Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 148 (2019) 181–190
Procedia Computer Science
www.elsevier.com/locate/procedia

Second International Conference on Intelligent Computing in Data Sciences (ICDS 2018)

An intelligent recommender system using social trust path for recommendations in web-based social networks

Imane Belkhadir*, Elamine Didi Omar, Jaouad Boumhidi

LIIAN Laboratory, Computer Science Department, Sidi Mohamed Ben AbdEllah University, Faculty of Sciences Dhar-Mahraz, Fez 30000, Morocco

Abstract

In this paper, we combine a social regularization approach that incorporates social network information to benefit recommender systems with the trust information between users. Both trust and rating records (tags) are employed to predict the missing values (tags) in the user-item matrix. Especially, we use an algorithm for best recommended trust path selection, to identify multiple recommended trust paths and to determine an aggregate path for generating different final recommendations. Empirical analyses on real datasets show that the combination of social information and trust achieves superior performance to existing approaches.

Keywords: Collaborative filtering, Big data, recommender system, Social network, Trust ;

1. Introduction

In the last years, the volume of available data on the social networks has exploded. In order to overcome information overload, recommender systems have become a key tool for providing users with personalized recommendations on

* Corresponding author. Tel.: +212-667-892-539.
E-mail address: belk.imanee@gmail.com

1877-0509 © 2019 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2019.01.035

items such as movies, music, books, news, and web pages. However, social networks generate a massive data including people connections, locations, interests…
With this rapid development, the study of social-based recommender systems takes a big interest in a lot of data science researches. The fact is, when we are confused by multiple choices, we may turn to the friends we trust the most for the best recommendations, since they are those who we can reach for immediate advice. Hence, in order to provide more accuracy and personalized recommendation results, relationship and trust information both should be incorporated.

Recommendation systems suggest items to the user by estimating the ratings that the user would give to that item (e.g. books, movies, vacations). The estimation of ratings can be performed by using heuristics and machine learning approaches. In the literature there are three base approaches to give recommendations, namely content based, collaborative filtering and hybrid. While content based filtering uses item similarity to give recommendations collaborative filtering uses user similarity. [1] affirms that users like to receive recommendations from people they know or they are similar and trust based recommendation approaches perform better than approaches that are only based on user similarity.

This paper is based on [2] that integrates social network graph and the user-item matrix to improve the prediction accuracy of the traditional recommender systems and we integrate an algorithm for best recommended trust path selection, to identify multiple recommended trust paths and to determine an aggregate path between users.
In the process of recommendation, friendships among users and the tags labeled by the users are used for recommendation. The user-item-tag can be considered as a two-dimensional matrix. Similar users are clustered to calculate the similarity between users and the correlation between a user and an item. The purpose in clustering is to identify the most suitable friends for realistic recommendation tasks. Based on the approach in [3], the above two detailed aspects of social network information are employed in designing social regularization terms.

In [2], the situation that different friends may have dissimilar or even opposite tastes was taken into consideration. Even if the friends of the same group focus on the same item, they may have different favourite degree. Then, we apply Dijkstra algorithm to identify the aggregate path. We have conducted experiments on real dataset to evaluate the performance of this proposition on the prediction accuracy. The experiments show significant improvement over the traditional recommender systems.

The remainder of the paper is organized as follows. Section 2 presents the overview of related work. Section 3 defines the problem and presents the details of the approach. Section 4 presents the experiments results. Finally, we draw the conclusion in Section 5.

2. Related work

Trust is one of the most important concept in social network. It's an important social concept which can rapidly affect users decisions [4]. People use this concept of trust to help decide the extent to which they interact with others [5]. Based on this, decision support systems, as a tool to support decision making process, also use trust information among users to more effectively help them make their decisions in social networks. In particular, most existing successful recommender systems consider trust relations and recommend items to a target user from her trusted users [6]. It has been shown that incorporating trust into recommender systems can improve the quality and coverage of recommendations [7–10]. As a result, trust inference has been the focus of the great deal of attentions in recent years.

In the studies of trust propagation, Golbeck [11] proposed TidalTrust for establishing the trust relation between a source user and the target one based on averaging trust values along the strongest trust paths among all the shortest ones. The author has investigated the performance of the proposed algorithm with respect to simulated and actual trust networks. The SUNNY algorithm [12] uses a probabilistic sampling technique to estimate the confidence and computes the trust based on only those information sources with the highest confidence estimates. Actually, SUNNY executes the trust inference procedure from TidalTrust on a more confident sub network. Work in [13] described a trust inference algorithm called MoleTrust which discovers all shortest paths from the source user to a given target user and aggregates all direct trust values by calculating the weighted average. To improve accuracy, MoleTrust ignores any trust information from users whose trustworthiness is less than 0.6.

Shakeri et al. [14] introduced an interval-based trust model to provide an integrated representation of trust and confidence. They also proposed two operators, named as trust interval Multiplication and summation, where the former propagates trust and confidence along all paths from a source user to a given target user and the latter aggregates two or more trust opinions. Work in [15] proposed a multi-dimensional evidence-based trust management system called MeTrust. The algorithm MeTrust uses -norm to evaluate the trust for each path and the weighted average to combine the trust among multiple paths, where the weight of each path is deduced from the uncertainty of trust along the path.

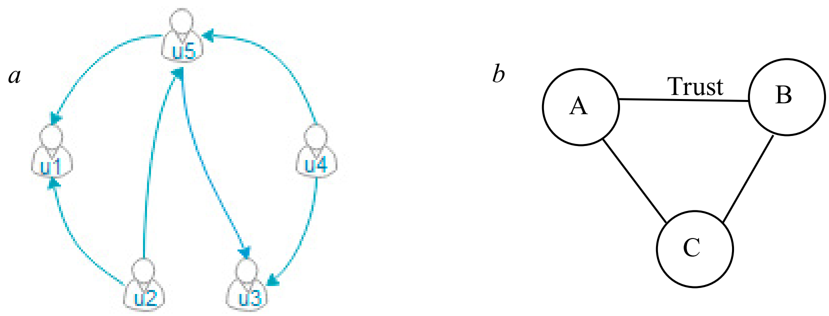
Kim and Song [16] investigated the impact of the length of available trust paths and different aggregation methods on the accuracy of trust propagation. Authors presented four strategies for predicting the value of trust based on reinforcement learning and evaluated the prediction accuracy of those strategies: weighted mean aggregation among the shortest paths, min–max aggregation among the shortest paths, weighted mean aggregation among all paths, and min–max aggregation among all paths. They observed that the best is the combination strategy min–weighted mean among all trust paths. Kim also presented an enriched trust propagation approach based on this strategy in [17]. In his work, by combining a homophily based trust network with an expertise-based trust network, he tackled the sparsity problem of the trust network.

Hang et al. [18] have proposed a trust path selection approach, where belief is considered as a most relevant trustworthy service. In [13], a mechanism based on indirect trust has been presented for removing the untrustworthy recommendations. However, the recommended trust value has not been considered. In the above mentioned approaches, although the trust value is taken into account, they are not applicable to determine trustworthy decision making in online social networks.

Recently, user generated data items in online social networks originate the new age of Big Data problems [19]. The huge volume of data cannot be processed or analyzed efficiently using statistical tools or traditional data analytic methods. Big Data creates many challenging research issues in the context of online social networking analysis [20]. In the new era of Big Data, it is challenging to identify the most relevant trust information in online social networks. In [21], a geometric differential learning model has been proposed to handle multimedia Big Data in online social networks for video recommendations. Therefore, we have proposed arecommended trust path selection approach which helps the participants to identify the untrustworthy recommendations and the recommended trust paths.

3. Social Recommendation Framework

In this section, we first use a synthetic example to illustrate some and abbreviations to social recommendation which are used throughout the paper. Then, we describe the model which integrates with social network information. The brief flow chart of the frame work is shown in Fig.2.In [3] and as mentioned before, the users are clustered to obtain suitable groups of friends, then we calculate the shortest path between users to have the most trusted users. Lastly, we interpret how to utilize regularization terms to model the framework.



c

	I_1	I_2	I_3	I_4	I_5	I_6	I_7
U_1	T_{11}		T_{13}				T_{17}
U_2		T_{22}			T_{25}	T_{26}	
U_3	T_{31}		T_{33}	T_{34}			T_{37}
U_4		T_{42}		T_{44}			
U_5	T_{51}		T_{53}		T_{55}	T_{56}	T_{57}

Fig. 1. (a) Social network; (b) Trust network; (c) The user-item matrix

As Fig 1.(a) shows the typical friends network graph. There are 5 users (nodes) with 6 relations (edges) between users. Each edge represents the connection between two users. The users often rate some items either on a 5-point integer scale (the bigger the better) or on tags to express the level of the favour of each item. The target is to predict the missing values or tags of the user-item matrix which is illustrated in Fig 1. (c). In Fig 1. (c) (3), T_{ij} denotes the tags (value) user *i* gave to item *j*.

For example, in Fig. 1(a), we can see that U_1 has a link with U_2 and U_5. U_2 has a link with U_1 and U_5. In Fig. 2(c), U_1 pays attention to items I_1, I_3 and I_7 and uses tags (T_{11}, T_{13} and T_{17}) to label them respectively. U_3 pays attention to item I_4 besides items I_1, I_3 and I_7. On the contrary, U_2 pays attention to items I_2, I_5 and I_6 which are quite different from the concerns of U_1.So; obviously, U_1 would not ask U_2 for help because of the different favors. Although U_4 has a link with U_1 and he/she also pays attention to item I_4, U_1 would turn to U_3 rather than U_4 for help to make the final decisions when consulting something about item I_4. It is obvious that U_1 and U_3 have the most similar favors.

The flowchart below, presents the framework architecture:

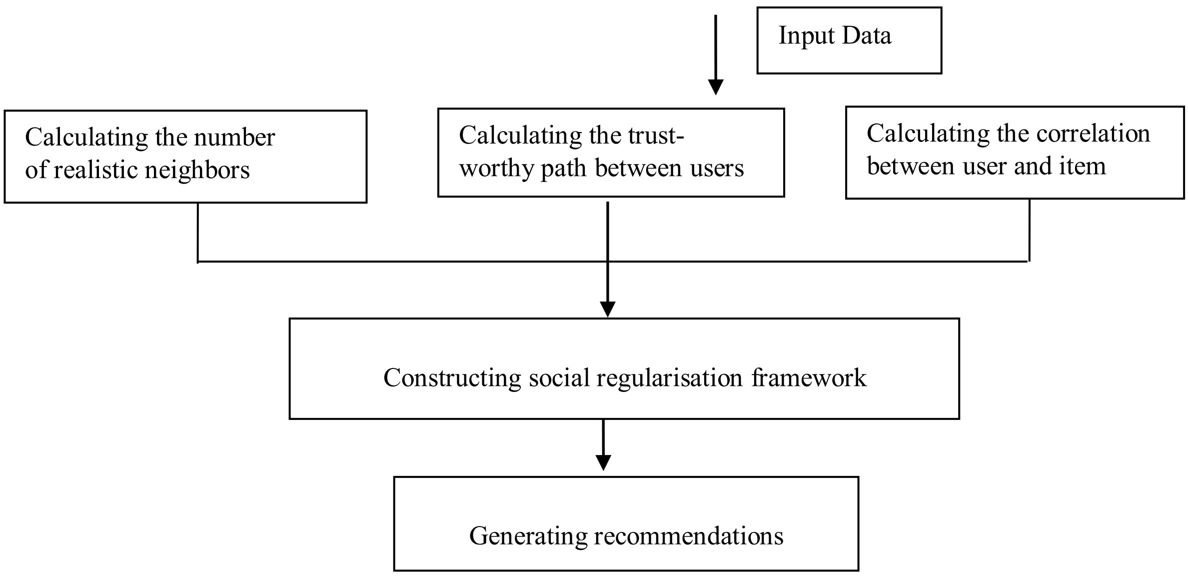


Fig. 2. The flowchart of the RSTN framework.

Table 1 The symbols

Symbol	Definition and Description
U	The user list
T	The user-item rating matrix
R*i*	The tag vector of user *i*
F(i)	The friend list of user *i*
I	The item list
S*i*	The vector of user *i*
R	The item-tag matrix

3.2 Proposed Model Description

The traditional recommender systems ignore the friendships among users. They just utilize the user-item matrix to generate recommendation. In fact, we often need to listen to the recommendations of friends intentionally or unintentionally according to the following rules: (1) The recommendations of the users who have the same or similar tastes or favors. (2) The recommendations of the experts in some field. Based on the above two considerations, the proposed matrix factorization framework with social regularization [3]. As we can see on the flowchart (Figure 2), in order to achieve better recommender results, the suitable group of friends are clustered and the shortest path is calculated and the correlations among users and items. Friendships and tags are combined as regularization terms to constrain the matrix factorization framework. So, we take the realistic situation that the friends with different favors recommend different results into consideration.

(a) User-item correlation

The traditional recommender systems often ignore the correlation between user and item. In [3], in order to calculate the correlation between user and item, the user and item are mapped to the tag space based on [4] and the similarity is calculated according to the formula as follows:

$$l(u,j) = vss\ (\boldsymbol{R}u\ ,\ Vj) \qquad (1)$$

where Ru denotes the tag vector of user u; and Vj denotes the vector of item j.
Similar to (1), the weights of tags are defined according to the following formula:

$$w_{jt} = \sum_u \frac{1}{|\boldsymbol{M}_{uj}|} \qquad if\ t \in \boldsymbol{M}_{uj}\ \ u \in \boldsymbol{U} \qquad (2)$$

where w_{jt} denotes the weight of tag t of item j, M_{uj} is the tags list and $|\ M_{uj}|$ is the number of tags which user u gave to item j. The metric here is also cosine similarity.

(b) Trust calculation formula

To compute the trust value between users, we follow the approach proposed by Lathia et al. [23] based on difference of a user's rating and its recommender's rating to their common item(s). Hence, as the distance between their rating values increases, trust decreases linearly. Assume we have two users Ua and Ub. Trust between them is formalized as follows [23]:

$$T(u_a, u_b) = 1 - \frac{\sum_{i=1}^{n}\left(r_{u_a,i_i} - r_{u_b,i_i}\right)}{r_{max} * n} \qquad (3)$$

Calculates the total differences between user's and its recommender's rating values over n historical ratings of u_a multiplied by the maximum value in each rating scale.

And For obtaining trust value between two indirect users, we adapt regular multiplication of trust value assigned to edges of their connecting path:

$$T(u_a, u_c) = T(u_a, u_b) * T(u_b, u_c) \tag{4}$$

This trust propagation formula facilitates u_a as a source to find its mutual trust values with all of its indirect neighbours.

Prediction of the Recommendations collected from direct or indirect neighbors are done by the weighted average of their rating based on their trust values calculated either through computation or propagation

$$p(a,i) = \frac{\sum_{b \in N(a,i)} (tr(r_{b,i}) * T(a,b))}{\sum_{b \in N(a,i)} (T(a,b))} \tag{5}$$

3.3 Social Regularization

A matrix factorization framework with social regularization was proposed in [4]. It firstly incorporates all the social connections of each user. However, it does not comply with the practical situation. Based on the intuition that we should identify the most suitable group of friends for different recommendation tasks, we incorporate the users' friendships into the matrix factorization framework. In this paper, we only consider the individual-based regularization approach. Based on the social recommendation model proposed in [4], the objective function is defined as follows.

$$min_{S,V}\ L(\boldsymbol{T}, \boldsymbol{S}, \boldsymbol{V}) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} I_{ij}\ \ (\boldsymbol{T}_{ij} - \boldsymbol{S}_i^T \boldsymbol{V}_j)^2 + \frac{\beta}{2} \sum_{i=1}^{m} \sum_{f \in F(i)} S(i,f) \left\| \boldsymbol{S}_i - \boldsymbol{S}_f \right\|_F^2 + \frac{\lambda_1}{2} ||\boldsymbol{S}||_F^2 + \frac{\lambda_2}{2} ||\boldsymbol{V}||_F^2 \tag{6}$$

Where $\mathbf{I}_{ij}$ is the indicator function that is equal to 1 if user i rated item j, ij T denotes the tags which user i gave to item j, $\mathbf{T}_{ij}$ denotes the item vector of user i, $\mathbf{V}_{ij}$ is the vector of item j,$\| \cdot \|^2_F$ represents the Frobenius norm, ,$\beta > 0$ and $F(i)$ is the friend list of user i. The last two regularization terms in formula (3) are used to avoid over-fitting. In the above objective function, a social regularization term is imposed:

$$\frac{\beta}{2} \sum_{i=1}^{m} \sum_{f \in F(i)} S(i,f)\ ||\boldsymbol{S}_i - S_f||_F^2 \tag{7}$$

Where $S(i, f)$ denotes the friendship between user i and f, the function allows the regularization term to treat users' friends differently. The more details can be found in [10]. In this paper, we integrate not only the friendships among users but also the correlation between the user and item into the model. We propose the following regularization terms to impose constraints between one user and their friends individually:

$$\frac{\alpha}{2} \sum_{i=1}^{m} \sum_{f \in F(i)} l(i,f)\ ||\boldsymbol{S}_i - S_f||_F^2 + \frac{\beta}{2} \sum_{i=1}^{m} \sum_{f \in F(i)} S(i,f)\ ||\boldsymbol{S}_i - S_f||_F^2 \tag{8}$$

Where $\alpha > 0, \beta > 0$, $l(j, f)$ denotes the correlation between item j and user f. The function can treat the friends differently based on the item. If user i gave tags to item j, and the correlation between item j and user f is high, such as $l(j, f) = 0.95$, which means user f makes great contributions to the tastes of user i. $S(i, f)$ denotes the friendship between user i and f. A small value of $S(i, f)$ or $l(j, f)$ represents the distance between feature vectors $\mathbf{S}_i$ and $\mathbf{S}_f$ should be larger. The objective function can be defined as follows:

$$min_{S,V}\ L(\boldsymbol{T},\boldsymbol{S},\boldsymbol{V}) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}\ (\boldsymbol{T}_{ij} - \boldsymbol{S}_i^T\boldsymbol{V}_j)^2 + \frac{\alpha}{2}\sum_{i=1}^{m}\sum_{f\in F(i)} l(i,f)\left|\left|\boldsymbol{S}_i - \boldsymbol{S}_f\right|\right|_F^2 + \frac{\beta}{2}\sum_{i=1}^{m}\sum_{f\in F(i)} S(i,f)\left|\left|\boldsymbol{S}_i - \boldsymbol{S}_f\right|\right|_F^2 + \frac{\lambda_1}{2}||\boldsymbol{S}||_F^2 + \frac{\lambda_2}{2}||\boldsymbol{V}||_F^2 \quad (9)$$

We can apply gradient descent algorithm to feature vector $\mathbf{S}_i$ and $\mathbf{V}_j$ to obtain a local minimum of the objective function.

4. Experimental Analysis

In this section, we conduct experiments on real dataset to validate the effectiveness of our approach. The proposed approach is implemented in JAVA. All the experiments are conducted on a windows machine with Intel processors (1.80GHz) and 4 GB memory.

4.1 Datasets

With the rapid development of Web 2.0 technology, a lot of data has been produced every day. People influence each other through the services of social networks. To carry out these experiments, Epinions, FilmTrust were used as sources of data to evaluate the approaches.

The first dataset used for the evaluation is Epinions. The Epinions website is a social platform where users can exchange their opinions on different types of articles. Users can vote for articles and are socially connected to each other through a trust graph. Each member of Epinions maintains a list of "trust" that presents a network of trusted relationships between users and "mistrust" list that presents a network of distrust relationships. This network is called "the Web of trust”.

Table 2 Epinions user-item matrix statistics.

Statistics	User	Item
Max.Num ratings	1,960	7,082
Mean.Num ratings	12.21	7.56

Table 3 Epinions dataset statistics.

Statistics	Trust per user	Trusted by user
Max.Num	1,763	2,443
Mean.Num	9.91	9.91

The second dataset is FileTrust data set was extracted from FilmTrust. This data set has two files, rating.txt and trust.txt. ratings.txt contains 35,497 data records, which has three attributes (userid, movieid and movieRating).

Table 4 FilmTrust dataset statistics.

	Users	Items	Ratings
Statistics	1,508	2,071	35,497

4.2. Metrics

In the experiment, we use the popular metrics, precision and recall, to measure the prediction quality of the proposed approach. The precision and recall are defined as follows:

$$Precision = \frac{|R(u) \cap T(u)|}{|T(u)|} \tag{10}$$

$$Recall = \frac{|R(u) \cap T(u)|}{|T(u)|} \tag{11}$$

Where *R(u)* denotes the tags that user u may label, and *T(u)* denotes the actual tags that user *u* labeled. The precision refers to the number of items which *u* labeled takes the proportion of the entire recommendation items. It reflects the possibility that *u* is interested in recommender item. The recall refers to the number of items which *u* labeled takes the proportion of all the items. We also calculate the Mean Absolute Error defined as follows:

$$MAE = \frac{1}{T} \sum_{i,j} |R_{ij} - \widetilde{\bar{R}_{ij}}| \tag{12}$$

Where *Rij* denotes the rating that the user has given to item *j*, $\widetilde{\bar{R}_{ij}}$ represents the notation that the user has given to item *j* as a value predicted by a method, and *T* denotes the number of ratings tested.

This measurement is the most popular error function. It evaluates the quality of the predictions provided by the recommendation system. We also calculate the Root Mean Squared Error, described by the equation (10):

$$RMSE = \sqrt{\frac{1}{T} \sum_{i,j} (R_{ij} - \widetilde{\bar{R}_{ij}})^2} \tag{13}$$

In the literature, the root of the mean squared error is widely used, instead of the MSE, to evaluate recommendation systems. It is used by the famous Netflix Prize contest to identify the best filtering algorithms.

4.3 Simulations and results analysis

The first experiment is conducted on the Epinions dataset, with the 2 approaches seen previously, the Recommender System based on Social Networks (RSboSN) approach and the RSboSN with the trust-worthy path selection (RSTP)

Table 5. MAE and RMSE for Epinions dataset.

Dataset	Approach	Metric	
		MAE	RMSE
Epinions	RSboSN	0,5432	0,6403
	RSTP	0,1080	0,2110

We note that the RSTP approach gives better performance with an absolute average error value equal to 0.1080, on the other hand comes the RSboSN approach with 0.5432.

The two metrics recall and accuracy have been defined before, the accuracy being equal to the number of relevant suggestions / number of suggestions, and the reminder as being the number of relevant suggestions offered to the user / number of relevant suggestions total.

To evaluate the accuracy and the performance of the two approaches, we calculate the precision and recall, results are presented below:

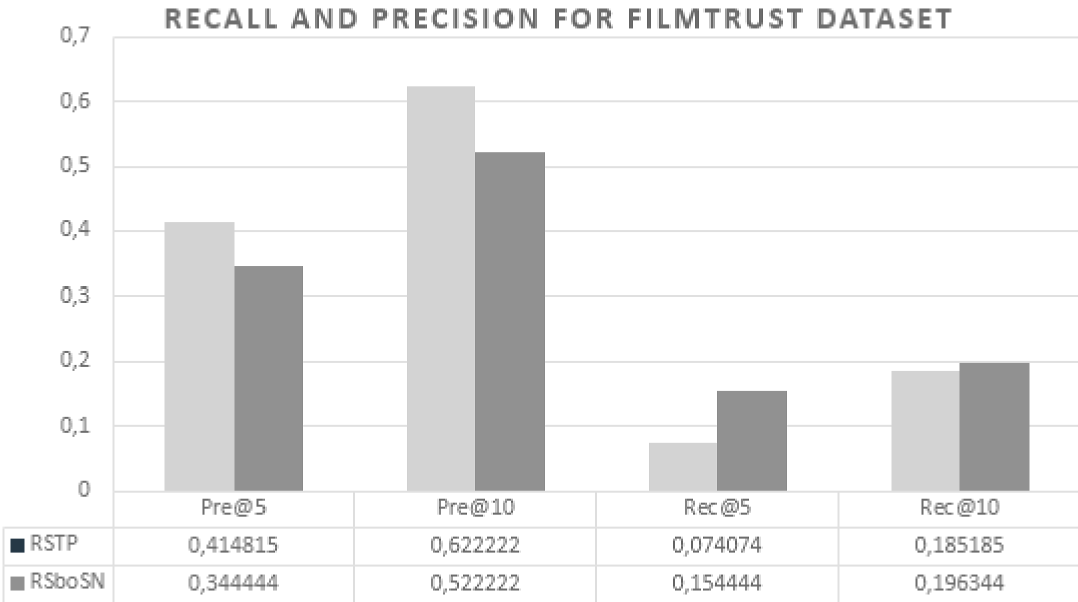


Fig. 3. Recall and precision for FilmTrust Dataset

The histogram above represents a comparison between the two approaches RSboSN and RSTP. *Pre@i* will be noted as the value of precision when the user recommends *i* items and *Rec@i* is the value of accuracy when the user recommends *i* items.

As shown, the RSTP approach offers more precision to the recommender system,for example when the users rec ommends 5 items the value of precision is about 0,41481 and for 10 items recommended, we achieve 0,62222 of pre cision. When it comes to performance the RSboSN is more performing.
The experiments shows that RSTP approach performs better than RSboSN regarding precision, but inversely regarding recall, particularly "at 5",we can say that RSTP approach offers more accuracy than RSboSN but when it c omes to performance the RSboSN win.

Table 6. Build time for Epinions dataset

Dataset	Approach	Build time (s)
	RSTP	408
Epinions	RSboSN	155

This table represents the time computing for the two approaches using Epinions dataset, the RSTP consume more time than the RSboSN which uses gradient descent for optimization.

5. Conclusion

In this paper, we presented a fusion of a social regularization approach that incorporates social network information to benefit recommender systems and trust information to identify an aggregate trust path in a social graph. Based on the intuition that a user's social network will affect this user's recommendation and the importance of interactions between users in social network, we presented a framework fusing a user-item rating matrix and the user's social network using matrix factorization with a selection of the recommended trust path to add more accuracy to the final recommendations. The experimental results show that this combination outperforms the traditional collaborative filtering algorithms and can avoid the cold start problem. Moreover, we used different metrics to evaluate the performance and accuracy of our recommender system.

References

[1]. J. Golbeck, "Trust and nuanced profile similarity in online social networks," TWEB, vol. 3, no. 4, 2009.

[2]. Sun, Z., Han, L., Huang, W., Wang, X., Zeng, X., Wang, M., Yan, H.,Recommender Systems Based On Social Networks, The Journal of Systems and Software (2014)

[3]. Ma H, Zhou D, and Liu C, et. al. Recommender systems with social regularization. In Proceedings of the 4th ACM International Conference on Web Search and Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011: 287-296.

[4]. U. Kuter, J. Golbeck, Using probabilistic confidence models for trust inference in Web-based social networks, ACM Trans. Internet Technol. 10 (2010)

[5]. E. Gray, J.-M. Seigneur, Y. Chen, C. Jensen, Trust propagation in small worlds, in: Int. Conf.Trust Manag., Springer, 2003 [6]]. J. Tang, X. Hu, H. Liu, Social recommendation: a review, Soc. Netw. Anal. Min. 3 (2013)

[7]. M. Ghavipour, M.R.M.R. Meybodi, An adaptive fuzzy recommender system based on learning automata, Electron. Commer. Res. Appl. 20 (2016)

[8]. M. Mao, J. Lu, G. Zhang, J. Zhang, Multirelational social recommendations via multigraph ranking, IEEE Trans. Cybern. (2017).

[9]. H. Wu, K. Yue, Y. Pei, B. Li, Y. Zhao, F. Dong, Collaborative topic regression with social trust ensemble for recommendation in social media systems, Knowledge-Based Syst. 97 (2016)

[10]. W.-P. Lee, C.-Y. Ma, Enhancing collaborative recommendation performance by combining user preference and trust-distrust propagation in social networks, Knowledge-Based Syst. 106 (2016)

[11]. J. Golbeck, J. Hendler, Filmtrust: Movie recommendations using trust in web-based social networks, in: Proc. IEEE Consum. Commun. Netw. Conf., Citeseer, 2006: pp. 282–286.

[12]. U. Kuter, J. Golbeck, Using probabilistic confidence models for trust inference in Web-based social networks, ACM Trans. Internet Technol. 10 (2010) 8. doi:10.1145/1754393.1754397.

[13]. P. Avesani, P. Massa, R. Tiella, A trust-enhanced recommender system application: Moleskiing,Sac. (2005) 1589–1593.

[14]. H. Shakeri, A.G. Bafghi, A Confidence-Aware Interval-based Trust Model, ISC Int. J. Inf.Secur. 4 (2013).

[15]. G. Wang, J. Wu, Multi-dimensional evidence-based trust management with multi-trusted paths,Futur. Gener. Comput. Syst. 27 (2011) 529–538. doi:10.1016/j.future.2010.04.015.

[16]. Y.A. Kim, H.S. Song, Strategies for predicting local trust based on trust propagation in socialnetworks, Knowledge-Based Syst. 24 (2011) 1360–1371. doi:10.1016/j.knosys.2011.06.009.

[17]. Y.A. Kim, An enhanced trust propagation approach with expertise and homophily-based trustnetworks, Knowledge-Based Syst. 82 (2015) 20–28. doi:10.1016/j.knosys.2015.02.023.

[18]. H. Beigy, M.R. Meybodi, Utilizing distributed learning automata to solve stochastic shortest path problems, Int. J. Uncertainty, Fuzziness Knowledge-Based Syst. 14 (2006)

[19]. W. Jiang, J. Wu, F. Li, G. Wang, H. Zheng, Trust Evaluation in Online Social Networks Using Generalized Network Flow, IEEE Trans. Comput. 65

[20]. Tan W , Blake MB , Saleh I , Dustdar S . Social-network-sourced big data analytics. IEEE Internet Comput 2013

[21]. Xu K , Zou K , Huang Y , Yu X , Zhang X . Mining community and inferring friendship in mobile social networks. Neurocomputing 2016

[22]. Hang C-W , Wang Y , Singh MP . Operators for propagating trust and their evaluation in social networks. In: Proceedings of the 8th international conference on autonomous agents and multiagent systems-volume 2. International Foundation for Autonomous Agents and Multiagent Systems; 2009.

[23]. N. Lathia, S. Hailes, and L. Capra, "Trust-based collaborative filtering," in IFIPTM 2008: Joint iTrust and PST Conferences on Privacy, Trust management and Security, Department of Computer Science, University College London, London, UK, 2008, p. 14.