SSL-STR: Semi-Supervised Learning for Sparse Trust Recommendation

Zhengdi Hu*, Guangquan Xu*, Xi Zheng†, Jiang Liu*, Xiaojiang Du ‡
* Tianjin Key Laboratory of Advanced Networking (TANK), College of Intelligence and Computing, Tianjin University, China
Email: hzd12@tju.edu.cn; losin@tju.edu.cn (Corresponding author); jiangliu@126.com
† Department of Computing, Macquarie University, Australia
Email: james.zheng@mq.edu.au
‡ Department of Computer and Information Sciences, Temple University, USA
Email: dxj@ieee.org

Abstract—**Trust is widely applied in recommender systems to improve recommendation performance by alleviating well-known problems, such as cold start, data sparsity, and so on. However, trust data itself also faces sparse problems. To solve these problems, we propose a novel sparse trust recommendation model, SSL-STR. Specifically, we decompose the aspects influencing trust-building into finer-grained factors, and combine these factors to mine the implicit sparse trust relationships among users by employing the Transductive Support Vector Machine algorithm. Then we extend SVD++ model with social trust and sparse trust information for rating prediction in the recommendation system. Experiments show that our SSL-STR improves the recommendation accuracy by up to 4.3%.**

Keywords—Sparse trust; SSL-STR; Transductive Support Vector Machine; SVD++; recommendation system

I. INTRODUCTION

The recommendation system plays an increasingly important role in large e-commerce websites. Collaborative filtering (CF) algorithms are widely used in recommendation systems and achieve huge success. But recommendation systems always face two major problems: cold start and evaluation data sparsity [1].

Trust is widely used in recommendation systems to improve prediction accuracy. By providing additional information about the interaction of users, trust can help us to alleviate some well-known problems, such as cold start and rating data sparsity. It is believed that users are easily influenced by the people they trust. Based on this phenomenon, a lot of work has been done to study how to improve the accuracy of recommendation systems by utilizing existing trust information effectively [2], [3]. However, there are still some problems when using trust information to improve the accuracy of recommendation systems.

As mentioned above, not only the rating data but also the trust data faces the data sparse problem [4]. It means that the system does not accumulate enough trust information for improving recommendation performance. There are many users having similar preferences and user-behavior, but their trust information is covered by big data noise.

Thus, we propose *SSL-STR*, a sparse trust recommendation model to mine sparse trust relationships more accurately and adopt sparse trust information more effectively. We will mine sparse trust based on two main aspects: user behavior and user preferences. We decompose them into four fine-grained factors: similarity, consistency, credibility, and predictability. Then we combine these factors by using Transductive Support Vector Machine (TSVM) [5]. Due to the particularity of the trust label, we adopt the semi-supervised algorithm which is more consistent with the features of trust data (detailed in Section IV). Our method can effectively reduce the sparsity of trust data. After that, we extend the SVD++ model [6] with social trust and sparse trust to improve the prediction accuracy. In addition, we use the weighted-λ-regularization technique to further avoid over-fitting for model learning.

Compared with existing trust-based recommendation models, our *SSL-STR* model has several advantages: First, we point out that social trust has limitations in helping the recommendation system. So, we consider the effect of both social trust and sparse trust on the recommendation system. Second, different from most traditional trust models, we decompose the aspects influencing trust-building into finer-grained factors. Third, we use a semi-supervised method to mine sparse trust data. It should be noted that in a very similar job, Fang et al. [7] predicted the trust between users by using the Support Vector Regression (SVR) model. However, they do not distinguish between distrust data and unlabeled trust data, but instead treat both as distrust data, which leads to a less accurate trained trust prediction model. On the contrary, we think about distrust and unlabeled trust separately, which makes the application of trust data more reasonable. The main contributions of our research are given as follows:

- The concept of sparse trust recommendation is proposed in this paper which provides a theoretical basis for the application of sparse trust.
- We divide the aspects affecting sparse trust into finer-grained factors and combine them through the semi-supervised algorithm to mine sparse trust.
- We combine the effects of social trust and sparse trust to improve the effectiveness of recommendations.

978-1-7281-0962-6/19/$31.00 ©2019 IEEE

- We extend the SVD++ model with social trust and sparse trust.

II. Related Work

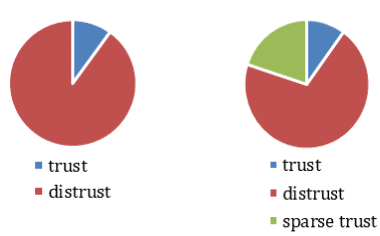
In order to overcome cold start and sparse evaluation, the trust information is incorporated into recommender systems. For example, Golbeck introduced a method based on trust propagation to calculate the rating projections for the target project [8]. Her method only allows the trust to reach a certain level for propagation between users. Janali and Ester [9] proposed the TrustWalker, which uses the random propagation nature of trust in the trust network to assist the recommendation of target users.

In addition to using the propagation mechanism of trust, another way of combining trust is to take advantage of the latent factors of trust. Ma et al. [10] proposed a latent feature model SoRec which adds social relationship constraints to the user-item matrix. SoRec connects rating data and trust data by sharing the same user latent feature space. Then, they proposed the RSTE model [11] which combines the user's preferences with his trusted neighbors' favors. Yang et al. [12] established the recommendation system from the trustor and trustee and then combined them to improve the recommendation performance. Then they improved their model with Probabilistic Matrix Factorization in [13]. Fang et al. [7] decomposed trust from a sociological perspective to predict trust relationships between users and help users make recommendations. Guo et al. [14] extended the SVD++ model with trust information, which effectively improves recommendation performance. K. Niemann et al. [15] introduce a new way of detecting semantic similarities between learning objects by analyzing their usage in web portals. They take this new similarity measure to enhance existing recommendation approaches.

III. Preliminaries

A. Concepts

The wide application of trust makes it a general term for multiple interpretations in different contexts [1], [8], [16]. We summarize the above definitions and refine the definition of trust in the recommendation scenario.



(a) (b)

Fig. 1. Relationship model of social trust and sparse trust. (a) Social trust; (b) Sparse trust.

Definition 1. (trust). In the recommendation scenario, trust is an emotional tendency that the subject (trustor) is willing to depend on the object (trustee or system), with a feeling of similarity, closeness, or security. It is often described in the form of probability, and it is highly time-dependent and space-dependent.

Definition 2. (sparse trust). Sparse trust is an implicit trust relationship which is covered and disturbed by big data noise [1]. The sparse trust relationship model is presented in Fig. 1.

As shown in Fig. 1(a), in the widely adopted social trust model, trust has two states: trust and distrust. However, in distrust data, big data noise covers some sparse trust relationships. In Fig. 1(b), we show the relationship among sparse trust, distrust, and social trust.

B. Basic Matrix Factorization

The rating matrix is represented by $R=[r_{u,i}]_{m\times n}$, where entity $r_{u,i}$ represents the rating of user u on item i , and the entity $r_{u,i}$ is normally an integer in the interval [1, $R_{\max}$ (from 1 to 5)]. The core idea of matrix factorization is to get two d-dimensional matrices: the user-feature matrix $P\in\mathbb{R}^{d\times m}$ and item-feature matrix $Q\in\mathbb{R}^{d\times m}$. Thus, we use p_u and q_j to denote the latent feature vector of user u in matrix P and item i in matrix Q , respectively. Then, we can predict the unknown rating by $\hat{r}_{u,j}=q_j^{\mathrm{T}}p_u$. The loss function is given as follows:

$$L=\frac{1}{2}\sum_{u}\sum_{j\in I_u}(q_j^{\mathrm{T}}p_u-r_{u,j})^2+\frac{\lambda}{2}(\sum_{u}\|p_u\|_F^2+\sum_{J}\|q_j\|_F^2) \quad (1)$$

where λ is a parameter controlling the model complexity, $\|\cdot\|_F$ denotes the Frobenius norm, and the set of items rated by user u is represented by I_u .

We use a graph $G=(\mathcal{V},\varepsilon)$ to represent a trust network with m nodes, where node v represents the set of m users and edge ε represents the trust relationships between users. We use the adjacency matrix $T=[t_{u,i}]_{m\times n}$ to denote the user-user trust matrix (Both social trust networks and sparse trust networks adopt this method), where each entity $t_{u,i}$ represents the trust value of user u to user v. In order to establish an association between the rating matrix and trust matrix, we share the same user-feature space of the user between the two matrices. We use $P\in\mathbb{R}^{d\times m}$ and $Z\in\mathbb{R}^{d\times m}$ to denote the trustor-feature matrix and trustee-feature matrix, respectively. We use p_u and z_v to denote the d-dimensional latent feature vector of trustor u and trustee v , respectively. So, the unknown trust relationship between user u and v can be predicted by $\hat{t}_{u,z}=z_v^{\mathrm{T}}p_u$. The loss function is given as follows:

$$L=\frac{1}{2}\sum_{u}\sum_{v\in T_u}(z_v^{\mathrm{T}}p_u-t_{u,v})^2+\frac{\lambda}{2}(\sum_{u}\|p_u\|_F^2+\sum_{v}\|z_v\|_F^2) \quad (2)$$

where T_u denotes the set of users trusted by user u .

Next, in Section IV, we will detail the mining process of sparse trust, and in Section V, we will apply the sparse trust to the recommendation.

IV. SPARSE TRUST MODEL

We will introduce our sparse trust mining model in this section. The sparse trust mining process is shown in Fig. 2.

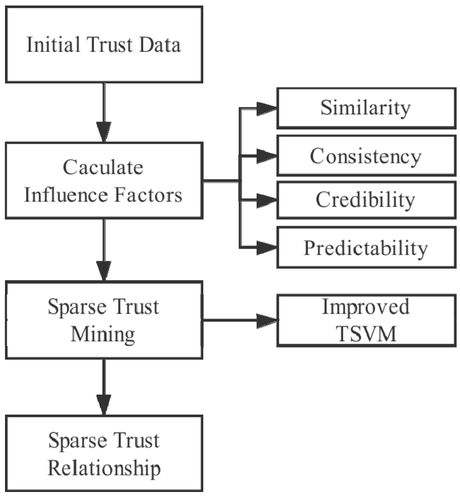


Fig. 2. Sparse trust mining process.

A. *Trust Influence Factor Decomposition*

As shown in Fig. 2, we will mine sparse trust based on two main aspects: the user preferences and user behavior. We decompose user preferences into two finer-grained factors: similarity and consistency. The similarity in this section refers specifically to the number or proportion of items jointly rated by the trustor and trustee. Thus, the similarity is computed by Jaccard similarity coefficient:

$$Sim(u,v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \tag{3}$$

where I_u and I_v represent the set of items that are rated by user u and user v, respectively.

The consistency refers to the rating habits of users. Because the rating ranges of different users are different, the consistency of user ratings is an important criterion for measuring whether their preferences are similar, that is, whether users have the same criteria for good or bad. Hence, the consistency can be computed by the Pearson correlation coefficient:

$$Con(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,i} - \overline{r_u})(r_{v,i} - \overline{r_v})}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,i} - \overline{r_u})^2} \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \overline{r_v})^2}} \tag{4}$$

where $I_{u,v} = I_u \cap I_v$ is the set of items that u and v jointly rated, and $\overline{r_u}$, $\overline{r_v}$ are the average rating of user u and user v respectively.

On the other hand, we decompose user behavior into two finer-grained factors: credibility and predictability. Credibility refers to whether the rating behavior of the user is honest. The credibility of a trustee's behavior can be analyzed from two aspects: his own rating habits and the difference between his rating and the ratings of most people. The credibility of users is also affected by their liveness. Hence, the credibility is computed by the following formula:

$$C_1 = \frac{\sum_{i \in I_v} (r_{v,i} - \overline{r_v})^2}{N} ; \ C_2 = \frac{\sum_{i \in I_v} (r_{v,i} - \overline{r_i})^2}{N} \tag{5}$$

where C_1 denotes the volatility of the ratings of user v, C_2 denotes the difference between the ratings of user v and average rating, $\overline{r_i}$ is the average rating on item i, and N is the number of items that are rated by user v. The liveness l is defined by the following formula:

$$l = \begin{cases} \frac{N}{N^u} & N < N^u \\ 1 & otherwise \end{cases} \tag{6}$$

$$Cre(v) = l(C_1 + \frac{1}{C_2}) \tag{7}$$

where N^u is the minimum number of items that users who we consider to be active users need to rate.

The objectivity refers to whether the evaluations of users are consistent with objective facts. In addition to whether the user's rating reflects the true value of the item, we use the consistency of the user ratings with the average ratings of most people as a measure. Thus, the objective is computed by the Pearson correlation coefficient between them:

$$Obj(v) = \frac{\sum_{i \in I_v} (r_{v,i} - \overline{r_v})(\overline{r_i} - \overline{r})}{\sqrt{\sum_{i \in I_v} (r_{v,i} - \overline{r_v})^2} \sqrt{\sum_{i \in I_v} (\overline{r_i} - \overline{r})^2}} \tag{8}$$

where $\overline{r}$ is the average rating of all items.

Formally, the set of the sparse trust influence factors is denoted by $A_{u,v} = \{Sim(u,v), Con(u,v), Cre(v), Obj(v)\}$.

B. *Sparse Trust Mining*

According to the definition of sparse trust, we divide the trust between user u and user v into three states: trust ($t_{u,v}$ = 1), distrust ($t_{u,v}$ = -1) and unclear trust relationships ($t_{u,v}$ = 0).

Firstly, in order to make the prediction more accurate, we directly label the unlabeled samples with poor performance on $A_{u,v}$ as distrust ($t_{u,v}$ = -1). Then we get the training set $D_l = \{(x_1, y_1), (x_2, y_2), ..., (x_l, y_l)\}$ and $D_u = \{x_{l+1}, x_{l+2}, ..., x_{l+u}\}$, where x_i represents the set of four aspects of the $i-th$ sparse trust relationship, and $y_i \in \{-1,+1\}$ is the trust label of the $i-th$ sparse trust relationship. For example, suppose the $i-th$ sparse trust relationship is between user u and v , $x_i = A_{u,v}$ and $y_i = t_{u,y}$. The learning goal of TSVM is to give labels $\hat{y} = (\hat{y}_{l+1}, \hat{y}_{l+2}, ..., \hat{y}_{l+u})$, $\hat{y}_i = \{-1,+1\}$, for unlabeled samples in D_u , so that:

$$\min_{\omega,b,\hat{y},\xi} \frac{1}{2}\|\omega\|_2^2 + C_l \sum_{i=1}^{l} \xi_i + C_u \sum_{i=l+1}^{m} \xi_i \quad (9)$$

where ξ is the slack variable, C_l and C_u are artificially parameters that balance the complexity of the model and the importance of the sample. Then, we will get a sparse trust network in which the sparse trust between users has been built.

V. SSL-STR: SPARSE TRUST RECOMMENDATION

In order to reasonably adopt the trust information, we consider the combined influence of sparse trust and social trust. Our model is based on the SVD++ model proposed by Koren [6]. Inspired by [16], we use the same approach to integrate sparse trust into the model. Formally, the rating $\hat{r}_{u,i}$ can be predicted by:

$$\hat{r}_{u,i} = b_i + b_u + \mu + q_i^{\mathrm{T}}(p_u + |I_u|^{-\frac{1}{2}} \sum_{j \in I_u} y_j + \alpha |T_u|^{-\frac{1}{2}} \sum_{v \in T_u} w_v + (1-\alpha)|S_u|^{-\frac{1}{2}} \sum_{z \in S_u} f_z) \quad (10)$$

where vector b_i represents the deviation of the ratings of item i from the average rating, vector b_u represents the deviation of the ratings given by user u from the average rating, μ denotes the average rating and y_i represents the influence factors of other items rated by user u on item i. So, the user implicit feature matrix p_u can be replaced by the sum of the user features and the effects of the items they rated on each feature, that is $(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i)$. T_u and S_u are the set of users that trusted by user u in social trust and sparse trust networks respectively, w_v and f_z are the feature vectors of the users that trusted by user u in social trust and sparse trust networks respectively. The parameter α denotes the weight of the influence of social trust and sparse trust in the recommendation. We use $|T_u|^{-\frac{1}{2}} \sum_{v \in T_u} w_v$ to represent the effects caused by the users that trusted by user u in social trust, and $|S_u|^{-\frac{1}{2}} \sum_{z \in S_u} f_z$ in sparse trust.

In Section Ⅲ, we introduced the basic matrix factorization of trust data and rating data. Next, we will merge them and give a matrix factorization model. The objective function that needs to be minimized is given as (11):

$$L = \frac{1}{2}\sum_u \sum_{i \in I_u} (\hat{r}_{u,i} - r_{u,i})^2 + \frac{\lambda_t}{2} \sum_u \sum_{v \in T_u} (\hat{t}_{u,v} - t_{u,v})^2 + \frac{\lambda_s}{2} \sum_u \sum_{z \in S_u} (\hat{s}_{u,z} - s_{u,z})^2 \quad (11)$$

where $\hat{t}_{u,v} = w_v^{\mathrm{T}} p_u$ and $\hat{s} = f_z^{\mathrm{T}} p_u$ are the trust constrains in social trust networks and sparse trust networks respectively. The parameters λ_t and λ_s are used to control the degree of trust regularization.

To avoid over-fitting, we adopt the weighted-regularization technique [16] into our model. The final loss function that needs to be minimized is given as follows:

$$\begin{aligned} L = {} & \frac{1}{2}\sum_u \sum_{i \in I_u} (\hat{r}_{u,i} - r_{u,i})^2 + \frac{\lambda_t}{2} \sum_u \sum_{v \in T_u} (\hat{t}_{u,v} - t_{u,v})^2 + \\ & \frac{\lambda_s}{2} \sum_u \sum_{z \in S_u} (\hat{s}_{u,z} - s_{u,z})^2 + \frac{\lambda}{2}\sum_u |I_u|^{-\frac{1}{2}} b_u^2 + \frac{\lambda}{2}\sum_i |U_i|^{-\frac{1}{2}} b_i^2 \\ & + \sum_u (\frac{\lambda}{2}|I_u|^{-\frac{1}{2}} + \frac{\lambda_t \alpha}{2}|T_u|^{-\frac{1}{2}} + \frac{\lambda_s(1-\alpha)}{2}|S_u|^{-\frac{1}{2}})\|p_u\|_F^2 \\ & + \frac{\lambda}{2}\sum_i |U_i|^{-\frac{1}{2}} \|q_i\|_F^2 + \frac{\lambda}{2}\sum_j |U_j|^{-\frac{1}{2}} \|y_j\|_F^2 + \frac{\lambda}{2}|T_v^+|^{-\frac{1}{2}} \|w_v\|_F^2 \\ & + \frac{\lambda}{2}|S_z^+|^{-\frac{1}{2}} \|f_z\|_F^2 \end{aligned} \quad (12)$$

where U_i and U_j are the sets of users who rated item i and item j, respectively; T_v^+ is the set of users who trust user v in the social trust network and S_z^+ is the set of users who trust user z in the sparse trust network. We can minimize (13) by performing the following gradient descents:

$$\begin{aligned} & \frac{\partial L}{\partial b_u} = \sum_{i \in I_u} e_{u,i} + \lambda |I_u|^{-\frac{1}{2}} b_u \\ & \frac{\partial L}{\partial b_i} = \sum_{u \in U_i} e_{u,i} + \lambda |U_i|^{-\frac{1}{2}} b_i \\ & \frac{\partial L}{\partial q_j} = \lambda |U_i|^{-\frac{1}{2}} q_j + \\ & \sum_{u \in U_i} e_{u,i}(p_u + |I_u|^{-\frac{1}{2}} \sum_{j \in I_u} y_j + \alpha |T_u|^{-\frac{1}{2}} \sum_{v \in T_u} w_v + (1-\alpha)|S_u|^{-\frac{1}{2}} \sum_{z \in S_u} f_z) \\ & \frac{\partial L}{\partial p_u} = \sum_{i \in I_u} e_{u,i} + \lambda_t \sum_{v \in T_u} e_{u,v} w_v + \lambda_s \sum_{z \in S_u} e_{u,z} f_z + (\lambda |I_u|^{-\frac{1}{2}} + \lambda_t \alpha |T_u|^{-\frac{1}{2}} + \\ & \lambda_s(1-\alpha)|S_u|^{-\frac{1}{2}}) p_u \\ & \forall j \in I_u, \frac{\partial L}{\partial y_j} = \sum_{i \in I_u} e_{u,i} |I_u|^{-\frac{1}{2}} q_i + \lambda |U_i|^{-\frac{1}{2}} y_i \\ & \forall v \in T_u, \frac{\partial L}{\partial w_v} = \sum_{i \in I_u} e_{u,i} |T_u|^{-\frac{1}{2}} q_i + \lambda_t e_{u,v} p_u + \lambda |T_v^+|^{-\frac{1}{2}} w_v \\ & \forall z \in S_z, \frac{\partial L}{\partial f_z} = \sum_{i \in I_u} e_{u,i} |S_u|^{-\frac{1}{2}} q_i + \lambda_s e_{u,z} p_u + \lambda |S_z^+|^{-\frac{1}{2}} f_z \end{aligned} \quad (13)$$

where $e_{u,i} = \hat{r}_{u,i} - r_{u,i}$ is the error formula of user ratings, $e_{u,v} = \hat{t}_{u,v} - t_{u,v}$ is the error formula of the trust from user u

towards user v in the social trust network, and $e_{u,z} = \hat{t}_{u,z} - t_{u,z}$ is the error formula of the trust from user u towards user z in the sparse trust network.

VI. EXPERIMENTS

In this section, we conduct our experiments on four real-world datasets. The source code of the *SSL-STR* model is publicly accessible at Github: https://github.com/legenddi.

A. Dataset

Four real-world datasets are used in our experiments: Epinions1(Ep1), Epinions2(Ep2), FilmTrust(FT), and CiaoDVD(CD). Both rating information and social trust relationships are available in the four datasets. To test the performance of our approach in different data environments, we randomly select two subsets of Epinions: Epinions1 and Epinions2. We use CiaoDVD to simulate the trust cold start situation, Epinions1 to simulate the rating cold start situation of the system and Epinions2 to simulate the situations with both the rating cold start and trust cold start. Besides, in Epinions1, Epinions2, and CiaoDVD, the ratings are integers from 1 to 5 with step 1, and in FilmTrust the ratings are numbers from 0.5 to 4 with step 0.5. The statistic information is summarized in Table I.

TABLE I. STATISTIC INFORMATION ABOUT THE DATASETS

Aspect	FilmTrust	CiaoDVD	Epinions1	Epinions2
Num of Users	1508	7375	5000	5000
Num of Items	2071	99746	87485	83800
Num of Ratings	35497	280391	59742	58072
Num of Trust	1853	111781	25700	1764

B. Evaluation Metrics

In this paper, we use the 5-fold cross-validation approach. Each dataset will be divided into five folds. And then, five iterations will be conducted on each dataset. Two well-known measures are used to evaluate the recommendation performance of the methods. That is, the mean absolute error and root mean square error (MAE and RMSE). The two measures are used to evaluate the deviation between the predicted values and the original values, and the smaller the values of them, the smaller the prediction error.

$$MAE = \frac{\sum_{u,i} \left| \hat{r}_{u,i} - r_{u,i} \right|}{N}, RMSE = \sqrt{\frac{\sum_{u,i} (\hat{r}_{u,i} - r_{u,i})^2}{N}} \tag{14}$$

where N denotes the number of ratings.

C. Recommendation Performance Comparison

Four state-of-the-art approaches are used to compare with our approach, which can be divided into two groups: (1) Rating-only models: SVD++ [6]; (2) Trust-based models: RSTE [11], TrustMF(TM) [12], and TrustSVD(TS) [16]. For the above methods, we set the optimal parameters that are suggested in the literature as illustrated in Table II.

D. Results and Analysis

We use " * " to mark the method that has the best performance on each dataset except our method in Table III. Moreover, the "improve" lines in the table present our improvements over other approaches.

TABLE II. PARAMETER SETTINGS OF COMPARED METHODS

Approaches	Parameters Setting
SVD++	Recommended in [6]
RSTE	α=1.0, λ=0.001, λ_T=1
TrustMF	λ=0.001, λ_T=1
TrustSVD	λ=0.001, λ_T=1
SSL-STR	α=0.3, λ=0.001, λ_T=1

TABLE III. EXPERIMENTAL RESULTS

Data	Metrics	SVD++	RSTE	TM	TS	SSL-STR
FT	MAE	0.642	0.656	0.637	0.593*	**0.584**
	Improve	9.03%	10.98%	8.32%	1.52%	-
	RMSE	0.870	0.852	0.815	0.745*	**0.737**
	Improve	15.29%	13.50%	9.57%	1.07%	-
CD	MAE	0.869	0.934	0.860	0.845*	**0.808**
	Improve	7.02%	13.49%	6.05%	4.38%	-
	RMSE	1.138	1.203	1.083	1.078*	**1.045**
	Improve	8.17%	13.13%	3.51%	3.06%	-
Ep1	MAE	0.894*	1.007	0.899	0.896	**0.886**
	Improve	0.89%	12.02%	1.45%	1.12%	-
	RMSE	1.184*	1.299	1.192	1.184	**1.181**
	Improve	0.25%	9.08%	0.92%	0.25%	-
Ep2	MAE	0.866	1.063	0.883	0.866*	**0.864**
	Improve	0.12%	18.72%	2.15%	0.23%	-
	RMSE	1.246	1.383	1.267	1.171*	**1.170**
	Improve	6.10%	15.40%	7.66%	0.09%	-

As the comparison shown in Table III, we can summarize the following conclusions:

- Analysis from the perspective of model types, the rating-only approaches remains competitive. However, in the cold start situation, trust-based approaches have some advantages. Specifically, as shown in Table III, for all the comparison approaches, TrustSVD performs best in FilmTrust, CiaoDVD, and Epinions2, and SVD++ performs best in Epinions1.
- It can be concluded that trust cannot provide effective help for recommendations in cases where trust data is very sparse. So, the rating-only approach achieves better performance in Epinions1. In Epinions2, we simulate the situation in which rating cold start coexists with trust cold start. The performance of the two methods differs little in value. But the trust-based have a better performance.
- Our approach performs best on all four datasets. Although the numerical value increases in Epinions2 is not obvious. Koren [6] noted that even small improvements in RMSE and MAE can make a significant difference in practice.

E. Impact of Parameter

We investigate the impact of the parameter α on our method. As shown in Fig. 3, in FilmTrust and CiaoDVD, as parameter α decreases, the values of MAE and RMSE also decreases. Until α=0.3, MAE and RMSE get the lowest values. When α continues to decrease, the values of MAE and RMSE increases slightly. We can get the minimum values of MAE and RMSE when α=0.3. So, the most appropriate proportion coefficient between social trust and sparse trust is 0.3. Different from the first two data sets, in Epinions1 and Epinions2, as parameter α increases, the values of MAE and RMSE decrease. Our model performs best in Epinions1 when α=0.8, and performs best in Epinions2 when α=0.9.

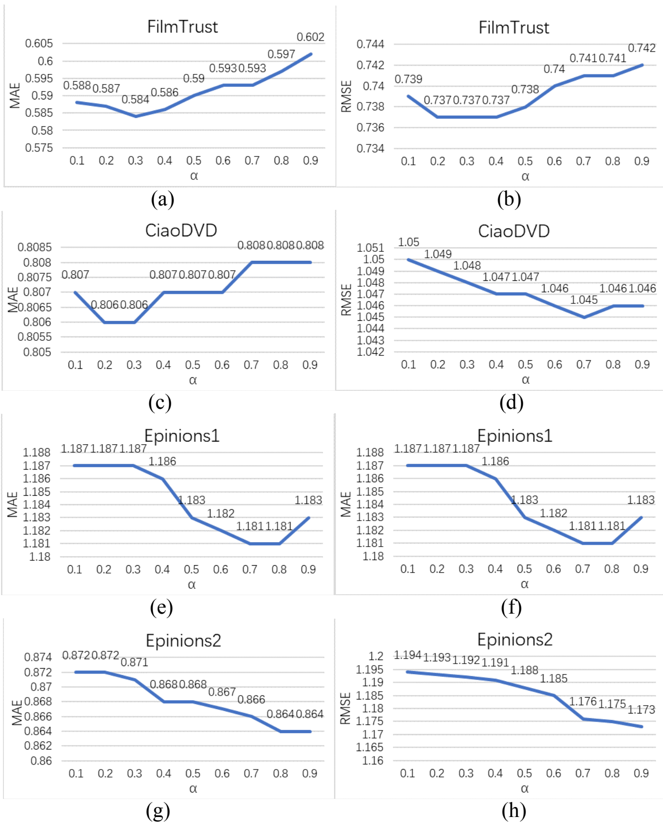


Fig. 3. Impact of parameter on the SSL-STR recommendation performance: ((a) and (b)) FilmTrust; ((c) and (d)) CiaoDVD; ((e) and (f)) Epinions1; ((g) and (h)) Epinions2.

VII. Conclusion

To improve the performance of the recommender system, we mined the sparse trust covered by big data noise and then proposed a sparse trust recommendation method SSL-STR. Specifically, we decomposed the aspects influencing trust-building into four finer-grained factors: similarity, consistency, credibility, and predictability. After that, we mined the sparse trust between users through the TSVM algorithm, which makes full use of unlabeled trust data. Finally, we extended the SVD++ model with social trust and sparse trust, and balanced the impact of these two trust relationships on the recommendation effect. Experiments show that our SSL-STR can effectively mine potential trust relationships among users and significantly improve the accuracy of recommendation.

References

[1] Liu H , Xia F , Chen Z , et al, "TruCom: Exploiting Domain-Specific Trust Networks for Multicategory Item Recommendation," *IEEE Systems Journal*, vol 11, no 1, pp. 295-304, 2017.

[2] L. Sun, X. Wang, Z. Wang, H. Zhao, and W. Zhu, "Social-aware video recommendation for online social groups," *IEEE Transactions on Multimedia*, vol. 19, no. 3, pp. 609–618, March 2017.

[3] Yiding Liu, Tuan-Anh Nguyen Pham, Gao Cong, and Quan Yuan, "An experimental evaluation of point-of-interest recommendation in location-based social networks," *Proc. VLDB Endow*, vol. 10, no.10, pp. 1010–1021, 2017.

[4] Mengdi Liu, et al, "Roundtable Gossip Algorithm: A Novel Sparse Trust Mining Method for Large-scale Recommendation Systems," in *International Conference on Algorithms and Architectures for Parallel Processing*. Springer, May. 2018, pp. 495-510.

[5] T Joachims, "Transductive Inference for Text Classification using Support Vector Machines," in *Proceedings of the Sixteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc,1999, pp.200-209.

[6] Yehuda Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp.426-434.

[7] Fang Hui, Bao Yang and Zhang Jie, "Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation," in *Proceedings of Twenty-Eighth AAAI Conference on Artificial Intelligence.* AAAI Press, pp.30-36, 2014.

[8] J. Golbeck, "Generating predictive movie recommendations from trust in social networks.," in *International Conference on Trust Management.* Springer, 2006, pp. 93–104.

[9] Mohsen Jamali and Martin Ester, "TrustWalker: a random walk model for combining trust-based and item-based recommendation," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp.397-406.

[10] Hao Ma, Haixuan Yang, Michael R. Lyu and Irwin King, "SoRec: social recommendation using probabilistic matrix factorization," in *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 2008, pp.931-940.

[11] Hao Ma, Irwin King and Michael R. Lyu, "Learning to recommend with social trust ensemble," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM*, 2009, pp.203-210.

[12] Bao Yang, Yu Lei, Dayou Liu and Jiming Liu, "Social Collaborative Filtering by Trust," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*. AAAI Press, 2013, pp.2747-2753.

[13] Bao Yang, Yu Lei, Jiming Liu and Wenjie Li, "Social Collaborative Filtering by Trust," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.39, no.8, pp.1633-1647, 2017.

[14] Guibing Guo, Jie Zhang and Neil Yorke-Smith, "TrustSVD: Collaborative Filtering with Both the Explicit and Implicit Influence of User Trust and of Item Ratings," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*. AAAI Press, 2015, pp.123-129.

[15] K. Niemann and M. Wolpers, "Creating usage context-based object similarities to boost recommender systems in technology enhanced learning," *IEEE Transactions on Learning Technologies,* vol. 8, no. 3, pp. 274–285, July 2015.

[16] Peixin Gao, Hui Miao, John S. Baras, and Jennifer Golbeck, "STAR: Semiring Trust Inference for Trust-Aware Social Recommenders." in *Proceedings of the 10th ACM conference on Recommender systems*. ACM, 2016, pp.301–308.