USER RECOMMENDATION WITH TENSOR FACTORIZATION IN SOCIAL NETWORKS

Zhenlei Yan, Jie Zhou

Department of Automation
Tsinghua University, Beijing, China

ABSTRACT

The rapid growth of population in social networks has posed a challenge to existing systems for recommending to a user new friends having similar interests. In this paper, we address this user recommendation problem in social networks by proposing a novel framework which utilizes users' tagging information with tensor factorization. This work brings two major contributions: (1) A tensor model is proposed to capture the potential association among user, user's interests and friends in social tagging systems; (2) A novel approach is proposed to recommend new friends based on this model. The experiments on a real-world dataset crawled from Last.fm show that the proposed method outperforms other state-of-the-art approaches.

***Index Terms*—** user recommendation, tensor factorization, social networks, tagging systems

1. INTRODUCTION

With the explosive growth of users and digital multimedia resources (music, photos or videos) on the web, lots of social networks, such as Last.fm[1] and Flickr[2] have adopted social tagging systems to organize the massive data. Social tagging system allows users to annotate resources on the web with their favorite words termed Tag. Tags not only furnish the meta information of multimedia data which is hard extracted directly, but also indicate users' interests [1]. On the other hand, users want to find people with similar interests, such as $friend$ in Last.fm or $contact$ in Flickr. However, the results of existing friends recommender systems are usually not satisfied [2]. It's not easy for a user to find new appropriate friends, especially in the rapidly growing population of social networks. Solving this problem brings two significant benefits. Firstly, it helps users to discover new interesting multimedia resources. Secondly, this recommendation service encourages interaction between users with similar interests and improves users' satisfaction which means more advertising revenue for web sites.

In this paper, we propose a novel framework with Tensor Factorization to perform the task of user recommendation. The proposed framework consists of three stages: (a) Constructing User-Interest-Friend model with tensor factorization; (b) Learning the best model parameters; (c) Ranking new friends for users. We summarize the contributions in the following: (1) We propose a new model with tensor factorization to capture the potential association among user, user's interests and friends; (2) Based on this model, we propose a novel approach to recommend new friends with similar interests for users.

The rest of this paper is organized as follows. In Section 2, we give an overview of previous related work. In Section 3, we describe the problem formally. We present the proposed framework for user recommendation In Section 4. In Section 5, we discuss the evaluation of our method compared with other state-of-the-art approaches. Finally, we draw conclusions in Section 6.

2. RELATED WORK

By now, several approaches have been proposed to solve the problem of user recommendation in social tagging systems [3]. Plenty of existing recommender systems are based on Collaborative Filtering (CF) [4, 5], which has been widely employed, such as Amazon[3] and MovieLens[4]. Besides, Google Follower Finder (GF) adopts a method based on social graph [6]. This method only utilizes the link information of social graph and predicts new friends based on common friends of users. Recently, Zhou [2] proposed a two-stage framework (UR) in social tagging systems. This approach calculates the modularity of tags to represent users' interests, and recommends users based on the KL-divergence between their interests. Differently with previous methods, the proposed method considers both link structure and users' tagging content.

This work was supported by the Natural Science Foundation of China under Grant 60721003, 60875017, 61020106004 and 61021063. This work was supported by the Science & Technology Support Program of China under Grant 2009BAH40B03.

[1] http://www.lastfm.com
[2] http://www.flickr.com
[3] http://www.amazon.com
[4] http://www.movielens.org

3. PROBLEM STATEMENT

Typically, a social tagging system consists of entities (i.e. users, tags and resources) and relations between entities (e.g. friendship between users). We define the set of all users $U = \{u_i\}_{i=1}^{I}$, the set of all tags $T = \{t_j\}_{j=1}^{J}$ and the set of resources $R = \{r_k\}_{k=1}^{K}$. User-tag relation $(u,t) \in O \subseteq U \times T$ means that user u has annotated resources with tag t. The set of all tags which have been used by u is denoted as $T(u)$. User-friend $(u, u_i) \in P \subseteq U \times U$ means that u_i is one of friends of u. The set of all friends of u is denoted as $F(u)$.

Given a user u, the user recommender system is set to predict a personalized ranking list of Top-N users whom u wants to make friends with. This means that given a predictor $\hat{Y}$, we should predict a score $\hat{y}_{u,u_i}$ for each candidate friend u_i. For avoiding ambiguous notations , we use f to represent user's friends in stead of u_i. Consequently, the Top-N highest scoring users for u can be calculated by:

$$Top(u,N) = arg \max_{f \in U/\{u\}}^{N} \hat{y}_{u,f} \tag{1}$$

Where the superscript N denotes the number of users to be recommended.

4. THE PROPOSED METHOD FOR USER RECOMMENDATION

4.1. User-Interest-Friend Model with Tensor Factorization

Previous research efforts have shown that social tags could be used to indicate users' interests on the web [1]. Thus we propose the following assumption.

Assumption 1. *User's tags indicate user's interests.*

Under this assumption, we can treat $T(u)$ as the set of interests of u, and (u,t) means that u is interested in t (e.g. $rock$, pop and etc.). McPherson has proposed that users are more likely to make friends with others with similar interests [7]. This results in the following assumption.

Assumption 2. *User makes friends with others based on the similar interests.*

Combining Assumption 1 and 2, we can construct a set Ω of 3-order tensors to model the association among user, user's interests and friends as described in Proposition 1.

Proposition 1. $\forall f \in F(u), \forall t \in T(u) \cap T(f) \Rightarrow (u,t,f) \in \Omega$

A 3-order tensor (u,t,f) means that u makes friends with f possibly based on interest t. Given a predictor $\hat{Z}$ on Ω, $\hat{z}_{u,t,f_1} > \hat{z}_{u,t,f_2}$ means that u is more likely to make friends with f_1 than f_2 based on interest t. Then computing $\hat{y}_{u,f}$ can be decomposed to subtasks of computing $\hat{z}_{u,t,f}, \forall t \in T(u)$ as below.

$$\hat{y}_{u,f} = \sum_{t \in T(u)} \phi(\hat{z}_{u,t,f})$$

Where $\phi(\cdot) : \hat{Z} \rightarrow \hat{Y}$. Nextly, we estimate $\hat{Z}$ by factorizing tensors to three low-rank feature matrices which represents users, interests and friends respectively, and a core tensor. The predictor $\hat{Z}$ is made by multiplying the three feature matrices to the core tensor:

$$\hat{Z} = \hat{C} \times_u \hat{U} \times_t \hat{T} \times_f \hat{F}$$

Where the core tensor $\hat{C}$ and the feature matrices $\hat{U}$, $\hat{T}$ and $\hat{F}$ are the model parameters to be learned. The parameters are denoted as $\hat{\theta} = (\hat{C}, \hat{U}, \hat{T}, \hat{F})$ and have the following sizes: $\hat{C} \in \Re^{k_U \times k_T \times k_F}$, $\hat{U} \in \Re^{|U| \times k_U}$, $\hat{T} \in \Re^{|T| \times k_T}$, $\hat{F} \in \Re^{|U| \times k_F}$. k_U, k_T and k_F are the dimensions of the corresponding low-rank features. Then, given $\hat{\theta}$, $\hat{z}_{u,t,f}$ can be calculated as follows:

$$\hat{z}_{u,t,f} = \sum_{\tilde{u}=1}^{k_U} \sum_{\tilde{t}=1}^{k_T} \sum_{\tilde{f}=1}^{k_F} \hat{c}_{\tilde{u},\tilde{t},\tilde{f}} \cdot \hat{u}_{u,\tilde{u}} \cdot \hat{t}_{t,\tilde{t}} \cdot \hat{f}_{f,\tilde{f}} \tag{2}$$

4.2. Learning Model Parameters

In this paper, we employ an optimization criterion to find the best model parameters $\hat{\theta}$ by maximizing the ranking statistics AUC (area under the ROC-curve) as described in Equation 3 and 4.

$$arg \max_{\hat{\theta}} \sum_{(u,t) \in O} AUC(\hat{\theta}, u, t) \tag{3}$$

Where

$$AUC(\hat{\theta}, u, t) = \frac{1}{|F_{u,t}^{+}||F_{u,t}^{-}|} \sum_{f^+ \in F_{u,t}^{+}} \sum_{f^- \in F_{u,t}^{-}} H_{0.5}(\hat{z}_{u,t,f^+} - \hat{z}_{u,t,f^-}) \tag{4}$$

Where

$$F_{u,t}^{+} = \{f \mid (u,t) \in O \wedge (u,t,f) \in \Omega\}$$
$$F_{u,t}^{-} = \{f \mid (u,t) \in O \wedge (u,t,f) \notin \Omega\}$$

and $H_{0.5}$ is the Heaviside function. Then we use the gradient descent algorithm to learn the best model parameters. The learning process is similar with [8].

4.3. Friends Ranking

Once $\hat{\theta}$ is learned, we calculate $\hat{z}_{u,t,f}$ as Equation 2. Then for each (u,t), we get a ranking list $r_{u,t} = \{f_1, \ldots, f_{N_u} \mid \hat{z}_{u,t,f_1} > \ldots > \hat{z}_{u,t,f_{N_u}}\}$. N_u is the number of friends recommended for u. Unfortunately, $\{\hat{z}_{u,t,f}\}_{t \in T(u)}$ can't be directly sorted to generate $Top(u,N)$. To address this problem,

Table 1. Characteristic of the experimental dataset

Dataset	Users	Tags	Friends	User-Friend
Last.fm	988	16,895	3,802	22,051

we apply the Reciprocal Rank Fusion method [9] as function $\phi(\cdot)$:

$$\phi(\hat{z}_{u,t,f}) = \frac{1}{\Delta + r_{u,t}(f)} \tag{5}$$

Where Δ is a fixed parameter and $r_{u,t}(f)$ returns the ranking of f in $r_{u,t}$. Finally, we get $Top(u, N) = \{f_1, \ldots, f_N \mid \hat{y}_{u,f_1} > \ldots > \hat{y}_{u,f_N})\}$.

5. EXPERIMENTAL EVALUATION

In this section, we investigate the performance of the proposed method comparing with other state-of-the-art approaches, i.e. UR [2], CF [4] and GF [6].

5.1. Dataset

Since there are few available public datasets suitable for this scenario, we collected a real-world dataset by crawling Last.fm via its official API[5] from April 2011 to June 2011. To achieve the final experimental dataset, we sampled 1,000 users at random, then collected the friends of users and tags of all users and friends. After pruning invalid records, the characteristics of experimental dataset are shown in Table 1.

5.2. Evaluation Methodology

We use four well-known metrics to evaluate the performance of methods, namely Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), Precision, and Recall, which are widely used in recommendation problems [10]. For each user in the dataset, we select one of user-friend relations at random to compose the test set S_{test}, while the remaining relations and tags of users are the training set S_{train}. Then we learn models on S_{train} and predict Top-N friends for each user in S_{test}.

5.3. Results and Discussion

5.3.1. Experimental Parameters

For the proposed method, we set k_u = k_t = k_f = k_{dim} conveniently and run method with k_{dim} = 32. The corresponding method is termed as UTF-32. The other hyperparameters in our approach are: learning rate of gradient descent algorithm $\alpha = 0.1$; iteration number is 100 and $\Delta = 100$. The feature matrices $\hat{\theta}$ are initialized with the random values drawn from normal distribution $N(0, 0.01)$. For CF, we adopt the memory-based approach and use cosine distance to measure users' similarity. The number of neighbors is set to 100.

[5]http://www.last.fm/api

Table 2. Performance comparison of various methods

Methods	MRR	MAP	Precision	Recall
UTF-32	**0.0574**	**0.0088**	**0.0062**	**0.6238**
CF	0.0446	0.0070	0.0046	0.4593
UR	0.0305	0.0030	0.0033	0.3257
GF	0.0089	0.0016	0.0013	0.1303

5.3.2. Comparison of Methods

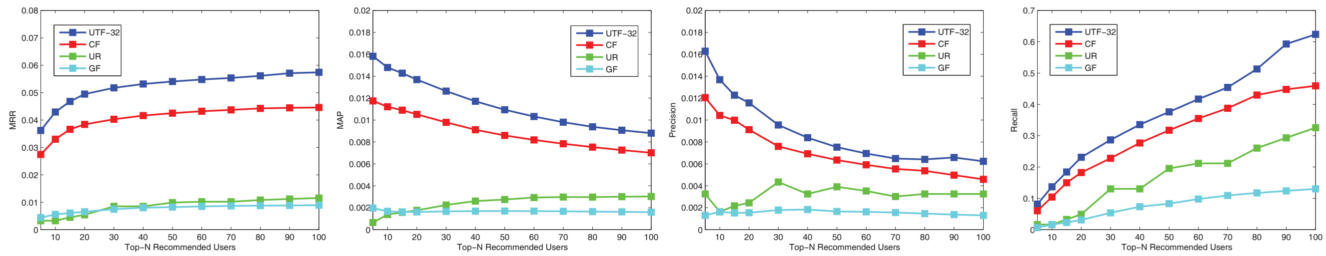
Table 2 demonstrates the results of evaluation in all metrics of our method and other approaches. The number of recommended friends N is set to 100. From Table 2, we can see that UTF-32 outperforms other approaches in all metrics. In addition, the results of interest-based approaches UR and CF outperform the graph-based method GF. In order to further compare the performance of the proposed method with other approaches extensively. We show results of our method and results of CF, UR and GF with different N. Figure 1(a)-(d) show the results of performance comparison of various methods in all metrics, i.e. MRR, MAP, Precision and Recall respectively. From Figure 1, we can see that the proposed method UTF-32 consistently outperforms other approaches in all metrics. The interest-based methods CF, UR still outperform GF and the performance of CF is better than UR.

5.3.3. Impact of Parameters k_u, k_t, k_f

Further more, we investigate the impact of parameters k_U, k_T and k_A. We run UTF models with $k_{dim} = \{16, 32, 64\}$ respectively. The corresponding methods are termed as UTF-16, UTF-32 and UTF-64 respectively. Figure 2(a)-(d) show the results of performance comparison of UTF methods in all metrics, i.e. MRR, MAP, Precision and Recall respectively. From Figure 2, we can see that as k_{dim} increasing from 16 to 64, the performance of corresponding method is also improved. UTF-32 outperforms UTF-16 in MAP, Precision and Recall (except MRR), while UTF-64 outperforms the other methods in all metrics. On the other hand, we should spend more time to train the corresponding model as k_{dim} increasing. In practice, we should strike a balance between the performance and complexity of the method.

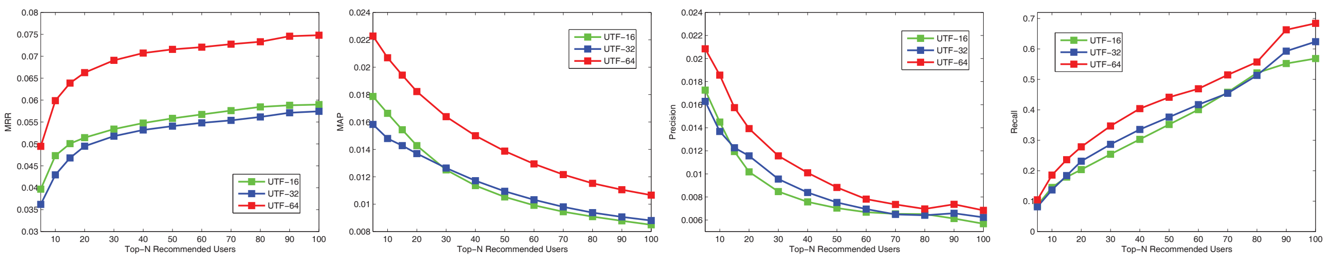
6. CONCLUSIONS

In this paper, we present a novel framework which utilizes users' tagging information with tensor factorization to address the user recommendation problem in social networks. Firstly, we propose a tensor model to capture the potential association among user, user's interests and friends. Next, we learn the model and calculate a personalized ranking list



(a) MRR (b) MAP (c) Precision (d) Recall

Fig. 1. Performance comparison of various methods



(a) MRR (b) MAP (c) Precision (d) Recall

Fig. 2. Impact of Parameters k_u,k_t,k_f

of new friends for each user. The results of experiments on a real-world dataset crawled from Last.fm show that the proposed method UTF outperforms other state-of-the-art approaches, i.e. UR, CF and GF.

7. REFERENCES

[1] F.M. Suchanek, M. Vojnovic, and D. Gunawardena, "Social tags: meaning and suggestions," in *Proceeding of the 17th ACM conference on Information and knowledge management*. ACM, 2008, pp. 223–232.

[2] T.C. Zhou, H. Ma, M.R. Lyu, and I. King, "Userrec: A user recommendation framework in social tagging systems," in *AAAI*, 2010.

[3] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, "A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 2, pp. 179–192, 2010.

[4] J.S. Breese, D. Heckerman, C. Kadie, et al., "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the 14th conference on Uncertainty in Artificial Intelligence*. Madison: Morgan Kaufmann, 1998, pp. 43–52.

[5] Quanquan Gu, Jie Zhou, and Chris H. Q. Ding, "Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs," in *SDM*, 2010, pp. 199–210.

[6] M. Pennacchiotti and S. Gurumurthy, "Investigating topic models for social media user recommendation," in *WWW*. ACM, 2011, pp. 101–102.

[7] M. McPherson, L. Smith-Lovin, and J.M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, pp. 415–444, 2001.

[8] S. Rendle, L. Balby Marinho, A. Nanopoulos, and L. Schmidt-Thieme, "Learning optimal ranking with tensor factorization for tag recommendation," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 727–736.

[9] G.V. Cormack, C.L.A. Clarke, and S. Buttcher, "Reciprocal rank fusion outperforms condorcet and individual rank learning methods," in *SIGIR*. ACM, 2009, pp. 758–759.

[10] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 5–53, 2004.