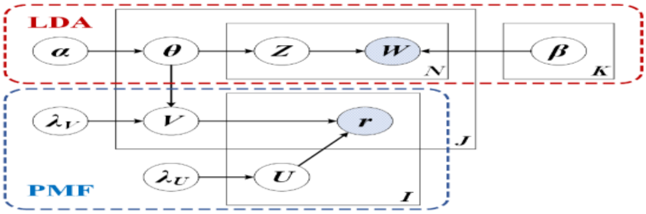

Figure 1. Collaborative topic regression model

Assume K topics $\beta=\beta_{1:K}$; the generative process of CTR model is shown as follows:

I. For each user i, draw user latent vector

$u_i \sim N(0, \lambda_u^{-1}I_K)$

II. For each item j;

(i) Draw the topic proportions $\theta_j \sim$ Dirichlet(α)

(ii) Draw item latent offset $\varepsilon_j \sim N(0, \lambda_v^{-1}I_K)$ and set the item latent vector as $v_j = \varepsilon_j + \theta_j$

(iii)For each word/tag w_{jn},

a. Draw the topic assignment $z_{jn} \sim$ Mult(θ_j)

b. Draw the word/tag $w_{jn} \sim$ Mult(β_{zjn})

III. For each user-item pair (i,j), draw the rating

$r_{ij} \sim N(u_i^T v_j, c_{ij}^{-1})$

CTR successfully uses item content information for recommendation. However, this model does not exploit social information and cannot reliably learn the user latent space for new users. To address this issue, some approaches have been proposed using different variants that incorporate social information into CTR. For instance, CTRSMF (Purushotham, Liu, & Kuo, 2012) integrated CTR with social matrix factorization models using a strategy that is similar to SoRec. To consider the social correlation among users, the social matrix is simultaneously factorized with the rating matrix. Different from CTRSMF, LACTR (Kang & Lerman, 2013) directly learns the amount of attention that users allocate to other users and leverages this learned influence to make recommendations. In addition, Wang *et al*., (2015) developed a novel hierarchical Bayesian model RCTR, which extends CTR by seamlessly integrating the user rating, item content, and network structure among items into the same model. Wu *et al*., (2016) proposed an effective CTR model that combines CTR with social trust ensemble, topic modeling and probabilistic matrix factorization. Although these researchers have improved CTR in separate aspects, a critical problem remains, i.e., the effective integration of social information into the CTR model

Stacked denoising autoencoders

SDAE is a deep neural network that is stacked by multiple denoising autoencoders (DAE). Each layer of SDAE is trained as a DAE by minimizing the error in reconstructing its input (which is the output of the previous layer). Usually we consider the first half layers of the network as an encoding part and the last half layers as a decoding part. Encoding part tries to learn the feature representations of the noise-corrupted input, and decoding part tries to reconstruct the clean input itself in the output. An example of 2-layer SDAE with the number of layer L=4 is shown in Figure 2. X_0 and X_c are the corrupted input and clean input respectively, the hidden layers are in the middle, and the output of the l–th layer is denoted by X_l. Generally, given a set of input vectors as raw content information of all items, an L/2-layer SDAE solves the optimization problem in the equation (2).

$$\min_{\{W_l\},\{b_l\}} \|X_c - X_L\|_F^2 + \lambda \cdot \sum_l \left(\|W_l\|_F^2 + \|b_l\|_2^2\right) \tag{2}$$

where X_L denotes the output of layer L of the neural network, W_l and b_l denote the weight matrix and bias vector of layer l of the network respectively, λ is a regularization hyperparameter and $\|\cdot\|_F$ denotes the Frobenius norm. Once the model is trained, item content features could be obtained from the hidden layer $X_{L/2}$ of the network. Here, an L/2-layer SDAE corresponds to an L-layer network. And the feature representation of a given item i is a low-dimensional vector.

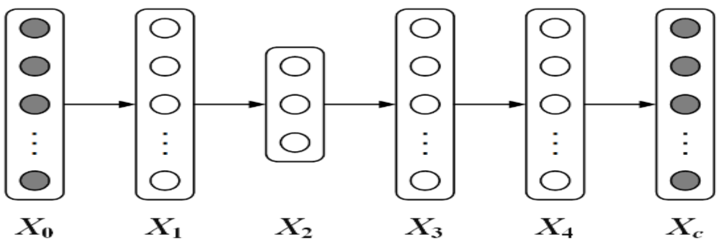

Figure 2. A 2-layer SDAE with L=4

It is noted that apart from the goal of learning the features from the rating records, another goal of using SDAE is to reduce the dimensionality of the item content-based vectors to be same with latent factor vectors, which can then be fused into the CF process.

Trust-aware Neural Collaborative Topic Regression

Our proposed model

CTR models is a probabilistic graphical model that usually works very well for products/services that fall easily into established categories, and provides precise and interpretable recommendations. However, its representation learning process is often not effective enough especially when content information is very sparse, especially cold-start situations, such as new products like non-sequels movies or new users without insufficient profiles. In social networks, users can be easily influenced by their trusted friends and prefer the recommendations of these friends, according to conformist mentality. Users have individual tastes, preferences, and methods for rating behaviors due to their personalities and characteristics. Therefore, users' decisions about items rating in social networks are balanced between their preferences and their trusted friends' favors. As a result, a new neural collaborative topic regression combined with social trust (NCTRS) is proposed to accurately reflect this observation in the e-learning recommender systems, as shown in Figure 3. NCTRS integrates SDAE into PMF, which tightly combines perception component (DNN) and task-specific component (PMF). Specifically, the task-specific component introduces the social trust relationship into the rating prediction to balance the influences of between users' personal tastes and the favors of their trusted friends. The rating prediction process of NCTRS is described as follows.

It is assumed that $U=\{u_i|i=1, 2,\ldots, I\}$ is a set of I users and $V=\{v_j|j=1, 2,\ldots, J\}$ is a set of J distinct items (e-learning resources) in social network G=(U,E). Based on matrix factorization methods, users U and items V can also be represented as latent vectors in the shared latent K-dimensional space R^K. The user ratings set is denoted by a I×J matrix $R=\{r_{ij}|i=1, 2,\ldots, I; j=1, 2,\ldots, J\}$, where $r_{ij}=1$ if user i rated item j; otherwise, $r_{ij}=0$. The content information of items is characterized by a J×D matrix X_c, where row j is the bag-of-words vector $X_{c,j*}$ for item j based on a vocabulary of size D. And, X_c is the clean input to the SDAE while the noise corrupted matrix X_0 that is also a J×D matrix. The output of layer l of the SDAE is denoted by X_l which is a J×K_l matrix. Similar to X_c, row j of X_l is denoted by $X_{l,j}$. W_l and b_l are the weight matrix and bias vector of layer l respectively, $W_{l,n}$ denotes column n of W_l, L is the number of layers, and the set of weight matrices and biases of all layers is denote by Ω.

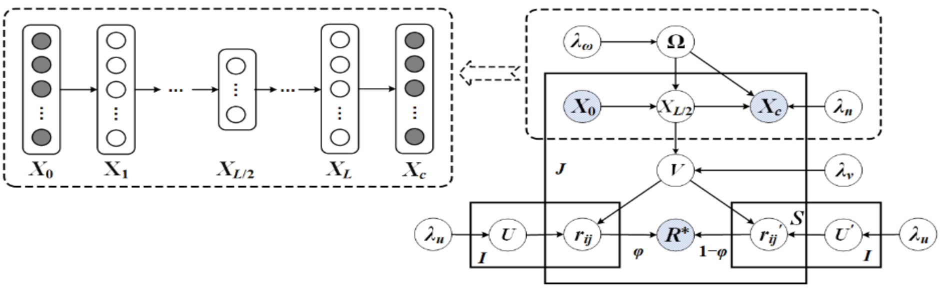

Figure 3. The framework of Neural Collaborative Topic Regression

In addition, the social trust matrices are marked with $S=\{S_i|i=1, 2,\ldots, I\}$, where U' is the set of users who U directly trusts. The trade-off between user ratings and the influence of the social trust is determined by parameter $\varphi \in[0,1]$, which fuses the moderate amount of real-world recommendation processes into the recommender systems. The parameter φ controls how much users trust their friends. Apparently, the influence of social trust is disregarded as φ=1, and φ=0 assigns the highest possible weight to social trust.

After the assumptions, the next step is to predict ratings based on users' and items' latent features, defined as follows:

I. For each layer l of the SDAE,

(i) For each column n of matrix W_l, draw

$W_{l,n} \sim N(0, \lambda_\omega^{-1} \mathrm{I}_{K_l})$.

$b_l \sim N(0, \lambda_\omega^{-1} \mathrm{I}_{K_l})$.

(ii) Draw the bias vector.

(iii)For each row j of X_l, draw

$X_{l,j} \sim N(\sigma(X_{l-1,j} \cdot W_l + b_l), \lambda_d^{-1} \mathrm{I}_{K_l})$.

2. For each item v_j,

(i) Draw clean input $X_{c,j} \sim N(X_{L,j}, \lambda_n^{-1} \mathrm{I}_J)$.

(ii) Draw item latent offset $\varepsilon_j \sim N(0, \lambda_v^{-1} \mathrm{I}_K)$, set item latent vector as $v_j = \varepsilon_j + X_{L/2,j}^T$.

3. Draw a latent user vector for each user

$u_i \sim N(0, \lambda_u^{-1} \mathrm{I}_K)$.

4. For each user-item pair (i,j), draw the rating

$r_{ij} \sim N(u_i^T v_j, C_{ij}^{-1})$.

where W_l and b_l are the weight matrix and biases vector for layer l; I_K is a K-dimensional identity matrix of layer l; λ_u, λ_v, λ_d, λ_n, λ_ω are regularization hyperparameters; C_{ij} is a confidence parameter similar to that for CTR, which is to measure the confidence to observations.

Then, the influence of social trust S(u) is introduced, and rating can be computed by

$$R^* = \varphi u_i^T v_j + (1-\varphi) \sum_{u_k \in U'} S(u_k) u_k^T v_j \tag{3}$$

The influence S(u) is the normalization trustworthiness in the social network G by propagating users' positive and negative opinions. Given two users, u_i and u_j in G, p_{ij} is defined as a real valued attribute that represents the trust from u_i to u_j. If p_{ij} is positive, u_i is a follower of u_j. In other cases, u_i would be a detractor of u_j. A propagation algorithm (Cruz *et al.*, 2012) is introduced to propagate positive and negative information via a network. The trust scores S can be obtained, as shown in equations (4) ~ (6).

$$\begin{aligned} S^+(u_i) &= (1-\psi) e_i^+ + \psi \sum_{j \in In^+(u_i)} \frac{p_{ji}}{\sum_{j \in Out(u_i)} |p_{jk}|} S^+(u_j) \\ &+ \psi \sum_{j \in In^-(u_i)} \frac{-p_{ij}}{\sum_{j \in Out(u_j)} |p_{jk}|} S^-(u_j) \end{aligned} \tag{4}$$

$$S^-(u_i) = (1-\psi)e_i^- + \psi \sum_{j \in In^+(u_i)} \frac{p_{ji}}{\sum_{j \in Out(u_i)} |p_{jk}|} S^-(u_j) + \psi \sum_{j \in In^-(u_i)} \frac{-p_{ij}}{\sum_{j \in Out(u_j)} |p_{jk}|} S^+(u_j) \tag{5}$$

$$S(u_i) = \frac{S^+(u_j) - S^-(u_j)}{S^+(u_j) + S^-(u_j)} \tag{6}$$

where S^+ and S^- are a user's trust degree and distrust degree, respectively, and $S \in [-1,1]$, where -1 and 1 represent a totally untrustworthy user and a totally trustworthy user, respectively. ψ is the damping factor that represents the probability of choosing a neighbor of the current user in the next step of the random walk. Both e^+ and e^- are the personalization vectors, which are intended to compute a biased ranking algorithm in which certain users have a higher probability of being trusted than others. Therefore, equation (3) can be rewritten using matrix notation by equation (7).

$$R^* = \left[\varphi \mathbf{I} + (1-\varphi) S\right] U^T V \tag{7}$$

where I is an identity matrix. This equation employs users rating, items content and the social trust information for rating prediction.

Compared with other CTR-based models (Purushotham er al., 2012; Kang *et al.*, 2013; Wu *et al.*, 2016), NCTRS can reveal the trust relations among the users, reflect the transitivity of trust and distrust in a social network, and provide an intuitive explanation of predictions. To exploit the principle of social trust in the NCTRS, the conditional distribution of observed ratings is represented by a Gaussian $N(\varphi u_i^T v_j + (1-\varphi)\Sigma S u_k^T v_j, C_{ij}^{-1})$ instead of $N(u_i^T v_j, C_{ij}^{-1})$, and the probability of full ratings R when given U, V, φ and S is assumed to be factorial, as shown in equation (8).

$$p(R|U,V,\varphi,S) = \prod_{i=1}^{I} \prod_{j=1}^{J} N\left(r_{ij} \middle| \varphi u_i^T v_j + (1-\varphi) \sum_{u_k \in U'} S \cdot u_k^T v_j,\ C_{ij}^{-1} \right) \tag{8}$$

Similar to the CTR model, the complete model of NCTRS is expressed as follows:

$$\begin{aligned} & p(R,U,V,X_l,\Omega | \lambda_u, \lambda_v, \lambda_n, \lambda_d, \lambda_\omega, \varphi, S) \\ & \propto p(R|U,V,\varphi,S) \times p(U|\lambda_u) \times p(V|\lambda_v, X_{L/2}) \\ & \times p(X_c | \lambda_n, X_L) \times p(X_l | \lambda_{l-1}, \lambda_d, W_l) \times p(\Omega | \lambda_\omega) \end{aligned} \tag{9}$$

$$p(U|\lambda_u) = \prod_{i=1}^{I} N(0, \lambda_u^{-1} \mathbf{I}_K) \tag{10}$$

$$p\left(V \middle| \lambda_v, X_{L/2}\right) = \prod_{j=1}^{J} N\left(v_j \middle| X_{L/2},\ \lambda_v^{-1}\mathbf{I}_K\right) \tag{11}$$

$$p\left(X_c \middle| \lambda_n, X_L\right) = \prod_{j=1}^{J} N\left(X_{L,j}, \lambda_n^{-1}\mathbf{I}_J\right) \tag{12}$$

$$\begin{aligned} & p\left(X_l \middle| \lambda_{l-1}, \lambda_d, W_l\right) \\ & = \prod_{j=1}^{J} N\left(\sigma\left(X_{l-1,j}W_l + b_l\right), \lambda_d^{-1}\mathbf{I}_{K_l}\right) \end{aligned} \tag{13}$$

$$p\left(\Omega \middle| \lambda_\omega\right) = \prod_{l=1}^{L} N\left(0, \lambda_\omega^{-1}\mathbf{I}_{K_l}\right) \tag{14}$$

where $p(U|\lambda_u)$, $p(V|\lambda_v, X_{L/2})$ and $p(\Omega|\lambda_\omega)$ are derived by placing Gaussian prior on users, items and weight matrix; $p(X_c|\lambda_u, X_L)$, $p(X_l|\lambda_{l-1}, \lambda_d, W_l)$ is derived by placing corresponding conditional probability based on assumption of Gaussian distribution. Similar to SDAE, if λ_d goes to infinity, the maximization of posterior probability is equivalent to maximizing the joint log-likelihood of U, V, X_c, X_l, Ω.

Learning parameters

To learn the parameters of NCTRS, an EM-style algorithm is developed to compute the maximum a posteriori (MAP) estimates, which is based on the similar process of CTR (Wang *et al*., 2011) and CDL (Wang *et al*., 2015). MAP estimation is equivalent to maximizing the joint log-likelihood of U, V, X_c, X_l and Ω, when given λ_u, λ_v, λ_n, φ and S, as follows.

$$\begin{aligned} \mathcal{L} = & -\frac{\lambda_u}{2} \cdot \sum_i \left\| u_i \right\|_2^2 - \frac{\lambda_v}{2} \cdot \sum_j \left\| v_j - X_{L/2,j}^{2} \right\|_2^2 \\ & -\frac{\lambda_n}{2} \cdot \sum_i \left\| X_{L,j} - X_{c,j} \right\|_2^2 - \frac{\lambda_\omega}{2} \cdot \sum_l \left(\left\| W_l \right\|_F^2 + \left\| b_l \right\|_2^2 \right) \\ & -\frac{\lambda_d}{2} \cdot \sum_l \sum_j \left\| \sigma\left(X_{l-1,j}W_l + b_l\right) - X_{l,j} \right\|_2^2 \\ & -\sum_i \sum_j \frac{C_{ij}}{2} \left(R_{ij} - R_{ij}^*\right) \end{aligned} \tag{15}$$

The objective function $\mathcal{L}$ is optimized by using a coordinate ascent approach by iteratively optimizing the CTR model, social network variables $\{u_i, v_j\}$ and the given weight matrix Ω. For $\{u_i, v_j\}$, the process of maximization is similar to matrix factorization. Given a current estimate of Ω, we calculate the gradient of $\mathcal{L}$ with respect to $\{u_i, v_j\}$ and set it to zero to determine $\{u_i, v_j\}$ in terms of U, V, R, λ_u, λ_v and the confidence matrix C that corresponds to R. Derived update equations are shown as follows:

$$u_i \leftarrow \left(\varphi^2 V C_i V^T + (1-\varphi)^2 S^2 V C_i V^T + \lambda_u \mathbf{I}_K\right)^{-1} \times \left(\varphi^2 V C_i R_i + (1-\varphi)^2 S^2 V C_i R_i\right) \tag{16}$$

$$v_j \leftarrow \left(\varphi^2 U C_j U^T + (1-\varphi)^2 S^2 U C_j U^T + \lambda_v \mathbf{I}_K\right)^{-1} \times \left(\varphi U C_j R_j + (1-\varphi) S U C_j R_j + \lambda_v X_{L/2,j}^2\right) \tag{17}$$

Given the updated variables $\{u_i, v_j\}$, weights matrix W_l and biases vector b_l for layer l are updated using the back-propagation learning algorithm, a local optimum for objective function L can be found. The gradients of the likelihood with respect to W_l and b_l are as follows:

$$\nabla_{W_l} \mathcal{L} = -\lambda_\omega W_l - \lambda_v \sum_j \nabla_{W_l} X_{L/2,j}^T \left(X_{L/2,j}^T - v_j\right) - \lambda_n \sum_j \nabla_{W_l} X_{L/2,j} \left(X_{L/2,j} - X_{c,j}\right) \tag{18}$$

$$\nabla_{b_l} \mathcal{L} = -\lambda_\omega b_l - \lambda_v \sum_j \nabla_{b_l} X_{L/2,j}^T \left(X_{L/2,j}^T - v_j\right) - \lambda_n \sum_j \nabla_{b_l} X_{L/2,j} \left(X_{L/2,j} - X_{c,j}\right) \tag{19}$$

Rating Prediction

After the optimal parameters are learned, the NCTRS model can be employed for in-matrix (non cold-start) and out-matrix (cold-start) prediction. Assume that the OBD is the observed test data. Both types of predictions can be estimated in following equations.

$$E\left[r_{ij} \mid OBD\right] \approx E\left[\varphi u_i + (1-\varphi) S u_i \mid OBD\right]^T \times \left(E\left[X_{L/2,j}^T \mid OBD\right] + E\left[\varepsilon_j \mid OBD\right]\right) \tag{20}$$

$$R_{ij}^* \approx \left[\varphi u_i + (1-\varphi) S u_i\right]^T \left(X_{L/2,j}^T + \varepsilon_j\right) = \varphi u_i^T v_j + (1-\varphi) S u_i^T v_j \tag{21}$$

For item-based cold-start prediction, the item is new and has not been rated by other users. Thus, $E[\varepsilon_j]=0$, and NCTRS predicts the rating as follows:

$$R_{ij}^* \approx u_i^T X_{L/2,j}^T \tag{22}$$

For user-based cold-start prediction, the user is new and has not rated any item, i.e. $E[u_i]=0$, and NCTRS predicts the rating as follows,

$$R_{ij}^* \approx S u_k^T v_j \tag{23}$$

Experiments and Results

Datasets

In this section, Douban and Epinions datasets are collected to evaluate NCTRS. The Douban dataset contains a social friend network, whereas the Epinions dataset has a trust network. Douban is the largest online Chinese language book, movie and music database and is one of the largest online learning communities in China. Users can assign a five-scale integer ratings (from one to five) to movies, books and music. Users on Douban can join different interesting groups, which exceed 700 in the "Movie" subcategory. The Douban dataset contains 129,490 users and 58,541 movies with 16,830,839 movie ratings, and a total of 1,692,952 claimed social relationships in the social friend network. The statistics of the Douban user rating matrix and social network are summarized in Table 1.

Table 1
Statistics of Douban Dataset

Statistics	User	Item	Statistics	Friends/User
Max. Rating	6,328	49,504	Max. Num.	986
Avg. Rating	129.98	287.51	Avg. Num.	13.07

Table 2
Statistics of Epinions Dataset

Statistics	User	Item
Max. Rating Num.	1,960	7,082
Avg. Rating Num.	12.21	7.56
Statistics	Trust/User	Trusted/User
Max. Num.	1763	2443
Avg. Num.	9.91	13.73

Epinions is a well-known consumer review site that was established in 1999. At Epinions, visitors can read reviews regarding a variety of items to help them make a decision about a purchase or they can join for free and begin writing reviews that may earn them reward and recognition. To post a review, members need to rate the product or service on a rating scale from one star to five stars. Every member of Epinions maintains a trust list that presents a network of trust relationships among users and a distrust list that presents a network of distrust relationships. The Epinions dataset consists of 51,670 users who rated 83,509 different items. The total number of ratings is 631,064. In the case of the social trust network in the Epinions dataset, the total number of issued trust statements is 511,799. The statistics of this dataset are listed in Table 2.

For the Douban and Epinions datasets, we employed 90% and 80% of data as the training datasets and 10% and 20% of data as the test set.

Evaluation metrics and baselines

In this paper, two popular metrics: the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE), are employed to measure the prediction quality of our proposed approach compared with other recommendation methods. MAE and RMSE are defined as follows:

$$MAE = \frac{1}{N}\sum_{i,j}\left|R_{ij} - R_{ij}^{*}\right| \tag{24}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i,j}\left(R_{ij} - R_{ij}^{*}\right)^{2}} \tag{25}$$

where N is the total number of predictions, R_{ij} is the real rating and R_{ij}^{*} is its corresponding predicted rating.

To demonstrate the effectiveness of our NCTRS, we compare the recommendation results of four state-of-art CTR-based recommendation methods, listed as follows:

(1) CTR: CTR is a point-wise algorithm that combines LDA and PMF as mentioned in the previous section (Wang *et al*., 2011).

(2) CTRSMF: CTRSMF integrates CTR with social matrix factorization, which is incorporates ratings, item contents and social ensemble (Purushotham, Liu, & Kuo, 2012).

(3) LACTR: LACTR is a sophisticated CTR model which makes recommendation based on users' limited attention, ratings, items content and social network (Kang, & Lerman, 2013).

(4) CDL: CDL is a point-wise hierarchical Bayesian model, which seamlessly couples deep representation feature of items content and CTR (Wang, Wang, & Yeung, 2015).

Experimental settings

Table 3
Impact of Number of Layers at MAE @ 90% on Douban and Epinions

	L=2	L=4	L=6	L=8	L=10
Douban	0.5789	0.5631	0.5542	0.5496	0.5517
Epinions	0.8436	0.8318	0.8235	0.8191	0.8184

Table 4
Impact of Number of Layers at MAE @ 80% on Douban and Epinions

	L=2	L=4	L=6	L=8	L=10
Douban	0.5798	0.5626	0.5573	0.**5527**	0.5535
Epinions	0.8581	0.8462	0.8389	0.8324	0.8316

In the experiments, we use a validation set to find the optimal hyperparameters for CTR, CTRSMF, LACTR and CDL. For CTR, α=1, β=0.01, λ_u=0.01 on Douban, λ_u=0.1 on Epinions, and λ_v=10. For CTRSMF and LACTR, their settings of parameters α, β, λ_u and λ_v are same as CTR, λ_q=0.01, λ_s= 0.01, and λ_φ=1. For CDL and NCTRS, we set α=1, β=0.01, φ=0.5 and perform grid search on the hyperparameters λ_u, λ_v and λ_n. The SDAE part employs a mixture of edge detectors and masking noise with a noise level of 20 % to obtain the corrupted input X_0 from the clean input X_c. Meanwhile, dropout rate is set to 0.1 for achieving adaptive regularization when the number of layers is more than 2. The number of hidden units K_l is set to 1000, while the number of middle layer is 200. Note that K_0 and K_L are same as the size of vocabulary. After searching, it is found that the hyperparameters setting: λ_u = 0.01, λ_v=10, λ_n=100, and λ_ω=0.001 can achieve good performance. Finally, for convenience, we examine the impact of hidden layers number L to MAE, and the results are shown in Tables 3 and 4.

Table 3 shows the value of MAE decreases as the number of layers increases on Epinions @ 90%. On Douban @ 90%, NCTRS begins to overfit when L>8.

From Table 4, we can find that the trend of MAE value is similar @ 80% on Douban and Epinions with different settings of L. That means, NCTRS model can guide the further features learning with increasing of train set size. Thus, we set L=8 on Douban, L=10 for on Epinions for both CDL and NCTRS.

Experimental settings

The experimental results for two datasets are listed in Tables 5 and 6. The percentages in two tables represent the improvements of our method over the corresponding approaches.

From Table 5, we notice that CTRSMF, LACTR, CDL and NCTRS work better than CTR on two datasets in the 90% training data setting. By introducing social trust into the CTR model, NCTRS significantly improves the prediction accuracy. The MAE values of NCTRS is 3.03%, 1.87% and 1.08% lower than that of CTRSMF, LACTR and CDL respectively, and NCTRS outperforms other three methods in term of RMSE. The MAE and RMSE values generated by all methods on the Epinions dataset are higher than the MAE and RMSE values generated by all methods on the Douban dataset. This finding indicates that the trust network in Epinions probably contains significant noise. Because NCTRS treats users' trusted users differently, we can reduce a number of the errors that are caused by this noise.

From Table 6, we observe that the proposed model outperforms other three approaches in the setting of 80% on both datasets, again. Generally, NCTRS improves the baselines by a minimum of 1.02% and 1.08% on Douban and Epinions in terms of MAE, and it improves the RMSE value by a minimum of 1.06% and 1.11% on two datasets. Therefore, NCTRS can achieve better performance than other four CTR-based models.

Table 5

Performance Comparisons @ 90% on Douban and Epinions Datasets

Dataset	Metrics	CTR	CTRSMF	LACTR	CDL	NCTRS
Douban	MAE	0.5802	0.5668	0.5601	0.5556	**0.5496**
	IMP	5.27%	3.03%	1.87%	1.08%	
	RMSE	0.7259	0.709	0.7004	0.6947	**0.6869**
	IMP	5.37%	3.12%	1.93%	1.12%	
Epinions	MAE	0.8835	0.8618	0.8377	0.8279	**0.8184**
	IMP	7.37%	5.04%	2.30%	1.15%	
	RMSE	1.1533	1.1251	1.0934	1.0806	**1.0677**
	IMP	7.42%	5.10%	2.35%	1.19%	

Table 6

Performance Comparisons @ 80% on Douban and Epinions Datasets

Dataset	Metrics	CTR	CTRSMF	LACTR	CDL	NCTRS
Douban	MAE	0.5839	0.5676	0.5628	0.5584	**0.5527**
	IMP	5.34%	2.63%	1.79%	1.02%	
	RMSE	0.733	0.7125	0.7062	0.7007	**0.6933**
	IMP	5.42%	2.69%	1.83%	1.06%	
Epinions	MAE	0.8943	0.8719	0.8498	0.8407	**0.8316**
	IMP	7.01%	4.62%	2.14%	1.08%	
	RMSE	1.1591	1.1302	1.1014	1.0893	**1.0772**
	IMP	7.07%	4.69%	2.20%	1.11%	

Conclusion

In this paper, a novel trust-aware neural CF model (NCTRS) is proposed for social e-learning recommendation. NCTRS employs SDAE to extract deep feature representation from side information and then introduces social trust into rating prediction. NCTRS can integrate item content, user rating, and trust information into the CTR-based recommendation process. The results on different datasets indicate that NCTRS can generate better predictions than other CTR-based models. The experimental results reveal that exploiting social network information in the NCTRS model can significantly improve prediction accuracy, and users' preferences have greater influence on their decisions regarding rating items than do their social circles in social networks.

In the future, we plan to use other deep learning methods to replace SDAE for boosting further performance in our model, and examine advanced measures that qualify social trust between users in a social network to investigate recommendation effectiveness.

References

Abdel-Zaher, A. M., & Eldeib, A. M. (2016). Breast cancer classification using deep belief networks. *Expert Systems with Applications*, *46*, 139-44. http://dx.doi.org/10.1016/j.eswa.2015.10.015

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender Systems Survey. *Knowledge-Based Systems, 46*(1), 109-32. http://dx.doi.org/10.1016/j.knosys.2013.03.012

Cela, K. L., Sicilia, M. Á., & Sánchez, S. (2015). Social network analysis in e-learning environments: A preliminary systematic review. *Educational Psychology Review, 27*(1), 219-246. http://dx.doi.org/10.1007/s10648-014-9276-0

Cruz, F. L., Vallejo, C. G., Enri, F., & Troyano, J. A. (2012). Polarityrank: Finding an equilibrium between followers and contraries in a network. *Information Processing & Management*, *48*(2), 271-82. http://dx.doi.org/10.1016/j.ipm.2011.08.003

Georgiev, K., & Nakov, P. (2013). A non-IID framework for collaborative filtering with restricted Boltzmann machines. *Proceedings of the 30th International Conference on Machine Learning*, *2013*, 1148-1156.

Groh, G., Birnkammerer, S., & Köllhofer, V. (2012). Social recommender systems. *Recommender Systems for the Social Web.* Berlin: Springer Press, *32*.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. *Proceedings of the 26th International Conference on World Wide Web*, 173-82. http://dx.doi.org/10.1145/3038912.3052569

Hu, L. J., Tang, L., Pan, Q., Song, H., & Wen, P. G. (2016). Research and analysis of PI control strategy based on neural network in power grid. *Mathematical Modelling of Engineering Problems, 3*(1), 25-28.

Jamali, M., & Ester, M. (2009). Trust walker: A random walk model for combining trust-based and item-based recommendation. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 397-410. http://dx.doi.org/10.1145/1557019.1557067

Kang, J. H., & Lerman, K. (2013). LA-CTR: A limited attention collaborative topic regression for social media. *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, 1128-1134.

Klašnja-Milićević, A., Ivanović, M., & Nanopoulos, A. (2015). Recommender systems in e-learning environments: A survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review, 44*(4), 571-604. http://dx.doi.org/10.1007/s10462-015-9440-z

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer, 42*(8), 30-37.

Koren, Y., & Bell, R. (2011). *Advances in collaborative filtering. Recommender Systems Handbook*. New York: Springer Press.

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing, 234*, 11-26.

Ma, H., King, I., & Lyu, M. R. (2009). Learning to recommend with social trust ensemble. *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2009*, 203-210.

Ma, H., Yang, H., Lyu, M. R., & King, I. (2008). Sorec: Social recommendation using probabilistic matrix factorization. *Proceedings of 17th ACM Conference on Information and Knowledge Management*, 931-40. http://dx.doi.org/10.1145/1458082.1458205

Ma, H., Zhou, D., Liu, C., Lyu, M. R., & King, I. (2011). Recommender systems with social regularization. *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 287-296. http://dx.doi.org/10.1145/1935826.1935877

Massa, P., & Avesani, P. (2007). Trust metrics on controversial users: Balancing between tyranny of the majority. *International Journal on Semantic Web & Information Systems*, *3*(1), 39-64.

Mothe, J., & Rakotonirina, A.J. (2018). Contextual collaborative filtering. A LDA-based approach, *Ingenierie des Systemes d'Information*, *23(1)*, 89-109. http://dx.doi.org/10.3166/ISI.23.1.89-109

Najim, J. (2016). An introduction to large random matrix theory, *Traitement du Signal, 33*(2-3), 161-222. http://dx.doi.org/10.3166/TS.33.161-222

Purushotham, S., Liu, Y., & Kuo, C. C. J. (2012). Collaborative Topic Regression with Social Matrix Factorization for Recommendation Systems. *Proceedings of the 29th International Conference on Machine Learning*.

Song, Y., Zhang, L., & Giles, C. L. (2011). Automatic tag recommendation algorithms for social recommender systems. *ACM Transactions on the Web*, *5*(1), 1-35. http://dx.doi.org/10.1145/1921591.1921595

Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence, 12*, 1-19. http://dx.doi.org/10.1155/2009/421425

Wang, C., & Blei, D. M. (2011). Collaborative topic modeling for recommending scientific articles. *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 448-456.

Wang, H., & Li, W. (2015). Relational collaborative topic regression for recommender systems. *IEEE Transactions on Knowledge & Data Engineering, 27*(5), 1343-55.

Wang, H., Wang, N., & Yeung, D. Y. (2015). Collaborative deep learning for recommender systems. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1235-1244. http://dx.doi.org/10.1145/2783258.2783273

Wang, T. C., Xie, Y. Z., & Yan, H. (2016). Research of multi sensor information fusion technology based on extension neural network. *Mathematical Modelling of Engineering Problems, 3*(3), 129-134.

Wu, C. Y., Ahmed, A., Beutel, A., Smola, A. J., & Jing, H. (2017). Recurrent recommender networks. *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, 495-503. http://dx.doi.org/10.1145/3018661.3018689

Wu, H., Yue, K., Pei, Y., Li, B., Zhao, Y., & Dong, F. (2016). Collaborative Topic Regression with social trust ensemble for recommendation in social media systems. *Knowledge-Based Systems, 97*, 111-22. http://dx.doi.org/10.1016/j.knosys.2016.01.011

Ye, M., Liu, X., & Lee, W. (2012). Exploring social influence for recommendation: a generative model approach. *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 671-680.

Zhang, S., Yao, L., & Sun, A. (2017). Deep learning based recommender system: A survey and new perspectives. *arXiv preprint arXiv: 1707.07435*.

Zhang, Y., Chen, W., & Yin, Z. (2013). Collaborative filtering with social regularization for TV program recommendation. *Knowledge-Based System, 54*, 310-17. http://dx.doi.org/10.1016/j.knosys.2013.09.018

Zheng, L., Noroozi, V., & Yu, P. S. (2017). Joint deep modeling of users and items using reviews for recommendation. *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, 425-434.