HeterRank: Addressing Information Heterogeneity for Personalized Recommendation in Social Tagging Systems

Wei Feng
Tsinghua University
Beijing, China
feng-w10@mails.tsinghua.edu.cn

Jianyong Wang
Tsinghua University
Beijing, China
jianyong@tsinghua.edu.cn

ABSTRACT

A social tagging system provides users an effective way to collaboratively annotate and organize items with their own tags. A social tagging system contains heterogenous information like users' tagging behaviors, social networks, tag semantics and item profiles. All the heterogenous information helps alleviate the cold start problem due to data sparsity. In this paper, we model a social tagging system as a multi-type graph and propose a graph-based ranking algorithm called HeterRank for tag recommendation. Experimental results on three publicly available datasets, i.e., CiteULike, Last.fm and Delicious prove the effectiveness of HeterRank for tag recommendation with heterogenous information.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering, Retrieval Models, Selection Process*

General Terms

Algorithms

Keywords

Social Tagging System, Recommender System, Information Heterogeneity

1. INTRODUCTION

In social tagging systems, users can annotate and organize items with their own tags for future search and sharing. Many social tagging systems have achieved great success, such as Delicious[1]. Personalized tag recommendation is the key part of a social tagging system. When a user wants to annotate an item, the user may have her/his own vocabulary to organize items. Personalized tag recommendation tries to find the tags that can both meet the user's annotation habits and precisely describe the item. A social tagging system, as shown in Figure 1, contains heterogeneous information and can be modeled as a graph:

- Users(U), tags(T) and item(I) co-exist in the graph.
- Inter-relationships. Edges between users, tags and items can be derived from annotation behaviors <user, tag, item>. Suppose we have $u \in U$ and $t \in T$, the weight of $<u, t>$ is the times of tag t being used by user u. The same rule applies to $<u, i>$ and $<i, t>$ $(i \in I)$.
- Intra-relationships. (1) Social network among users.(2) Tag semantic network based on semantic relatedness. (3) Item network based on content similarities.

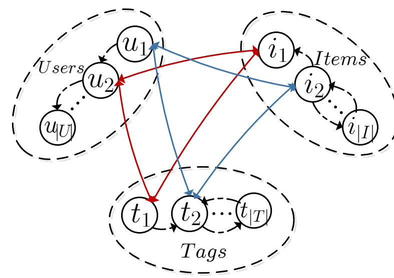

Figure 1: Social Tagging System

Our goal is to model all sources of information to address the cold start problem due to data sparsity. However, few work has been done in this field. To address information heterogeneity, we propose a graph based ranking method called HeterRank. Given a user u and an item i for tag recommendation, HeterRank performs a random walk with restart at user u and item i to assign each tag a visiting probability. Only tags that are both relevant to u and i can get a high visiting probability.

2. GRAPH BASED RECOMMENDATION

HeterRank extends the random walk with restart to the heterogenous graph. With frequently restart at u and i, all the tags are ranked according to their visiting probabilities from u and i. Formally, HeterRank is performed according to the following equation:

$$\begin{pmatrix} \mathbf{p}_U \\ \mathbf{p}_T \\ \mathbf{p}_I \end{pmatrix}^{(t+1)} = (1-\alpha)\mathbf{S} \begin{pmatrix} \mathbf{p}_U \\ \mathbf{p}_T \\ \mathbf{p}_I \end{pmatrix}^{(t)} + \alpha \begin{pmatrix} \mathbf{q}_U \\ \mathbf{q}_T \\ \mathbf{q}_I \end{pmatrix} \quad (1)$$

where α is the restart probability and t represents the the number of iterations. Vectors $\mathbf{p}_U$, $\mathbf{p}_T$ and $\mathbf{p}_I$ is the visiting probabilities of users, tags and items, respectively. $\mathbf{S}$ is the transition matrix based on the graph structure. Vectors $\mathbf{q}_U$, $\mathbf{q}_T$ and $\mathbf{q}_I$ represent the preferences of users, tags and items for restart. Now we introduce the the transition matrix $\mathbf{S}$ and the preference vector $\mathbf{q}^T = (\mathbf{q}_U^T, \mathbf{q}_T^T, \mathbf{q}_I^T)$ in detail.

Transition Matrix Let G denote the whole graph shown in Figure 1 and let G_{MN} $(M, N \in \{U, T, I\})$ denote the sub-graph made up by relation $<m, n>$ $(m \in M, n \in N)$. Let $\mathbf{A}_{MN}$ denote the adjacent matrix of the sub-graph G_{MN}.

[1] http://delicious.com

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

$\mathbf{A}_{MN}(i,j)$ represents the weight of the edge <j, i>[2]. The transition matrix $\mathbf{S}$ is computed by two steps: (1)Since different $\mathbf{A}_{MN}$ are measured in different metrics, each column of $\mathbf{A}_{MN}$ is normalized to have sum 1. (2) Since the importance of $\mathbf{A}_{MN}$ differs, we re-scale each $\mathbf{A}_{MN}$ by multiplying it with a factor t_{MN}. For example, assuming the social network is less important than the tagging history of users, we can set t_{UU} to be smaller than t_{TU} and t_{IU}. Formally, the transition matrix $\mathbf{S}$ is defined as follows:

$$\mathbf{S} = \begin{bmatrix} t_{UU}\mathbf{A}_{UU}\mathbf{D}_{UU}^{-1} & t_{UT}\mathbf{A}_{UT}\mathbf{D}_{UT}^{-1} & t_{UI}\mathbf{A}_{UI}\mathbf{D}_{UI}^{-1} \\ t_{TU}\mathbf{A}_{TU}\mathbf{D}_{TU}^{-1} & t_{TT}\mathbf{A}_{TT}\mathbf{D}_{TT}^{-1} & t_{TI}\mathbf{A}_{TI}\mathbf{D}_{TI}^{-1} \\ t_{IU}\mathbf{A}_{IU}\mathbf{D}_{IU}^{-1} & t_{IT}\mathbf{A}_{IT}\mathbf{D}_{IT}^{-1} & t_{II}\mathbf{A}_{II}\mathbf{D}_{II}^{-1} \end{bmatrix} \quad (2)$$

where $\mathbf{D}_{MN}$ $(M,N \in \{U,T,I\})$ is a diagonal matrix and the i-th entry is the sum of the i-th column of $\mathbf{A}_{MN}$. For each $N \in \{U,T,I\}$, we have $t_{UN} + t_{TN} + t_{IN} = 1$.

Preference Vector Initially, all entries in the preference vector $\mathbf{q}^T = (\mathbf{q}_U^T, \mathbf{q}_T^T, \mathbf{q}_I^T)$ are set to 1, which means all nodes have a small probability for restart. Given a user u and an item i for personalized tag recommendation, the corresponding entries $\mathbf{q}_U(u)$ and $\mathbf{q}_I(i)$ are respectively set to |U| and |I|. This is the same with FolkRank [1]. In other words, u and i have a much higher probability for restart. Then each $\mathbf{q}_M$ $(M \in \{U,T,I\})$ is normalized to have sum 1. Finally, considering the importance of each type of nodes differs, we re-scale each $\mathbf{q}_M$ $(M \in \{U,T,I\})$ by multiplying it with a factor r_M. The preference vector $\mathbf{q}$ is defined as follows

$$\mathbf{q}^T = (r_U\mathbf{q}_U^T/D_U, r_T\mathbf{q}_T^T/D_T, r_I\mathbf{q}_I^T/D_I) \quad (3)$$

where D_M $(M \in \{U,T,I\})$ is the sum of $\mathbf{q}_U$. We add a constraint r_U+r_T+r_I=1 to make $\mathbf{q}$ sum to 1.

With the the transition matrix $\mathbf{S}$ and the preference vector $\mathbf{q}$ defined, we can take a closer look at the intuition of how $\mathbf{p}_T$ is computed according to Equation 1:

$$\mathbf{p}_T^{(t+1)} = (1-\alpha)(\overline{\mathbf{A}}_{TU}\mathbf{p}_U^{(t)} + \overline{\mathbf{A}}_{TT}\mathbf{p}_T^{(t)} + \overline{\mathbf{A}}_{TI}\mathbf{p}_I^{(t)}) + \alpha\overline{\mathbf{q}}_T \quad (4)$$

where $\overline{\mathbf{A}}_{MN}$=$t_{MN}\mathbf{A}_{MN}\mathbf{D}_{MN}^{-1}$ and $\overline{\mathbf{q}}_M$=$r_M\mathbf{q}_M/D_M$ $(M,N \in \{U,T,I\})$. When α=0, $\mathbf{p}_T^{(t+1)}$ receives scores spread from $\mathbf{p}_U^{(t)}$, $\mathbf{p}_T^{(t)}$ and $\mathbf{p}_I^{(t)}$. The same rule applies to $\mathbf{p}_U^{(t+1)}$ and $\mathbf{p}_I^{(t+1)}$. In other words, users, tags and items reinforce each other through different types of relations until a stable state is reached. For $t \in T$, t will get a high ranking only when t has highly ranked neighbors of users, tags and items. When α is greater than 0, the personalized information is considered by frequently restart at the target user and item.

3. EXPERIMENTAL STUDY

To prove the effectiveness of HeterRank, we conducted extensive experiments on three publicly available datasets: CiteULike[3] with tag relations, Last.fm with user relations, and Delicious with user relations and item relations. Last.fm and Delicious are online available[4]. CiteULike has 3152 users, 9561 tags, 54816 items, 49006 tag relations and 483790 posts. Tag relatedness is computed by WikipediaMiner[5]. Last.fm has 1892 users, 9749 tags, 12523 items, 25434 user relations, 24164 posts. User relations are all mutual friends. Delicious has 1867 users, 69223 tags and 40678 items, 15328 user relations and 151971 item relations and 437593 posts. User relations are mutual and binary weighted. Item relations are computed using vector model with TF-IDF weighting for each word in the webpage. For each user, one of his post is held out to construct the test data. We use precision to measure the performance.

[2]Different from the convention, $\mathbf{A}_{MN}$ is column indexed.
[3]http://www.citeulike.org/faq/data.adp
[4]http://www.grouplens.org/node/462
[5]http://wikipedia-miner.cms.waikato.ac.nz

Table 1: CiteULike (Tag Relations)

Algorithm	P@1	P@2	P@3	P@4	P@5
FR	0.164	0.143	0.125	0.112	0.102
HR_∅	0.159	0.145	0.129	0.116	0.106
HR_T	**0.180**	**0.159**	**0.137**	**0.125**	**0.114**

Table 2: Last.fm (User Relations)

Algorithm	P@1	P@2	P@3	P@4	P@5
FR	0.305	0.262	0.228	0.202	0.182
HR_∅	0.341	0.293	0.256	0.226	0.206
HR_U	**0.349**	**0.299**	**0.263**	**0.233**	**0.212**

Table 3: Delicious (User and Item Relations)

Algorithm	P@1	P@2	P@3	P@4	P@5
FR	0.257	0.214	0.186	0.163	0.148
HR_∅	0.246	0.214	0.188	0.171	0.155
HR_I	0.272	0.231	0.203	**0.182**	**0.166**
HR_U	0.277	0.231	0.201	0.178	0.162
HR_UI	**0.281**	**0.235**	**0.204**	0.181	0.165

3.1 Parameter Estimation

We draw a small sample from the data to find the best $\{t_{MN}|\ M,N \in \{U,T,I\}\}$. Suppose M is U, once t_{UU} is set to a fixed value a_{uu}, we have $t_{UT} + t_{UI} = 1 - a_{uu}$. Then we only need to decide how (1 - a_{uu}) is split by t_{UI} and t_{UU}. We set the step size σ=±0.1, ±0.15, ±0.2 so that

$$t_{UI} = \frac{1-a_{uu}}{2} + \sigma \qquad t_{UT} = \frac{1-a_{uu}}{2} - \sigma$$

a_{uu} cannot be too large (greater than 0.5) because social network can be viewed as the background information and is a weak feature. The step size σ cannot be too small. A step size of 0.01 can hardly influence the ranking. This search strategy also applies to r_M $(M \in \{U,T,I\})$.

3.2 Experimental Results

We choose FolkRank[1] as our baseline, which is the state-of-the-art graph-based method. The results are shown in Tables 1, 2 and 3. FolkRank and HeterRank with only <user, tag, item> relations available is denoted by FR and HR_∅, respectively. HR_U, HR_T and HR_I denote HeterRank with user relations, tag relations and item relations, respectively. HR_UI combines user relations and item relations together. When performed only on <user, tag, item>, FolkRank and HeterRank are comparable. When intra relations are introduced, HeterRank successfully combined the newly introduced relations and outperforms the baseline.

3.3 Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grant No. 60833003, National Basic Research Program of China (973 Program) under Grant No. 2011CB302206, and an HP Labs Innovation Research Program award.

4. REFERENCES

[1] R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in folksonomies. In *PKDD*, 2007.