A Novel Recommendation Model Regularized with User Trust and Item Ratings

Guibing Guo, Jie Zhang, and Neil Yorke-Smith

Abstract—We propose TrustSVD, a trust-based matrix factorization technique for recommendations. TrustSVD integrates multiple information sources into the recommendation model in order to reduce the data sparsity and cold start problems and their degradation of recommendation performance. An analysis of social trust data from four real-world data sets suggests that not only the explicit but also the implicit influence of both ratings and trust should be taken into consideration in a recommendation model. TrustSVD therefore builds on top of a state-of-the-art recommendation algorithm, SVD++ (which uses the explicit and implicit influence of rated items), by further incorporating both the explicit and implicit influence of trusted and trusting users on the prediction of items for an active user. The proposed technique is the first to extend SVD++ with social trust information. Experimental results on the four data sets demonstrate that TrustSVD achieves better accuracy than other ten counterparts recommendation techniques.

Index Terms—Recommender systems, social trust, matrix factorization, implicit trust, collaborative filtering

1 Introduction

Recommender systems have been widely used to provide users with high-quality personalized recommendations from a large volume of choices. Robust and accurate recommendations are important in e-commerce operations (e.g., navigating product offerings, personalization, improving customer satisfaction), and in marketing (e.g., tailored advertising, segmentation, cross-selling). Collaborative filtering (CF) is one of the most popular techniques to implement a recommender system [1]. The idea of CF is that users with similar preferences in the past are likely to favour the same items (e.g., movies, music, books, etc.) in the future. CF has also been applied to tasks besides item recommendations, in domains such as image processing [2] and bioinformatics [3]. However, CF suffers from two well-known issues: *data sparsity* and *cold start* [4]. The former issue refers to the fact that users usually rate only a small portion of items, while the latter indicates that new users only give a few ratings (a.k.a. cold-start users). Both issues severely degrade the efficiency of a recommender system in modelling user preferences and thus the accuracy of predicting a user's rating for an unknown item.

To help resolve these issues, many researchers [5], [6], [7], [8], [9] attempt to incorporate social trust information into their recommendation models, given that model-based CF approaches outperform memory-based approaches [10]. These approaches further regularize the user-specific feature vectors by the phenomenon that friends often influence each other in recommending items. However, even the best performance reported by the latest work [9] can be inferior to that of other state-of-the-art models which are merely based on user–item ratings. For instance, a well-performing trust-based model [8] obtains 1.0585 on data set Epinions.com in terms of Root Mean Square Error (RMSE), whereas the performance of a user–item baseline (see, Koren [11], Section 2.1) can achieve 1.0472 in terms of RMSE.[1]

One possible explanation is that these trust-based models focus too much on the utility of user trust but ignore the influence of item ratings themselves. To investigate this phenomenon, we conduct an empirical trust analysis based on four real-word data sets (FilmTrust, Epinions, Flixster and Ciao). Three important observations emerge. First, trust information is also very sparse, yet complementary to rating information.

Hence, focusing too much on either one kind of information may achieve only marginal gains in predictive accuracy. Second, users are strongly correlated with their outgoing trusted neighbours (i.e., trustees) whereas they have a weakly positive correlation with their *trust-alike* neighbours (e.g., friends). We defer the definition of trust-alike relationships to Section 3.1. The third observation further indicates a similar conclusion with in-coming trusting neighbours (i.e., trusters). The implication is that existing trust-based models may not work well if there exists only trust-alike relationships. Given that very few trust networks exist, it is better to have a more general trust-based model that can well operate on both trust and trust-alike relationships. These observations motivate us to consider both explicit and implicit influence of item ratings and of user trust in a unified trust-based model. The influence can be explicit—real values of ratings and trust—or implicit—who rates

- *G. Guo is with the Software College, Northeastern University, China, and Nanyang Technological University, Singapore. E-mail: guogb@swc.neu.edu.cn.*
- *J. Zhang is with the School of Computer Engineering, Nanyang Technological University, Singapore. E-mail: ZhangJ@ntu.edu.sg.*
- *N. Yorke-Smith is with the Suliman S. Olayan School of Business, American University of Beirut, Lebanon. E-mail: nysmith@aub.edu.lb.*

Manuscript received 26 Apr. 2015; revised 18 Jan. 2016; accepted 31 Jan. 2016. Date of publication 11 Feb. 2016; date of current version 1 June 2016. Recommended for acceptance by Y. Koren.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2016.2528249

1. Smaller RMSE values indicate better predictive accuracy. The result is reported by the well-known recommendation toolkit MyMediaLite (www.mymedialite.net/examples/datasets.html).

what (for ratings) and who trusts whom (for trust). The implicit influence of ratings has been demonstrated useful in providing accurate recommendations [12]. We will later show that implicit trust can also provide added value over explicit trust.

In this article, we propose a novel trust-based recommendation model regularized with user trust and item ratings, termed *TrustSVD*. Our approach builds on top of a state-of-the-art model SVD++ [12] through which both the explicit and implicit influence of user–item ratings are involved to generate predictions. In addition, we further consider the influence of user trust (including trustees and trusters) on the rating prediction for an active user. To the authors' knowledge, our work is the first to extend SVD++ with social trust information. Specifically, on one hand the implicit influence of trust (who trusts whom) can be naturally added to the SVD++ model by extending the user modelling. On the other hand, the explicit influence of trust (trust values) is used to constrain that user-specific vectors should conform to their social trust relationships. This ensures that user-specific vectors can be learned from their trust information even if a few or no ratings are given. In this way, the concerned issues can be better alleviated. Our method is novel for its consideration of both the explicit and implicit influence of item ratings and of user trust. In addition, a weighted-λ-regularization technique is used to help avoid over-fitting for model learning. The experimental results on the four real-world data sets demonstrate that our approach works significantly better than other trust-based counterparts as well as high-performing ratings-only models (10 approaches in total) in terms of predictive accuracy, and is more capable of coping with the cold-start situations.

Summary of contributions. Our first contribution is to conduct an empirical trust analysis and observe that trust and ratings can complement to each other, and that users may be strongly or weakly correlated with each other according to different types of social relationships. These observations motivate us to consider both explicit and implicit influence of ratings and trust into our trust-based model. Potentially, these observations could be also beneficial for solving other kinds of recommendation problems, e.g., top-N item recommendation.

Our second contribution is to propose a novel trust-based recommendation approach (TrustSVD[2]) that incorporates both (explicit and implicit) influence of rating and trust information. No previous work has considered these two types of influence simultaneously, and this is the first work that extends SVD++ with social trust information. Specifically, the implicit influence of a user's trusters and trustees is used to model her feature-specific vector besides the implicit feedback of rated items. The explicit influence of trust values is used to factorize trust matrix into truster/trustee-specific vectors, bridging ratings and trust into a unified model.

Our third contribution is to conduct extensive experiments to evaluate the effectiveness of the proposed approach in two different types of testing views of all users and cold-start users. By comparing with 10 baseline and state-of-the-art recommendation models, we show that our approach performs better in terms of predictive accuracy in the two testing views. We further investigate the performance of trust-based models with respect to the number of trust neighbours per user, and find out that our approach consistently provides better recommendation performance.

Outline. The rest of this article is organized as follows. Section 2 provides a brief overview of trust-based recommender systems in the literature. The trust data from four real-world data sets is analyzed in Section 3 where three important observations are concluded. Then, our TrustSVD approach is elaborated in Section 4 regarding model formalization and learning, followed by the empirical evaluation conducted in Section 5. Finally, Section 6 concludes the present work and outlines future research.

2 Related Work

Trust-aware recommender systems have been widely studied [14], given that social trust provides an alternative view of user preferences other than item ratings [15]. Yuan et al. [16] find that trust networks are small-world networks where two random users are socially connected in a small distance, indicating the implication of trust in recommender systems. In fact, it has been demonstrated that incorporating the social trust information of users can improve the performance of recommendations [9], [17]. There are two main recommendation tasks in recommender systems, namely item recommendation and rating prediction. Most algorithmic approaches are only (or best) designed for either one of the recommendations tasks, and our work focuses on the rating prediction task.

2.1 Rating Prediction

Many approaches have been proposed for rating prediction, including both memory- and model-based methods. We survey some representative memory-based methods. Massa and Avesani [18] show that trust-aware recommender systems can help enable more items for recommendation while preserving competing predictive accuracy, where trust is propagated in trust networks to evaluate users' weights. Similarly, Golbeck [19] proposes an approach, *TidalTrust*, to aggregate the ratings of trusted neighbours for a rating prediction, where trust is computed in a breadth-first manner. Guo et al. [4] complement a user's rating profile by merging those of trusted users through which better recommendations can be generated, and the cold start and data sparsity problems can be better handled. However, memory-based approaches have difficulty in adapting to large-scale data sets, and are often consume much time in searching candidate neighbours in a large user space. In contrast, model-based approaches can be readily scaled up to large data sets and they generate rating predictions more efficiently. Most importantly, they have been demonstrated to achieve higher accuracy and better alleviate the data sparsity issue than memory-based approaches [10].

Quite a few trust-based recommendation models have been proposed to date. For example, Guo et al. [15] cluster users by multiviews of similarity and trust, in order to resolve the relative low accuracy and coverage issues of clustering-based recommendations. They also make use of both ratings and trust to properly cluster cold-start users, i.e.,

2. A preliminary report of TrustSVD was published at AAAI'15 [13].

mitigating the cold-start problem. Jamali et al. [20] propose a social-aware stochastic block model where users and items are jointly classified to user and item groups in a social rating network. The interactions of group memberships determine if a user will connect with another user (i.e., *link prediction*) or be interested in a target item. However, the empirical results show that this model is better at link prediction than rating prediction. The most popular and widely studied recommendation models are matrix factorization-based models [10], [21] which aim to factorize the user–item rating matrix into two low-rank user-feature and item-feature matrices. Then the predictions can be generated by the inner products of user- and item-specific latent feature vectors [22]. Specifically, Ma et al. [5], [6], [23] developed several approaches by adding different trust regularization terms to a matrix factorization model. They first proposed a social regularization method, *SoRec*, by considering the constraint of social relationships [5]. The idea is to share a common user-feature matrix factorized by ratings and by trust. Ma et al. [23] then proposed a social trust ensemble method, *RSTE*, to linearly combine a basic matrix factorization model and a trust-based neighbourhood model together. The same authors [6] further proposed that the active user's user-specific vector should be close to the average of her trusted users, and used it as a regularization to form a new matrix factorization model *SoReg*. A state-of-the-art approach, *SocialMF*, is proposed by Jamali and Ester [7]. It is built on top of SoRec by reformulating the contributions of trusted users to the formation of the active user's user-specific vector rather than to the predictions of items, and by enabling the property of trust propagation. Yang et al. [24] highlight the domain-specific property of trust and propose three different approaches to infer the trust in each friend circle. By substituting general trust with inferred circle-based trust, they show that the performance of the SocialMF model can be improved. Zhu et al. [25] propose a graph Laplacian regularizer to capture the potential social relationships among users, and form the social recommendation problem as a low-rank semidefinite problem. However, empirical evaluation indicates that very marginal improvements are obtained in comparison with the RSTE model.

More recently, Yang et al. [8] propose a hybrid method *TrustMF* that combines both a truster model and a trustee model from the perspectives of trusters and trustees, that is, both the users who trust the active user and those who are trusted by the user will influence the user's ratings on unknown items. They show that better predictive accuracy is achieved than other trust-based models. Tang et al. [26] consider both global and local trust as the contextual information in their model, where the global trust is computed by a separate algorithm. Yao et al. [17] take into consideration both the explicit and implicit interactions among trusters and trustees in a recommendation model. Most recently, Fang et al. [9] stress the importance of multiple aspects of social trust. They decompose trust into four general factors and then integrate them into a matrix factorization model through which better performance is achieved.

In summary, all these works have shown that a matrix factorization model regularized by trust outperforms the same model without trust. That is, trust is helpful in improving predictive accuracy. However, it is also noted that even the latest work [9] can be inferior to other well-performing ratings-only models such as SVD++ [12]. To explain this phenomenon, we next conduct a trust analysis to investigate the value of trust in recommender systems. We argue that the main reasons are in two-fold. First, the existing trust-based models consider only the explicit influence of ratings; the utility of ratings is not well exploited. The first observation in Section 3 will show that trust information could be even sparser than rating information. This motivates us to build a new trust-based model based on SVD++ that inherently and well considers both the explicit and implicit influence of ratings. Second, these trust-based models do not consider the explicit and implicit influence of trust simultaneously. As will be explained in the second and third observations in Section 3, this may lead to deteriorated performance when being applied to social relationships with smaller correlations with user similarity. Therefore, we incorporate into SVD++ both explicit and implicit influence of social trust, to enhance the generality of our proposed model. By doing so, a better way to utilize user–item ratings and user–user trust is proposed.

2.2 Item Recommendation

We give a short review of trust-based approaches for item recommendation. Specifically, Jamali and Ester [27] propose *TrustWalker*, a random walk model that combines an item-based ranking method and a trust-based nearest neighbour model. Yuan et al. [28] fuse two kinds of social relationships, i.e., friendship and membership in a unified matrix factorization model. In this article, we only consider one kind of social relationships, i.e., either trust or trust-alike, but we verify the generality and application of our model to both kinds of social relationships. Rendle et al. [29] give a state-of-the-art model, *Bayesian personalized ranking* (BPR), for item recommendation based on implicit feedback. The basic idea is to assume that a rated item for an active user is preferred to an unrated item. However, negative samples may be due to the unawareness of items rather than dislike; hence, this assumption may be invalid in practice. To relax this assumption, Zhao et al. [30] propose a *Social Bayesian personalized ranking* (SBPR) method, presuming that an item consumed by an active user is preferred to that consumed by her friends, which is then preferred to the item consumed by other users. However, the assumption of SBPR may not be valid in some social recommender systems. Note that there are some other trust-unaware BPR variants. For example, Pan et al. [31] propose an adaptive BPR to accommodate heterogeneous implicit feedback. These kinds of BPR variants are beyond the discussion of this article.

Essentially, item recommendation and rating predictions are two distinct recommendation tasks. They differ in the following aspects. First, the main objective is different. Item recommendation targets an ordered list of interesting items, and thus does not care about the possible ratings users may give. In contrast, rating prediction aims to predict the possible rating as closely as possible. It has been demonstrated that directly ranking by predicted ratings will result in poor ranking performance [32]. Second, the training process of item recommendation is necessary to consider both positive and negative samples, while rating predictions function only on positive samples, i.e., observed data. Third, item recommendation is often measured in terms of list ranking performance, while rating prediction is estimated by the errors between prediction and ground truth. Lastly, although item recommendation is more prevalent in reality, it is still valuable to predict a possible rating that a user may provide for an unrated item, e.g., in a peer-review system as pointed out by Barbieri et al. [33]. We would like to stress that our work is focused on the recommendation task of rating prediction rather than item recommendation.

3 Trust Analysis

We first introduce the concepts of trust and trust-alike relationships, and then proceed to analyze the influence of trust for rating prediction based on real-world data sets.

3.1 Trust versus Trust-Alike Relationships

For ease of exposition, we first classify the social relationships for recommender systems into two categories, namely *trust* and *trust-alike*, and then depict their similarities and differences. In this article, we adopt the definition of social trust given by Guo [34] as *one's belief towards the ability of others in providing valuable ratings*. It includes a positive and subjective evaluation about other's ability in providing valuable ratings. Trust can be further split into *explit trust* and *implicit trust*. Explicit trust refers to the trust statements directly specified by users. For example, users in Epinions and Ciao can add other users into their trust lists. By contrast, implicit trust is the relationship that is not directly specified by users and that is often inferred by other information, such as user ratings. In this article, we only exploit the value of explicit trust for rating prediction.

We define the *trust-alike* relationships as the social relationships that are similar with, but weaker (or more noisy) than social trust. The similarities are that both kinds of relationships indicate user preferences to some extent and thus useful for recommender systems, while the differences are that trust-alike relationships are often weaker in strength and likely to be more noisy. Typical examples are friendship and membership for recommender systems. Although these relationships also indicate that users may have a positive correlation with user similarity, there is no guarantee that such a positive evaluation always exists and that the correlation will be strong. It is well recognized that friendship can be built based on offline relations, such as colleagues and classmates, which does not necessarily share similar preferences. Trust is a complex concept with a number of properties, such as asymmetry and domain dependence [35], which trust-alike relationships may not hold, e.g., friendship is undirected and domain independent.

For clarity, in this article we refer *trust users* or *trust neighbours* to as the union set of users who trust an active user (i.e., trusters) and of users who are trusted by the active user (i.e., trustees).

3.2 Data Sets

The four data sets used in our analysis and also our later experiments are: Epinions,[3] FilmTrust,[4] Flixster[5] and Ciao.[6]

TABLE 1
Statistics of the Four Data Sets

Feature	Epinions	FilmTrust	Flixster	Ciao
# users	40,163	1,508	53,213	7,375
# items	139,738	2,071	18,197	99,746
# ratings	664,824	35,497	409,803	280,391
density	0.051%	1.14%	0.04%	0.03%
# trusters	33,960	609	47,029	6,792
# trustees	49,288	732	47,029	7,297
# trusts	487,183	1,853	655,054	111,781
density	0.029%	0.42%	0.03%	0.23%

These four data sets are among the few publicly-available data sets that contain both item ratings and social relationships specified by active users. They are used widely in the evaluation of previous trust-aware recommender systems. In particular, the items in Epinions and Ciao are of great variety, such as electronics, sports, computers, etc., while the items in FilmTrust and Flixster are movies only. The ratings in Epinions and Ciao are integers from 1 to 5, while those in the other data sets are real values, i.e., [0.5, 4.0] for FilmTrust, [0.5, 5.0] for Flixster, both with step 0.5. Users in these data sets can share their item ratings with each other and pro-actively connect with users of similar taste, whereby a social network can be constructed. Statistics of the data sets are illustrated in Table 1.

By definition, the social relationships in Epinions and Ciao are trust relationships whereas those in Flixster and FilmTrust are trust-alike relationships. To explain, users in Epinions and Ciao specify others as trustworthy usually based on the evaluation of quality of others' ratings and textual reviews. Flixster adopts the concept of friendship *per se* where user relations are symmetric and related with movies only. Although FilmTrust adopts the concept of trust (with original values from 1 to 10), the publicly available data set contains only binary values. Such degrading may cause much noise and thus we classify the relationships as trust-alike rather than trust.

3.3 Observations

Next we present three observations that are concluded from the four data sets, and that underpin the formation of our trust-based model.

Observation 1. Trust information is very sparse, yet is complementary to rating information.

On one hand, as shown in Table 1, the density of trust is much smaller than that of ratings in Epinions, FilmTrust and Flixster whereas trust is only denser than ratings in Ciao. Both ratings and trust are very sparse in general across all the data sets. In this regard, a trust-aware recommender system that focuses too much on trust (rather than rating) utility is likely to achieve only marginal gains in recommendation performance. As explained earlier, even the latest trust-based model cannot always beat the baseline approaches which generate predictions solely based on ratings. In fact, the existing trust-based models consider only the explicit influence of ratings. That is, the utility of ratings is not well exploited. In addition, the sparsity of explicit trust also implies the importance of involving implicit trust

3. trustlet.org/wiki/Epinions_datasets
4. www.librec.net/datasets.html
5. www.cs.sfu.ca/~sja25/personal/datasets/
6. www.public.asu.edu/~jtang20/datasetcode/truststudy.htm

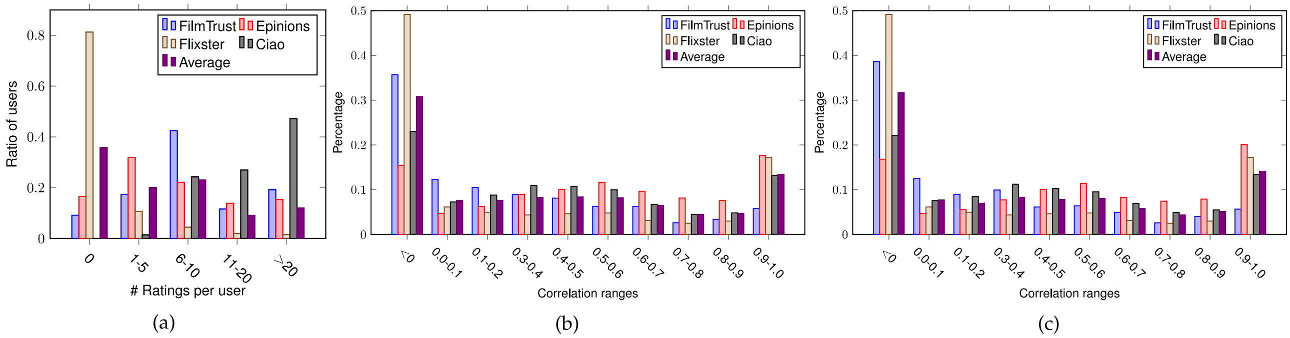


Fig. 1. (a) The distribution of ratio of users who have issued trust statements w.r.t. the number of ratings that they each have given. (b) The correlations between a user's ratings and those of her out-going trusted neighbors (i.e., trustees). (c) The correlations between a user's ratings and those of her in-coming trusting neighbors (i.e., trusters).

in collaborative filtering. Therefore, a better way may stress that both the influence of user trust and item ratings should be taken into account for rating prediction.

On the other hand, trust information is complementary to the rating information. Fig. 1 illustrates the distribution of ratio of users who have specified others as trusted friends (users) with respect to the number of ratings that each of these users has given. It shows that: (1) A portion of users have not rated any items but are socially connected with other users, e.g., 9.20 percent in FilmTrust, 16.65 percent in Epinions and up to 81.28 percent in Flixster.[7] (2) For the *cold-start* users who have rated few items (less than five in our case), trust information can provide a complementary part of source of information with ratio greater than 10 percent on average. (3) The *warm-start* users who have rated a lot of items (e.g., > 20) are not necessary to specify many other users as trustworthy (12 percent on average). As a consequence, although having distinct distributions across the different data sets, trust can be a complementary information source to item ratings for recommender systems.

This observation motivates us to consider both the explicit and implicit influence of item ratings and user trust, making better and more use of them to resolve the data sparsity and cold start problems.

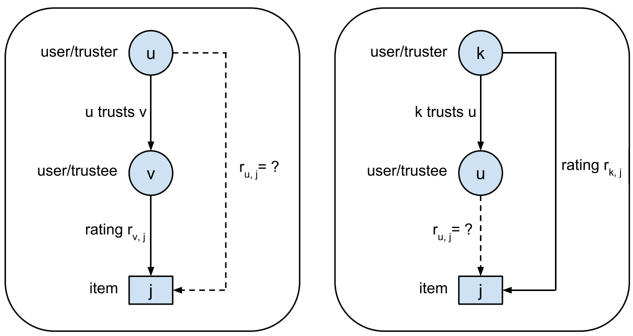
Observation 2. A user's ratings have a **weakly positive correlation** with the average of her **out-going** social neighbours under the concept of trust-alike relationships, and a **strongly positive correlation** under the concept of trust relationships.

Although a user's rating of a certain item is mainly determined by the intrinsic attributes (i.e., properties, features) of the item in question and how she appreciates these features, some extrinsic attributes may also have a non-negligible influence on the user's ratings. In this work, we focus on the influence of social trust in rating prediction, i.e., the influence of trust neighbours on an active user's rating for a specific item, a.k.a. *social influence*. A graphical explanation is given in Fig. 2a. Briefly, user u trusts user v, and user v has rated item j by giving a rating $r_{v,j}$. Then, user u may consider the ratings of her trustees when giving her own rating $r_{u,j}$. Yang et al. [8] have also shown that trusted users will affect users' ratings in their model.

7. All the users in the Ciao data set have rated at least one item.

To have an intuitive comprehension of trustees' influence, we calculate the Pearson correlation coefficient (PCC) between a user's ratings and the average of her social neighbours. The results are presented in Fig. 1b, indicating that: (1) A weakly positive correlation is observed between a user's ratings and the average of the social neighbours in FilmTrust (mean 0.183) and Flixster (0.063). The distributions of the two data sets are similar. (2) Under the concept of trust relationships, on the contrary, a user's ratings are strongly and positively correlated with the average of trusted neighbours. Specifically, a large portion (17.63 percent in Epinions, 13.14 percent in Ciao) of user correlations are in the range of $[0.9, 1.0]$, and (resp. 54.70, 39.14 percent) of user correlations are greater than 0.5. The average correlation is 0.446 in Epinions, and 0.322 in Ciao. Since PCC values are in the range of $[-1, 1]$, values of 0.446 and 0.322 indicate decent correlations. In the social networks with relatively weak trust-alike relationships, implicit influence (i.e., binary relationships) may be more indicative than explicit (but noisy) values for recommendations. In addition, most online social networks do not adopt the concept of trust relationships, but relatively weak trust-alike relationships. Hence, a trust-based model that ignores the implicit influence of item ratings and user trust may lead to deteriorated performance if being applied to such cases. We claim that a good trust-based model should function well not only for strong trust relationships, but also for relatively weak trust-alike ones.

As the strength of correlations depends on the type of social relationships, the performance (thus generality) of a recommendation model may be limited if only explicit



(a) (b)

Fig. 2. The influence of (a) trustees v and (b) trusters k on the rating prediction for the active user u and target item j.

influence of social influence is adopted. Hence, the second observation suggests that incorporating both the explicit and implicit influence of item ratings and user trust may promote the generality of a trust-based model to both trust and trust-alike social relationships.

Observation 3. A user's ratings have a **weakly positive correlation** with the average of her **in-coming** social neighbours under the concept of trust-alike relationships, and a **strongly positive correlation** under the concept of trust relationships.

In social networks, a user may pro-actively connect to a number of social friends, and may also be connected by some other users. Thus, the social influence of one's ratings may flow in both directions. In other words, a user's rating is influenced not just by her trustees, but also by the users who trust her (i.e., her trusters). Yang et al. [8] have also indicated that trusting users have an impact on users' rating prediction. Yao et al. [17] design and combine both truster and trustee regularizers in a unified recommendation model, indicating the value of the influence of trusters. Fig. 2b gives an illustrative presentation. Specifically, user k trusts user u, and gives a rating to the target item j the action of which may further influence user u's rating on the same item. Similar with the last observation, we calculate the correlations between a user's ratings and the average of her trusters' ratings. The results are presented in Fig. 1c. Note that the distribution in Flixster is the same with that in Fig. 1b; the reason is that the friendships in Flixster are symmetric and undirected.

Compared with Fig. 1b, similar distributions of user correlations can be observed, and the differences in each correlation range are relatively small. For example, there are 30.82 percent negative user correlations in Fig. 1b while the percentage in Fig. 1c is around 31.69 percent. Therefore, a similar observation can be drawn from the empirical results, i.e., users have a weakly (strongly) positive correlation with the average of her in-coming social neighbours under the concept of trust-alike (trust) relationships.

The third observation implies that the influence of trusters (in rating prediction) may be comparable with that of trustees, and thus may also provide added value to item ratings. Our approach presented next is built upon these three observations. Our findings also coincide with the conclusion reported by Ma [36], where it is again confirmed that trust and similarity are strongly correlated while friendship and similarity are correlated much less and more varied. In our case, we further classify and generalize the social relationships which have smaller correlations with user interest as trust-alike relationships.

4 TrustSVD: A Trust-Based Recommendation Model

In this section, we first mathematically define the recommendation problem in social rating networks, and then introduce the TrustSVD model in detail.

4.1 Problem Definition

In social rating networks, a user can label (add) other users as trusted friends and thus form a social network. Trust is not symmetric; for example, users u_1 trusts u_3 but u_3 does not specify user u_1 as trustworthy. Besides, users can rate a set of items using a number of rating values, e.g., integers from 1 to 5. These items could be products, movies, music, etc. of interest. The recommendation problem in this work is to predict the rating that a user will give to an unknown item, for example, the value that user u_3 will give to item i_3, based on both a user–item rating matrix and a user–user trust matrix. Other well-recognized recommendation problems include for example top-N item recommendation.

Suppose that a recommender system includes m users and n items. Let $R = [r_{u,i}]_{m\times n}$ denote the user–item rating matrix, where each entry $r_{u,i}$ represents the rating given by user u on item i. For clarity, we preserve symbols u, v for users, and i, j for items. Since a user only rated a small portion of items, the rating matrix R is only partially observed and oftentimes very sparse. Let $I_u = \{i | r_{u,i} \neq 0\}$ denote the set of items rated by user u. Let p_u and q_i be a d-dimensional latent feature vector of user u and item i, respectively. The essence of matrix factorization is to find two low-rank matrices: user-feature matrix $P \in \mathbb{R}^{d\times m}$ and item-feature matrix $Q \in \mathbb{R}^{d\times n}$ that can adequately recover the rating matrix R, i.e., $R \approx P^\top Q$, where $P^\top$ is the transpose of matrix P. The underlying assumption is that both users and items can be characterized by a small number of features. Hence, the rating on item j for user u can be predicted by the inner product of user-specific vector p_u and item-specific vector q_j, i.e., $\hat{r}_{u,j} = q_j^\top p_u$. In this regard, the main task of recommendations is to predict the rating $\hat{r}_{u,j}$ as close as possible to the ground truth $r_{u,j}$. Formally, we can learn the user- and item-feature matrices by minimizing the following loss (objective) function:

$$\mathcal{L}_r = \frac{1}{2}\sum_u \sum_{j\in I_u} \left(q_j^\top p_u - r_{u,j}\right)^2 + \frac{\lambda}{2}\Big(\sum_u \|p_u\|_F^2 + \sum_j \|q_j\|_F^2\Big),$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and λ is a parameter to control model complexity and to avoid over-fitting.

Now suppose that a social network is represented by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ includes a set of m nodes (users) and $\mathcal{E}$ represents the directed trust relationships among users. We can use the adjacency matrix $T = [t_{u,v}]_{m\times m}$ to describe the structure of edges $\mathcal{E}$, where $t_{u,v}$ indicates the extent to which users u trusts v. Usually, only binary values are used, i.e., $t_{u,v} = 1$ means that user u trusts user v whereas $t_{u,v} = 0$ indicates the non-trust relationship. Similarly to the user–item rating matrix R, the trust matrix T is also very sparse. We denote p_u and w_v as the d-dimensional latent feature vector of truster u and trustee v, respectively. We limit the trusters in the trust matrix and the active users in the rating matrix to share the same user-feature space in order to bridge them together. Hence, we have truster–feature matrix $P^{d\times m}$ and trustee–feature matrix $W^{d\times m}$. By employing the low-rank matrix approximation, we can recover the trust matrix by $T \approx P^\top W$. Thus, a trust relationship can be predicted by the inner product of a truster-specific vector and a trustee-specific vector $\hat{t}_{u,v} = w_v^\top p_u$. The matrices P and W can be learned by minimizing the following loss function:

$$\mathcal{L}_t = \frac{1}{2}\sum_u \sum_{v\in T_u^+} \left(w_v^\top p_u - t_{u,v}\right)^2 + \frac{\lambda}{2}\Big(\sum_u \|p_u\|_F^2 + \sum_v \|w_v\|_F^2\Big),$$

where T_u^+ is the set of users trusted by user u, i.e., the set of out-going trusted users.

In summary, by mapping both rating matrix and trust matrix into the same d-dimensional space, we can link the two kinds of information together and thus aim to predict an item's rating $\hat{r}_{u,j}$ as accurately as possible.

4.2 The TrustSVD Model

In line with the three observations of the previous section, our TrustSVD model is built on top of a state-of-the-art model known as SVD++ proposed by Koren [12]. The rationale behind SVD++ is to take into consideration user/item biases and the influence of rated items other than user/item-specific vectors on rating prediction. Formally, the rating for user u on item j is predicted by:

$$\hat{r}_{u,j} = b_u + b_j + \mu + q_j^\top \big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i\big),$$

where b_u, b_j represent the rating bias of user u and item j, respectively; μ is the global average rating; and y_i denotes the implicit influence of items rated by user u in the past on the ratings of unknown items in the future. Thus, user u's feature vector can be also represented by the set of items she rated, and finally modelled as $\big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i\big)$ rather than simply as p_u. Koren [12] has shown that integrating implicit influence of ratings can well improve predictive accuracy. We have already stressed the importance of trust influence for better recommendations, and its potential to be generalized to trust-alike relationships. Hence, we can enhance the trust-unaware SVD++ model by incorporating both the explicit and implicit influence of trust, described as follows.

Implicit influence of trusted users. Fig. 2a shows that the trusted users of an active user have an effect on rating prediction for a certain item. We take into account this effect by modelling user preference in the same manner as rated items, given by:

$$\hat{r}_{u,j} = b_{u,j} + q_j^\top \Big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u^+|^{-\frac{1}{2}} \sum_{v \in T_u^+} w_v\Big),$$

where $b_{u,j} = b_u + b_j + \mu$ hereafter represents bias terms, w_v is the user-specific latent feature vector of users (trustees) trusted by user u, and thus $q_j^\top w_v$ can be explained by the ratings predicted by the trusted users, i.e., the influence of trustees on the rating prediction. In other words, the inner product $q_j^\top w_v$ indicates how trusted users influence user u's rating on item j. An intuitive understanding has been illustrated in Fig. 2a. Similar to ratings, a user's feature vector can be interpreted by the set of users whom she trusts, i.e., $|T_u^+|^{-\frac{1}{2}} \sum_{v \in T_u^+} w_v$. Therefore, a user u is further modelled by $\big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u^+|^{-\frac{1}{2}} \sum_{v \in T_u^+} w_v\big)$ in the social rating networks, considering the influence of both rated items and trusted users.

Implicit influence of trusting users. Fig. 2b shows that the trusting users of an active user can also influence the rating prediction for a certain item. In fact, *Observation 3* has indicated that such influence may be comparable with that of trusted users. Similarly, the effect can be considered by modelling user preference, given by:

$$\hat{r}_{u,j} = b_{u,j} + q_j^\top \Big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u^-|^{-\frac{1}{2}} \sum_{k \in T_u^-} p_k\Big),$$

where T_u^- is the set of users who trust user u, i.e., the set of her trusters. Thus, $q_j^\top p_k$ can be explained by the ratings predicted by the trusting users, i.e., the influence of trusters on the predictions. Similarly, the inner product $q_j^\top p_k$ indicates how trusting users k influence user u's rating on item j. An intuitive understanding has been illustrated in Fig. 2b. Similar to ratings, a user's feature vector can be interpreted by the set of users whom trust her, i.e., $|T_u^-|^{-\frac{1}{2}} \sum_{k \in T_u^-} p_k$. Therefore, a user u is further modelled by $\big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u^-|^{-\frac{1}{2}} \sum_{k \in T_u^-} p_k\big)$ in the social rating networks, considering the influence of both rated items and trusting users.

Combinational implicit trust influence. The implicit influence of trust neighbours on rating prediction therefore consists of two parts: the influence of both trustees and trusters. To consider both cases, we propose the following three fusion approaches.

(1) *Linear combination:* A natural and straightforward way is to linearly combine the two kinds of implicit trust influence, given by:

$$\begin{aligned}\hat{r}_{u,j} = b_{u,j} + q_j^\top \Big(&p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i \\ &+ \alpha |T_u^+|^{-\frac{1}{2}} \sum_{v \in T_u^+} w_v + (1-\alpha)|T_u^-|^{-\frac{1}{2}} \sum_{k \in T_u^-} p_k\Big),\end{aligned} \tag{1}$$

where $\alpha \in [0, 1]$ controls the importance of influence of trustees in rating prediction. Specifically, $\alpha = 0$ means that we only consider the influence of trusting users; $\alpha = 1$ indicates that only the influence of trusted users are considered; and $\alpha \in (0, 1)$ mixes the two kinds of trust influence together. In the case of undirected social relationships (e.g., friendship in Flixster), T_u^+ will be equivalent with T_u^-, and thus the linear combination ensures that our model can be applied to both trust and trust-alike relationships.

(2) *All as trusting users:* In a trust relationship, a user u can be represented either by p_u as trustor or by w_u as trustee. An alternative way is to model the influence of user u's trust neighbours, including both trusted and trusting users, in the manner of trusting users such that we can yield the following function:

$$\hat{r}_{u,j} = b_{u,j} + q_j^\top \Big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u|^{-\frac{1}{2}} \sum_{k \in T_u} p_k\Big), \tag{2}$$

where $T_u = T_u^+ \cup T_u^-$ denotes the set of user u's trust neighbours. The underlying assumption is not to distinguish the roles of trust neighbours, but to treat them uniformly in terms of implicit trust influence.

(3) *All as trusted users:* With the same assumption, we may model the influence of all trust neighbours in the manner of trusted users. That is, we predict the user's possible rating on a target item by:

$$\hat{r}_{u,j} = b_{u,j} + q_j^\top \Big(p_u + |I_u|^{-\frac{1}{2}} \sum_{i \in I_u} y_i + |T_u|^{-\frac{1}{2}} \sum_{v \in T_u} w_v\Big), \tag{3}$$

where T_u is the set of user u's trust neighbours. Both Equations (2) and (3) will also influence the decomposition of trust relationships. However, since user-feature matrix P plays a key role in bridging both rating and trust information, the rating prediction by equation (2) may lead to a better performance than that by equation (3).

With the consideration of implicit trust influence, the objective function to minimize is then given as follows:

$$\mathcal{L} = \frac{1}{2}\sum_u\sum_{j\in I_u}(\hat{r}_{u,j} - r_{u,j})^2 + \frac{\lambda}{2}\Bigg(\sum_u b_u^2 + \sum_j b_j^2 + \sum_u \|p_u\|_F^2 + \sum_j \|q_j\|_F^2 + \sum_i \|y_i\|_F^2 + \sum_v \|w_v\|_F^2\Bigg),$$

where $\hat{r}_{u,j}$ is the prediction computed by equation (1). To reduce the model complexity, we use the same regularization parameter λ for all the variables. Finer control and tuning can be achieved by assigning separate regularization parameters to different variables, though it may result in greater complexity in model learning, and in comparison with other matrix factorization models.

*Explicit trust influence.*In addition, as explained earlier, we constrain that the user-specific vectors decomposed from the rating matrix and those decomposed from the trust matrix share the same feature space in order to bridge both matrices together. In this way, these two types of information can be exploited in a unified recommendation model. Specifically, we can regularize the user-specific vectors p_u by recovering the social relationships with other users. The new objective function (without the other regularization terms) is given by:

$$\mathcal{L} = \frac{1}{2}\sum_u\sum_{j\in I_u}(\hat{r}_{u,j} - r_{u,j})^2 + \frac{\lambda_t}{2}\sum_u\Bigg(\alpha\sum_{v\in T_u^+}\big(\hat{t}_{u,v} - t_{u,v}\big)^2 + (1-\alpha)\sum_{k\in T_u^-}\big(\hat{t}_{k,u} - t_{k,u}\big)^2\Bigg),$$

where $\hat{t}_{u,v} = w_v^\top p_u$ is the predicted trust between users u and v computed by the inner product of truster and trustee vectors, i.e., $w_v^\top p_u$. Similarly, $\hat{t}_{k,u} = w_u^\top p_k$ is the predicted trust for user k towards user u, and λ_t controls the degree of trust regularization.

Adaptive regularization. As suggested by Yang et al. [8], a technique called weighted-λ-regularization can be used to help avoid over-fitting when learning model parameters. In particular, they consider more penalties for the users who rated more items and for the items which received more ratings. However, we argue that such consideration may force the model to be more biased towards popular users and items. Instead, in this work we adopt a distinct strategy that the popular users and items should be less penalized (due to smaller chance to be over-fitted), and cold-start users and *niche items* (those receiving few ratings) should be more regularized (due to greater chance to be over-fitted). As a result, the new loss function to minimize is obtained as follows:

$$\begin{aligned}
\mathcal{L} = {} & \frac{1}{2}\sum_u\sum_{j\in I_u}(\hat{r}_{u,j} - r_{u,j})^2 \\
& + \frac{\lambda_t}{2}\sum_u\Big(\alpha\sum_{v\in T_u^+}\big(\hat{t}_{u,v} - t_{u,v}\big)^2 + (1-\alpha)\sum_{k\in T_u^-}\big(\hat{t}_{k,u} - t_{k,u}\big)^2\Big) \\
& + \frac{\lambda}{2}\sum_u |I_u|^{-\frac{1}{2}} b_u^2 + \frac{\lambda}{2}\sum_j |U_j|^{-\frac{1}{2}} b_j^2 \\
& + \sum_u\Big(\frac{\lambda}{2}|I_u|^{-\frac{1}{2}} + \frac{\lambda_t}{2}\big(\delta(\alpha)|T_u^+|^{-\frac{1}{2}} + \delta(1-\alpha)|T_u^-|^{-\frac{1}{2}}\big)\Big)\|p_u\|_F^2 \\
& + \frac{\lambda}{2}\sum_j |U_j|^{-\frac{1}{2}}\|q_j\|_F^2 + \frac{\lambda}{2}\sum_i |U_i|^{-\frac{1}{2}}\|y_i\|_F^2 \\
& + \frac{\lambda}{2}\sum_u\sum_{v\in T_u^+}\delta(\alpha)|T_v^+|^{-\frac{1}{2}}\|w_v\|_F^2 \\
& + \frac{\lambda}{2}\sum_u\sum_{k\in T_u^-}\delta(1-\alpha)|T_k^-|^{-\frac{1}{2}}\|p_k\|_F^2,
\end{aligned} \tag{4}$$

where U_j, U_i are the set of users who rate items j and i, respectively; and $\delta(x)$ is an indicator function which equals 1 if $x > 0$, and 0 otherwise. Specifically, we multiply $|I_u|^{-\frac{1}{2}}$ (i.e., number of items rated) to variables related to users including p_u and b_u. The same holds for items' variables, namely q_j, q_i and b_j. Besides, since the active users may be socially connected with other trust neighbours, the penalization on user-specific vector p_u takes into account two cases: trusted by others $|T_u^-|$ and trusting other users $|T_u^+|$. Similarly, we take into account the number of trusted and trusting users for variables p_k and w_v, respectively.

4.3 Model Learning

To obtain a local minimization of the objective function given by equation (4), we perform the following gradient descents on $b_u, b_j, p_u, q_j, y_i, w_v$ and p_k across all the users and items in a training data set:

$$\begin{aligned}
\frac{\partial\mathcal{L}}{\partial b_u} &= \sum_{j\in I_u} e_{u,j} + \lambda|I_u|^{-\frac{1}{2}} b_u \\
\frac{\partial\mathcal{L}}{\partial b_j} &= \sum_{u\in U_j} e_{u,j} + \lambda|U_j|^{-\frac{1}{2}} b_j \\
\frac{\partial\mathcal{L}}{\partial p_u} &= \sum_{j\in I_u} e_{u,j} q_j + \lambda_t\alpha\sum_{v\in T_u^+} e_{u,v} w_v + \Big(\lambda|I_u|^{-\frac{1}{2}} \\
&\qquad + \lambda_t\big(\delta(\alpha)|T_u^+|^{-\frac{1}{2}} + \delta(1-\alpha)|T_u^-|^{-\frac{1}{2}}\big)\Big) p_u \\
\frac{\partial\mathcal{L}}{\partial q_j} &= \sum_{u\in U_j} e_{u,j}\Big(p_u + |I_u|^{-\frac{1}{2}}\sum_{i\in I_u} y_i + \alpha|T_u^+|^{-\frac{1}{2}}\sum_{v\in T_u^+} w_v \\
&\qquad + (1-\alpha)|T_u^-|^{-\frac{1}{2}}\sum_{k\in T_u^-} p_k\Big) + \lambda|U_j|^{-\frac{1}{2}} q_j \\
\forall i\in I_u, \frac{\partial\mathcal{L}}{\partial y_i} &= \sum_{j\in I_u} e_{u,j}|I_u|^{-\frac{1}{2}} q_j + \lambda|U_i|^{-\frac{1}{2}} y_i \\
\forall v\in T_u^+, \frac{\partial\mathcal{L}}{\partial w_v} &= \sum_{j\in I_u} e_{u,j}\alpha|T_u^+|^{-\frac{1}{2}} q_j + \lambda_t\alpha e_{u,v} p_u \\
&\qquad + \lambda\delta(\alpha)|T_v^+|^{-\frac{1}{2}} w_v \\
\forall k\in T_u^-, \frac{\partial\mathcal{L}}{\partial p_k} &= \sum_{j\in I_u} e_{u,j}(1-\alpha)|T_u^-|^{-\frac{1}{2}} q_j \\
&\qquad + \lambda_t(1-\alpha) e_{k,u} w_u + \lambda\delta(1-\alpha)|T_k^-|^{-\frac{1}{2}} p_k,
\end{aligned} \tag{5}$$

where $e_{u,j} = \hat{r}_{u,j} - r_{u,j}$ indicates the rating prediction error for user u on item j, and $e_{u,v} = \hat{t}_{u,v} - t_{u,v}$ is the trust prediction error for user u towards trustee v as well as $e_{k,u} = \hat{t}_{k,u} - t_{k,u}$ for truster k towards user u.

The pseudocode for model learning is given in Algorithm 1. To explain, several arguments are taken as input, including user-item rating matrix R, user-user trust matrix T, regularization parameters λ and λ_t, and the initial learning rate γ. First, we randomly initialize the decomposed vectors and matrices with small values (line 1). Then, we keep training the model until the loss function is converged (line 2). Specifically, we compute variable gradients according to equation (5) (line 3), and then update variables by the gradient descent method (lines 4–10). Finally, we return the learned vectors and matrices as output (line 11).

Algorithm 1. Learning in the TrustSVD Model

Input: $R, T, d, \lambda, \lambda_t, \gamma$ (learning rate)
Output: Rating predictions $\hat{r}_{u,j}$
1 Initialize vectors B_u, B_j and matrices P, Q, Y, W with small and random values in $(0, 1)$;
2 **while** $\mathcal{L}$ *not converged* **do**
3 compute gradients according to Equation (5);
4 $b_u \leftarrow b_u - \gamma \frac{\partial \mathcal{L}}{\partial b_u}, u = 1 \dots m$
5 $b_j \leftarrow b_j - \gamma \frac{\partial \mathcal{L}}{\partial b_j}, j = 1 \dots n$
6 $p_u \leftarrow p_u - \gamma \frac{\partial \mathcal{L}}{\partial p_u}, u = 1 \dots m$
7 $q_j \leftarrow q_j - \gamma \frac{\partial \mathcal{L}}{\partial q_j}, j = 1 \dots n$
8 $\forall i \in I_u, \quad y_i \leftarrow y_i - \gamma \frac{\partial \mathcal{L}}{\partial y_i}, u = 1 \dots m$
9 $\forall v \in T_u^+, w_v \leftarrow w_v - \gamma \frac{\partial \mathcal{L}}{\partial w_v}, u = 1 \dots m$
10 $\forall k \in T_u^-, p_k \leftarrow p_k - \gamma \frac{\partial \mathcal{L}}{\partial p_k}, u = 1 \dots m$
11 return B_u, B_j, P, Q, Y, W;

4.4 Complexity Analysis

The computational time of learning the TrustSVD model is mainly taken by evaluating the objective function $\mathcal{L}$ and its gradients against feature vectors (variables). The time to compute the objective function $\mathcal{L}$ is $O(d|R| + d|T|)$, where d is the dimensionality of the feature space, and $|R|, |T|$ refer to the number of observed entries. Due to the sparsity of rating and trust matrices, the values will be much smaller than the matrix cardinality. The computational complexities for gradients $\frac{\partial \mathcal{L}}{\partial b_u}, \frac{\partial \mathcal{L}}{\partial b_j}, \frac{\partial \mathcal{L}}{\partial p_u}, \frac{\partial \mathcal{L}}{\partial q_j}, \frac{\partial \mathcal{L}}{\partial y_i}, \frac{\partial \mathcal{L}}{\partial w_v}, \frac{\partial \mathcal{L}}{\partial p_k}$ in equation (5) are $O(d|R|), O(d|R|), O(d|R| + d|T|), O(d|R| + d|T|), O(d|R|k)$, $O(d|R|p^+ + d|T|p^+)$ and $O(d|R|p^- + d|T|p^-)$, where k, p^+, p^- are the average number of ratings received by an item, trust statements given and received by a user, respectively. Hence, the overall computational complexity in one iteration is $O(d|R|c + d|T|c)$, where $c = \max(p^+, p^-, k)$. Due to $c \ll |R|$ or $|T|$, the overall computational time is linear with respect to the number of observations in the rating and trust matrices. It follows that our model has potential to scale up to large-scale data sets. Section 5.7 will investigate the scalability of our approach in four real-world data sets.

4.5 Insights into the TrustSVD Model

The key idea behind the TrustSVD model is to take into account both explicit and implicit influences of item ratings and of social trust information when predicting users' ratings for unknown items. Specifically, for ratings, the explicit information is the rating values which are approximated by the inner product $q_j^\top p_u$ of user- and item-specific latent feature vectors, while the implicit influence is represented by $q_j^\top y_i$ regarding the effect of rated items by the active users. Similarly, for trust statements, the explicit information is the trust values that are predicted by the inner product $w_v^\top p_u$ of truster- and trustee-specific latent feature vectors. To bridge the rating and trust matrices together, we limit the user/truster vector to be the same p_u. The implicit influence of trust neighbours can be further split into two parts: trustees' influence is modelled by the inner product $q_j^\top w_v$ while trusters' influence is given by the inner product $q_j^\top p_k$. Since the state-of-the-art model SVD++ naturally incorporates the implicit influence of item ratings, we build on top of this model by further incorporating the implicit influence of trust neighbours as well as the explicit one. Therefore, compared with other models, TrustSVD enables more information (both implicit and explicit) for rating prediction, resulting in better recommendation performance.

In the cold-start situations where users may have only rated a few items, the decomposition of trust matrix can help to learn more reliable user-specific latent feature vectors than ratings-only matrix factorization. In the extreme case where there are no ratings at all for some users, equation (5) ensures that the user-specific vector can be learned from the trust matrix. In this regard, incorporating trust in a matrix factorization model can alleviate the cold start problem. By considering both explicit and implicit influence of trust rather than either one, our model can better utilize trust to further mitigate the data sparsity and cold start issues.

5 Evaluation

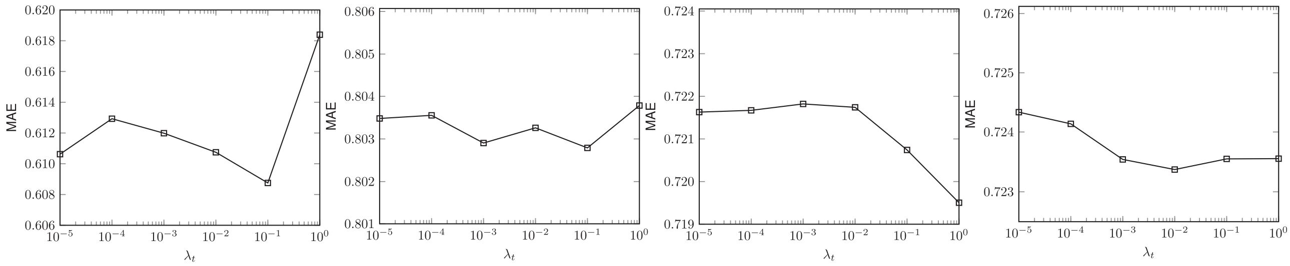
In this section we conduct a series of experiments in order to investigate the effectiveness of our approach in comparison with other counterparts across the four data sets of Table 1.

5.1 Experimental Settings

Testing views. Two data set views are created for testing. First, the *All* view indicates that all ratings are used as the test set. Second, the *Cold Start* view means that only the users who rate less than five items will be involved in the test set. Similar testing views are also defined and used in [4], [8]. We use five-fold cross-validation for learning and testing. Specifically, we randomly split each data set into five folds and in each iteration four folds are used as the training set and the remaining fold as the test set. Five iterations are conducted to ensure that all folds are tested. Then, the average test performance is given as the final result. In the *Cold Start* view, five-fold cross validation is still used but we only care about the performance for cold-start users.

Evaluation metrics. We adopt two well-known metrics to evaluate predictive accuracy, namely mean absolute error (MAE) and root mean square error, defined by:

$$\mathrm{MAE} = \frac{\sum_{u,j} |\hat{r}_{u,j} - r_{u,j}|}{N}, \ \mathrm{RMSE} = \sqrt{\frac{\sum_{u,j} (\hat{r}_{u,j} - r_{u,j})^2}{N}},$$



(a) FilmTrust (b) Epinions (c) Flixster (d) Ciao

Fig. 3. The effect of parameter trust regularization λ_t across all the data sets [$d = 10$].

where N is the number of test ratings. Smaller values of MAE and RMSE indicate better predictive accuracy. Since RMSE puts relatively high weights on large errors and all comparison models (except the baselines) adopt the least square errors as loss function, RMSE is more appropriate than MAE to measure the predictive performance for our work. In addition, the larger the difference between MAE and RMSE, the greater the variance of predictive errors.

Comparison methods. Up to ten recommendation models are compared with **TrustSVD** in our experiments, including: (1) **UAvg**, **IAvg** are baselines that predict a user's rating by the average of her historical ratings, and of ratings received by the target item, respectively; (2) **PMF** is a basic probabilistic matrix factorization model proposed by Salakhutdinov and Mnih [22]; (3) **RSTE** [23], **SoRec** [5] and **SoReg** [6] are earlier trust-based recommendation models by Ma et al.; (4) **SocialMF** [7], **TrustMF** [8], **Fang's** [9] are the latest and state-of-the-art trust-based models that are reported to achieve better performance than simple baselines and other counterparts [8], [9]; (4) **SVD++** [12] is a state-of-the-art recommendation method merely based on ratings, and also adopted as a key comparison method in Fang et al. [9].

Parameter settings. The optimal experimental settings for all the models are determined either by our experiments or suggested by previous works. Specifically, the common settings are $\lambda = 0.001$, and the number of latent features $d = 5/10$, the same as all the previous trust-based models. The other settings are: (1) RSTE: $\alpha = 0.4$ for Epinions, and 1.0 for the others; (2) SoRec: $\lambda_c = 0.1, 1.0, 0.001, 0.01$ corresponding to FilmTrust, Epinions, Flixster and Ciao, respectively; (3) SoReg: $\beta = 1.0$ for Flixster and 0.1 for the others; (4) SocialMF, TrustMF: $\lambda_t = 1$; (5) SVD++: $\lambda = 0.1, 0.35, 0.03, 0.1$ (resp.); (6) TrustSVD: $\lambda = 0.6$ for FilmTrust, $\lambda = 1$ for Epinions, $\lambda = 0.6$ for Flixster, and $\lambda = 0.1$ for Ciao. Parameters λ_t, α will be elaborated next.

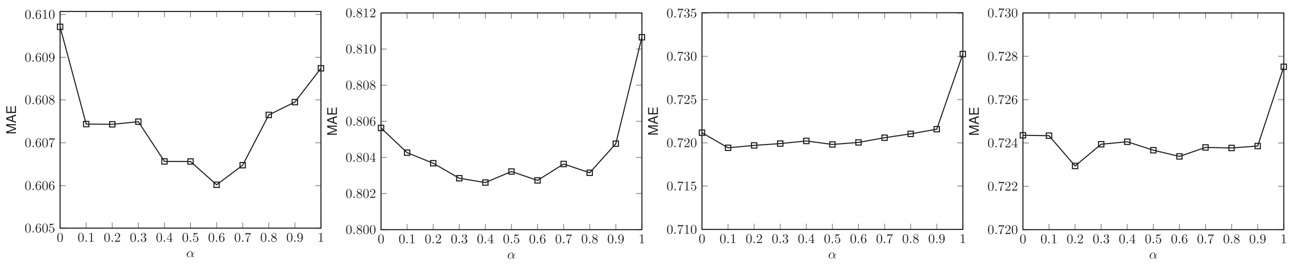
5.2 Impact of Parameters λ_t and α

Other than the parameter λ, two more parameters are used in our method, namely λ_t for the importance of trust regularization and α for the relative importance of influence of trustees. To determine their values for different data sets, we first fix the value of one parameter, and then adjust the values of the other to search the best parameter settings. Specifically, we tune the parameter λ_t in the range $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$ while fixing $\alpha = 0.5$, i.e., equal importance of both influence of trustees and trusters. The results are illustrated in Fig. 3 in terms of MAE when $d = 10$. The other settings (e.g., $d = 5$ or in terms of RMSE) show similar performance trends. MAE is adopted since it gives clearer changes of recommendation performance. The results clearly indicate that a proper value of λ_t for different data sets can help improve the recommendation performance. Generally, a value 0.1 would give relatively fair performance if necessary.

After determining λ_t's values, we proceed to tune the value of parameter α in [0, 1] with step 0.1. Parameter $\alpha = 1$ indicates that only the influence of trustees is taken into account while $\alpha = 0$ means that only the influence of trusters is considered in our method. The results are presented in Fig. 4. We observe that the performance of the extreme value $\alpha = 0$ is inferior to $\alpha = 1$ in FilmTrust, but performs better in the other data sets. In this respect, the impact is domain-specific. Nevertheless, the performance of $\alpha = 0$ and $\alpha = 1$ is much worse than the performance of other values. In other words, a proper combination of both the influence of trusters and trustees leads to better recommendation performance. Although the best value of α that reaches the superior performance may vary in different data sets, a value of 0.6 in general is a fair setting.

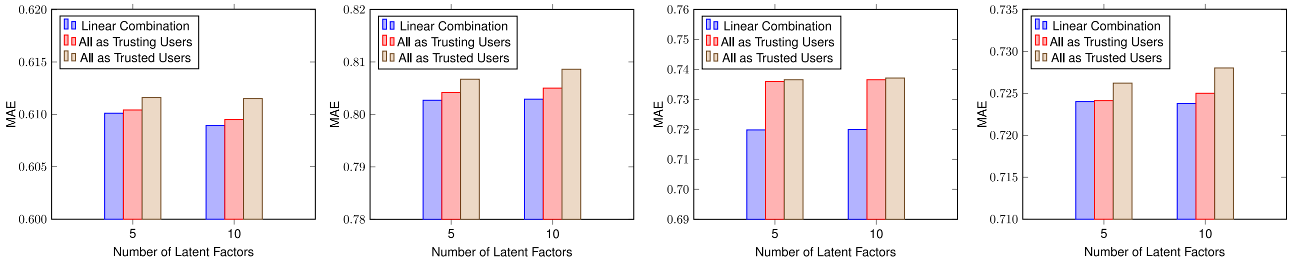
5.3 Comparison of Trust Influence Combination

We compared the different ways to combine the influence of trust influence. The results are illustrated in Fig. 5. The



(a) FilmTrust (b) Epinions (c) Flixster (d) Ciao

Fig. 4. The effect of parameter trustee's importance α across all the data sets [$d = 10$].



(a) FilmTrust (b) Epinions (c) Flixster (d) Ciao

Fig. 5. The effect of different ways to combine the implicit influence of both trusting and trusted users.

results show that linear combination consistently achieves better accuracy than *All as Trusting Users* which in turn outperforms *All as Trusted Users*. We infer that: (1) it is better to distinguish the role of trusting and trusted users; and (2) modelling all social neighbours as trusting users is more effective than as trusted users, since user vector p_u functions as a pivot in bridging both rating and trust information.

5.4 Impact of Adaptive Regularization

Adaptive regularization is adopted in equation (4) to help avoid over-fitting. We study its impact on predictive accuracy in comparison with the strategy of traditional L2-regularization. The best parameter settings from the previous analysis are taken in this part. The results on the four data sets are presented in Table 2. Generally, the average improvements our adaptive regularization achieves relative to traditional regularization are around 0.00275 in MAE and 0.00475 in RMSE, demonstrating the usefulness of finer-grained regularization techniques.

5.5 Comparison with Other Models

The experimental results are presented in Tables 3 and 4, corresponding to the testing views of *All* and *Cold Start*, respectively. For all the comparison methods in the testing view of *All*, SVD++ outperforms the other comparison methods in FilmTrust and Epinions, and UAvg performs the best in Flixster. This implies that these trust-based approaches cannot always beat other well-performing ratings-only approaches, and even simple baselines in trust-alike networks (i.e., FilmTrust and Flixster). Only in Ciao, trust-based approach (SocialMF) gives the best performance. Hence, previous trust-based approaches cannot always provide superior accuracy than ratings-only counterparts. On the contrary, our approach TrustSVD is consistently superior to the best approach among the others across all the data sets. Although the percentage of relative improvements is small (around 3.17 percent in RMSE on the average), Koren [11] has pointed out that even small improvements in MAE and RMSE may lead to significant differences of recommendations in practice. For example, the Netflix prize (netflixprize.com) competition offered USD $1M for 10 percent improvements in RMSE. Note that among all the trust-based approaches, SoReg is the only one method that achives significant improvements when tuning the number of latent factors from 5 to 10. This is due to no trust decomposition in this model. To have a fairer comparison, we compare the best performance of SoReg (by tuning $d \in [5, 50]$ with step 5) with TrustSVD (see Table 5). The results also verify the effectiveness of our approach.

For all the comparison methods in the testing view of *Cold Start*, SoRec and SVD++ perform respectively the best in FilmTrust and Flixster (trust-alike), while no single approach works the best in Epinions and Ciao (trust). Generally, our approach performs better than the others both in trust and trust-alike relationships. Although some exceptions are observed in Epinions in MAE, TrustSVD is more powerful in RMSE. Since all the trust-based models aim to optimize the square errors between predictions and real values, RMSE is more indicative than MAE, and thus TrustSVD still has the best performance overall. As discussed in Section 4.5, it is essential for our model to handle the data sparsity and cold start by considering both the explicit and implicit influence of ratings and trust.

Besides the above-compared approaches, some new trust-based models have been proposed recently. The most relevant model is presented by Fang et al. [9]; for clarity, we denote it by *Fang's*. It is reported to perform better than other trust-based models and than SVD++ (except in Ciao). Table 6 shows the comparison between Fang's and our approach TrustSVD. The results of Fang's approach are reported in [9],[8] and directly re-used in our article. Note that we sampled more data for Flixster than Fang's, and thus their experimental results are not comparable. Table 6 clearly shows that our approach performs better than Fang's in terms of both MAE and RMSE.

One more observation from Tables 3, 4 and 6 is that the performance of TrustSVD when $d = 5$ is very close to that when $d = 10$, indicating the reliability of our approach with respect to the feature dimensionality. We ascribe this feature to the consideration of both the explicit and implicit influence of item ratings and social trust in a unified recommendation model.

TABLE 2
Comparison of the Traditional L2 and Our Adaptive (in Bold) Regularization in the Testing View of 'All', Where the Top Rows Are in MAE and the Bottom in RMSE

FilmTrust		Epinions		Flixster		Ciao	
$d = 5$	$d = 10$	$d = 5$	$d = 10$	$d = 5$	$d = 10$	$d = 5$	$d = 10$
0.615	0.613	0.804	0.805	0.721	0.723	0.726	0.727
0.611	**0.609**	**0.802**	**0.803**	**0.720**	**0.720**	**0.723**	**0.724**
0.799	0.797	1.050	1.052	0.948	0.949	0.960	0.964
0.794	**0.792**	**1.048**	**1.049**	**0.943**	**0.942**	**0.956**	**0.957**

8. Since only the results in 'All' are reported and no results reported in the case of *Cold Start*, we do not merge them in Tables 3 and 4.

TABLE 3
Performance Comparison in the Testing View of 'All', Where * Indicates the Best Performance among All the Other Methods, and Column 'Improve' Indicates the Percentage of Improvements that TrustSVD Achieves Relative to the *Results

All	Metrics	UAvg	IAvg	PMF	RSTE	SoRec	SoReg	SocialMF	TrustMF	SVD++	TrustSVD	Improve
FilmTrust	MAE	0.636	0.725	0.714	0.628	0.628	0.661	0.638	0.631	**0.613***	**0.607**	0.98%
$d = 5$	RMSE	0.823	0.927	0.949	0.810	0.810	0.866	0.837	0.810	**0.804***	**0.791**	1.62%
	MAE	0.636	0.725	0.735	0.640	0.638	0.644	0.642	0.631	**0.611***	**0.605**	0.98%
$d = 10$	RMSE	0.823	0.927	0.968	0.835	0.831	0.838	0.844	0.819	**0.802***	**0.789**	1.62%
Epinions	MAE	0.930	0.928	0.979	0.950	0.882	0.996	0.825	0.818	**0.818***	**0.803**	1.83%
$d = 5$	RMSE	1.203	1.094	1.290	1.196	1.114	1.304	1.070	1.069	**1.057***	**1.043**	1.32%
	MAE	0.930	0.928	0.909	0.958	0.884	0.906	0.826	0.819	**0.818***	**0.803**	1.83%
$d = 10$	RMSE	1.203	1.094	1.197	1.278	1.142	1.182	1.082	1.095	**1.057***	**1.044**	1.23%
Flixster	MAE	**0.729***	0.858	0.814	0.751	0.750	0.825	0.770	0.890	0.794	**0.719**	1.37%
$d = 5$	RMSE	**0.979***	1.088	1.076	0.975	0.974	1.088	0.994	1.146	1.062	**0.942**	3.88%
	MAE	**0.729***	0.858	0.769	0.784	0.785	0.774	0.784	1.116	0.821	**0.719**	1.37%
$d = 10$	RMSE	**0.979***	1.088	1.009	1.015	1.018	1.016	1.009	1.441	1.091	**0.941**	3.88%
Ciao	MAE	0.781	0.760	0.920	0.767	0.765	0.899	0.749	**0.742***	0.752	**0.723**	2.56%
$d = 5$	RMSE	1.031	1.026	1.206	1.020	1.013	1.183	**0.981***	0.983	1.013	**0.956**	2.55%
	MAE	0.781	0.760	0.822	0.763	0.761	0.812	**0.749***	0.753	0.748	**0.723**	3.47%
$d = 10$	RMSE	1.031	1.026	1.078	1.013	1.010	1.073	**0.976***	1.014	1.001	**0.956**	2.05%

TABLE 4
Performance Comparison in the Testing View of 'Cold Start'

Cold Start	Metrics	UAvg	IAvg	PMF	RSTE	SoRec	SoReg	SocialMF	TrustMF	SVD++	TrustSVD	Improve
FilmTrust	MAE	0.709	0.722	0.814	0.680	**0.670***	0.881	0.697	0.674	0.677	**0.655**	2.24%
$d = 5$	RMSE	0.979	0.911	1.079	0.884	**0.857***	1.104	0.916	0.867	0.897	**0.839**	2.10%
	MAE	0.709	0.722	0.767	0.674	**0.668***	0.771	0.680	0.687	0.680	**0.659**	1.35%
$d = 10$	RMSE	0.979	0.911	1.009	0.900	**0.897***	1.034	0.907	0.900	0.905	**0.847**	5.57%
Epinions	MAE	1.047	**0.852***	1.451	1.051	0.892	1.398	0.884	0.891	0.889	**0.869**	-1.99%
$d = 5$	RMSE	1.430	1.127	1.770	1.266	1.138	1.735	1.133	**1.125***	1.162	**1.104**	1.87%
	MAE	1.047	0.852	1.153	0.981	**0.846***	1.139	0.857	0.853	0.891	**0.868**	-2.60%
$d = 10$	RMSE	1.430	**1.127***	1.432	1.313	1.180	1.437	1.152	1.176	1.166	**1.105**	1.95%
Flixster	MAE	0.869	0.906	1.097	0.872	0.872	1.058	0.881	0.901	**0.868***	**0.844**	2.76%
$d = 5$	RMSE	1.155	1.114	1.390	1.097	**1.096***	1.358	1.103	1.138	1.122	**1.056**	3.65%
	MAE	**0.869***	0.906	0.949	0.889	0.892	0.951	0.884	0.976	**0.869***	**0.846**	2.65%
$d = 10$	RMSE	1.155	1.114	1.206	1.137	1.144	1.218	1.112*	1.328	**1.112***	**1.059**	4.77%
Ciao	MAE	0.829	**0.735***	1.033	0.957	0.789	1.173	0.774	0.752	0.759	**0.726**	1.22%
$d = 5$	RMSE	1.138	1.005	1.334	1.113	0.998	1.430	1.001	**0.954***	1.039	**0.940**	1.47%
	MAE	0.829	0.735	0.926	0.803	**0.730***	0.949	0.741	0.770	0.749	**0.725**	0.68%
$d = 10$	RMSE	1.138	1.005	1.191	1.014	1.031	1.214	**0.978***	1.096	1.020	**0.939**	3.99%

In conclusion, the experimental results indicate that our approach TrustSVD outperforms the other methods in predicting more accurate ratings, and that its performance is reliable with different numbers of latent features.

5.6 Comparison in Trust Degrees

Another series of experiments are conducted to investigate the performance on users with different trust degrees, in order to further compare the performance of our approach with other trust-based counterparts, i.e., RSTE, SoRec, SoReg, SocialMF and TrustMF.[9] The trust degrees refer to the summation of the number of trusted neighbours specified by a user (i.e., out degree) and the number of trusting neighbours who trust the user (i.e., in degree). We split the trust degrees into (up to seven) categories: 1-5, 6-10, 11-20, 21-40, 41-100, 101-500, >500 as used by Yang et al. [8]. The results of trust-based models are illustrated in Fig. 6 in the

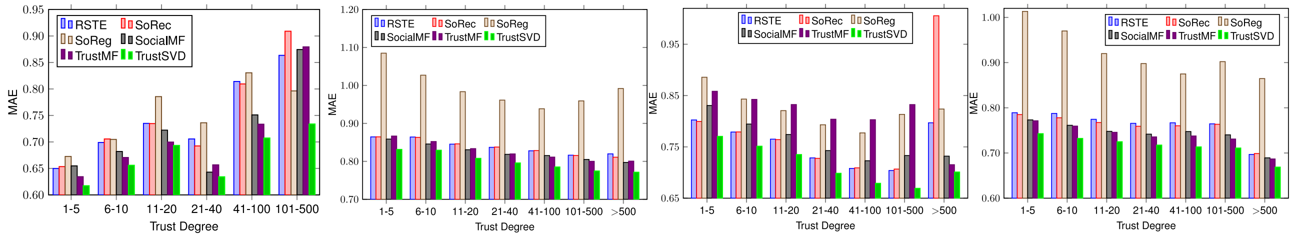
9. Other trust-unaware methods (e.g., SVD++, PMF) are not used in the experiments.

TABLE 5
Comparing with the Best Performance of SoReg

	SoReg	TrustSVD		SoReg	TrustSVD
FilmTrust	0.637	**0.607**	**Epinions**	0.846	**0.803**
$d = 15$	0.834	**0.791**	$d = 20$	1.130	**1.043**
Flixster	0.774	**0.719**	**Ciao**	0.776	**0.719**
$d = 10$	1.016	**0.942**	$d = 20$	1.044	**0.941**

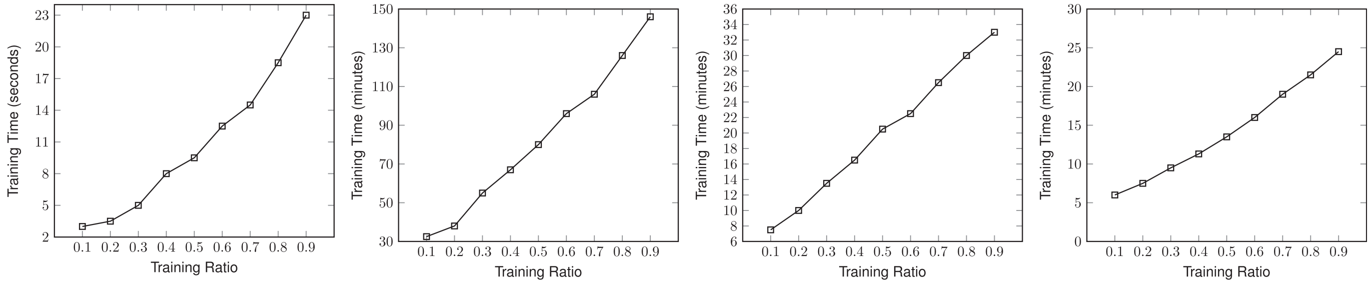
TABLE 6
Performance Comparing with Fang's Approach

Fang's vs.	Epinions		Ciao		FilmTrust	
TrustSVD	$d = 5$	$d = 10$	$d = 5$	$d = 10$	$d = 5$	$d = 10$
MAE	0.806	0.814	0.737	0.745	0.616	0.625
	0.804	**0.805**	**0.723**	**0.723**	**0.607**	**0.605**
RMSE	1.047	1.059	0.972	0.985	0.793	0.810
	1.043	**1.044**	**0.956**	**0.956**	**0.791**	**0.789**



(a) FilmTrust (b) Epinions (c) Flixster (d) Ciao

Fig. 6. Performance comparison on users with different trust degrees across data sets ($d = 5$) [best viewed in color].



(a) FilmTrust (b) Epinions (c) Flixster (d) Ciao

Fig. 7. The scalability of our approach across all the data sets [$d = 10$], where (a) is in seconds and the others are in minutes.

case of $d = 5$ in MAE. It is observed that our approach TrustSVD consistently outperforms the other trust-based models in terms of both MAE and RMSE across all the data sets. The statistic significance tests (paired t-tests, confidence 0.95) between our approach TrustSVD and other comparison models are conducted,[10] and show that our approach TrustSVD in general achieves statistically significant performance relative to other methods. It is noted that the predictive accuracy on FilmTrust decreases along with the increment of trust degrees. One possible explanation is that much noise is arose from converting real-valued trust to binary trust. Thus, the social relationships may be less useful in FilmTrust than in the others as indicated by Fang et al. [9].

5.7 Scalability

We proceed to investigate the scalability of TrustSVD in terms of training time when being applied to different percentages of data sets. Specifically, we vary the percentages from 0.1 to 1 stepping by 0.1 in each data set. The results, illustrated in Fig. 7, show that the training time increases linearly with the amount of training data. Hence, our approach can be applied to large-scale data sets.

6 Conclusion and Future Work

This article proposed a novel trust-based matrix factorization model which incorporated both rating and trust information. Our analysis of trust in four real-world data sets indicated that trust and ratings were complementary to each other, and both pivotal for more accurate recommendations. Our novel approach, TrustSVD, takes into account both the explicit and implicit influence of ratings and of trust information when predicting ratings of unknown items. Both the trust influence of trustees and trusters of active users are involved in our model. In addition, a weighted-λ-regularization technique is adapted and employed to further regularize the generation of user- and item-specific latent feature vectors. Computational complexity of TrustSVD indicated its capability of scaling up to large-scale data sets. Comprehensive experimental results on the four real-world data sets showed that our approach TrustSVD outperformed both trust- and ratings-based methods (ten models in total) in predictive accuracy across different testing views and across users with different trust degrees. We concluded that our approach can better alleviate the data sparsity and cold start problems of recommender systems.

As a rating prediction model, TrustSVD works well by incorporating trust influence. However, the literature has shown that models for rating prediction cannot suit the task of top-N item recommendation. For future work, we intend to study how trust can influence the ranking score of an item (both explicitly and implicitly). The ranking order between a rated item and an unrated item (but rated by trust users) may be critical to learn users' ranking patterns.

10. The details of p-values are omitted due to space limitation.

Acknowledgments

The authors thank the anonymous reviewers, the support of the MoE AcRF Tier 2 Grant M4020110.020, and the National Natural Science Foundation of China Grant 61402097. Neil Yorke-Smith thanks the Operations group at the Cambridge Judge Business School and the fellowship at St Edmund's College, Cambridge.

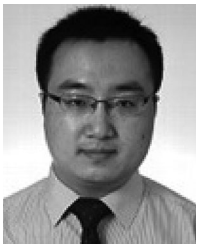
References

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Know. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[2] Y. Huang, X. Chen, J. Zhang, D. Zeng, D. Zhang, and X. Ding, "Single-trial ERPs denoising via collaborative filtering on ERPs images," *Neurocomputing*, vol. 149, pp. 914–923, 2015.

[3] X. Luo, Z. Ming, Z. You, S. Li, Y. Xia, and H. Leung, "Improving network topology-based protein interactome mapping via collaborative filtering," *Knowl.-Based Syst.*, vol. 90, pp. 23–32, 2015.
[4] G. Guo, J. Zhang, and D. Thalmann, "A simple but effective method to incorporate trusted neighbors in recommender systems," in *Proc. 20th Int. Conf. User Model., Adaptation Personalization*, 2012, pp. 114–125.
[5] H. Ma, H. Yang, M. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 31st Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval*, 2008, pp. 931–940.
[6] H. Ma, D. Zhou, C. Liu, M. Lyu, and I. King, "Recommender systems with social regularization," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 287–296.
[7] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. 4th ACM Conf. Recommender Syst.*, 2010, pp. 135–142.
[8] B. Yang, Y. Lei, D. Liu, and J. Liu, "Social collaborative filtering by trust," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 2747–2753.
[9] H. Fang, Y. Bao, and J. Zhang, "Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 30–36.
[10] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[11] Y. Koren, "Factor in the neighbors: Scalable and accurate collaborative filtering," *ACM Trans. Know. Discovery Data*, vol. 4, no. 1, pp. 1:1–1:24, 2010.
[12] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Know. Discovery Data Mining*, 2008, pp. 426–434.
[13] G. Guo, J. Zhang, and N. Yorke-Smith, "TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 123–129.
[14] R. Forsati, M. Mahdavi, M. Shamsfard, and M. Sarwat, "Matrix factorization with explicit trust and distrust side information for improved social recommendation," *ACM Trans. Inform. Syst.*, vol. 32, no. 4, pp. 17:1–17:38, 2014.
[15] G. Guo, J. Zhang, and N. Yorke-Smith, "Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems," *Know.-Based Syst.*, vol. 74, pp. 14–27, 2015.
[16] W. Yuan, D. Guan, Y. Lee, S. Lee, and S. Hur, "Improved trust-aware recommender system using small-worldness of trust networks," *Know.-Based Syst.*, vol. 23, no. 3, pp. 232–238, 2010.
[17] W. Yao, J. He, G. Huang, and Y. Zhang, "Modeling dual role preferences for Trust-aware recommendation," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval*, 2014, pp. 975–978.
[18] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Proc. 1st ACM Conf. Recommender Syst.*, 2007, pp. 17–24.
[19] J. Golbeck, "Generating predictive movie recommendations from trust in social networks," in *Proc. 4th Int. Conf. Trust Manage.*, 2006, pp. 93–104.
[20] M. Jamali, T. Huang, and M. Ester, "A generalized stochastic block model for recommendation in social rating networks," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 53–60.
[21] Y. Shi, P. Serdyukov, A. Hanjalic, and M. Larson, "Nontrivial landmark recommendation using geotagged photos," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 3, pp. 47:1–47:27, 2013.
[22] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inform. Process. Syst.*, 2008, vol. 20, pp. 1257–1264.
[23] H. Ma, I. King, and M. Lyu, "Learning to recommend with social trust ensemble," in *Proc. 32nd Int. ACM SIGIR Conf. Res. Development Inform. Retrieval*, 2009, pp. 203–210.
[24] X. Yang, H. Steck, and Y. Liu, "Circle-based recommendation in online social networks," in *Proc. 18th ACM SIGKDD Int. Conf. Know. Discovery Data Mining*, 2012, pp. 1267–1275.
[25] J. Zhu, H. Ma, C. Chen, and J. Bu, "Social recommendation using Low-rank semidefinite program," in *Proc. 26th AAAI Conf. Artif. Intell.*, 2011, pp. 158–163.
[26] J. Tang, X. Hu, H. Gao, and H. Liu, "Exploiting local and global social context for recommendation," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 2712–2718.
[27] M. Jamali and M. Ester, "Trustwalker: A random walk model for combining trust-based and item-based recommendation," in *Proc. 15th ACM SIGKDD Int. Conf. Know. Discovery Data Mining*, 2009, pp. 397–406.
[28] Q. Yuan, L. Chen, and S. Zhao, "Factorization vs. regularization: Fusing heterogeneous social relationships in Top-n recommendation," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 245–252.
[29] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Conf. Uncertainty Artif. Intell.*, 2009, pp. 452–461.
[30] T. Zhao, J. McAuley, and I. King, "Leveraging social connections to improve personalized ranking for collaborative filtering," in *Proc. 23rd ACM Int. Conf. Inform. Know. Manage.*, 2014, pp. 261–270.
[31] W. Pan, H. Zhong, C. Xu, and Z. Ming, "Adaptive bayesian personalized ranking for heterogeneous implicit feedbacks," *Know.-Based Syst.*, vol. 73, pp. 173–180, 2015.
[32] X. Yang, H. Steck, Y. Guo, and Y. Liu, "On Top-k recommendation using social networks," in *Proc. 6th ACM Conf. Recommender Syst.*, 2012, pp. 67–74.
[33] N. Barbieri, G. Manco, and E. Ritacco, *Probabilistic Approaches to Recommendations*. San Rafael, CA, USA: Morgan & Claypool, 2014.
[34] G. Guo, "Integrating trust and similarity to ameliorate the data sparsity and cold start for recommender systems," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 451-454.
[35] C. Castelfranchi and R. Falcone, *Trust Theory: A Socio-Cognitive and Computational Model*. Chichester, U.K.: Wiley, 2010.
[36] H. Ma, "On measuring social friend interest similarities in recommender systems," in *Proc. 37th Int. ACM SIGIR Conf. Res. Development Inform. Retrieval*, 2014, pp. 465–474.

Guibing Guo received the PhD degree in 2015 from the Nanyang Technological University, Singapore. He is an associate professor in the Software College, Northeastern University, China. His research is to resolve the data sparsity and cold start problems of recommender systems by: (1) incorporating social trust; (2) better utilizing exiting user ratings; (3) proposing the concept of prior ratings to elicit user ratings; and (4) developing a Java library for recommender systems—LibRec: www.librec.net.



Jie Zhang received the PhD degree from the University of Waterloo, Canada, in 2009. During the PhD degree study, he held the prestigious NSERC Alexander Graham Bell Canada Graduate Scholarship. He is currently an associate professor in the School of Computer Engineering, Nanyang Technological University, Singapore. He is also an academic fellow at the Institute of Asian Consumer Insight and Associate of the Singapore Institute of Manufacturing Technology (SIMTech).

Neil Yorke-Smith received the doctorate degree from the Imperial College London, United Kingdom. He is an associate professor of business information and decision systems in the Suliman S. Olayan School of Business, American University of Beirut, Lebanon. His research interests include planning and scheduling, agent-based methodologies, machine learning and data analytics, simulation, and their real-world applications.