Weng, Li-Tung and Xu, Yue and Li, Yuefeng and Nayak, Richi (2007) Improving Recommendation Novelty Based on Topic Taxonomy. In *Proceedings 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology Workshops*, pages pp. 115-118, Silicon Valley.

2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops

Improving Recommendation Novelty Based on Topic Taxonomy

Li-Tung Weng, Yue Xu, Yuefeng Li, Richi Nayak
Queensland University of Technology
soloman1124@hotmail.com, yue.xu@qut.edu.au, y2.li@qut.edu.au, r.nayak@qut.edu.au

Abstract

Clustering has been a widely applied approach to improve the computation efficiency of collaborative filtering based recommendation systems. Many techniques have been suggested to discover the item-to-item, user-to-user, and item-to-user associations within user clusters. However, there are few systems utilize the cluster based topic-to-topic associations to make recommendations. This paper suggests a taxonomy-based recommender system that utilizes cluster based topic-to-topic associations to improve its recommendation quality and novelty.

1. Introduction

Hybrid recommenders combine the features of collaborative filtering and content based recommenders are very popular nowadays[1-3]. There are two major benefits about hybrid recommenders. Firstly, their recommendations are usually lesser content-centric than standard content based recommenders. Secondly, they are lesser prone to the data-sparsity problem than collaborative filtering based recommenders. Because the recommendations generated by hybrid recommenders (or collaborative-filtering based recommenders) are not based on the item content similarity, they are generally considered more novel than recommendations generated by standard content based recommenders. However, this additional aid to the novelty aspect of the recommendations is not considered very effective [2, 3].

In this paper, we proposed a taxonomy-based recommender to improve the novelty and quality of recommendations. Instead of the simple content similarity measurement used by most hybrid recommenders, our system uses association rule mining techniques to mine the taxonomy interest associations between different users, and the recommendations with high novelty and quality are generated based on the discovered user taxonomy interests.

2. Definitions

We envision a world with a finite set of users $U = \{u_1, u_2, \ldots, u_n\}$ and a finite set of items $T = \{t_1, t_2, \ldots, t_m\}$. For each user $u \in U$, he or she is associated with a set of items he or she has rated, we denote these items $RT_{ALL}(u) \subseteq T$. These rated items can be further divided into two categories according to the rating method: implicitly rated items $RT_{impl}(u) \subseteq RT_{ALL}(u)$ and explicitly rated items $RT_{\exp l}(u) \subseteq RT_{ALL}(u)$ where $RT_{\exp l}(u) \cap RT_{impl}(u) = \phi$.

In explicit ratings, users express their preferences to items in numeric form. We use $rating(u,t)$ to denote user u 's rating to item $t \in RT_{\exp l}(u)$, where $0 \le rating(u,t) \le 1.0$.

Our system uses taxonomy based descriptors to describe items. Specifically, for every $t_i \in T$, $D(t_i) = \{d_1, d_2 \ldots, d_n\}$ denotes a set of descriptors characterizing t_i's taxonomy. Importantly, an item may possess more than one taxonomy descriptor to cover multiple taxonomic aspects of the item.

A taxonomy descriptor is a sequence of ordered topics, denoted as $d = (p_0, p_1, \ldots, p_q)$ where $d \in D(t_i)$ and $t_i \in T$. The topics within a descriptor are sequenced so that the former topics are super topics of the latter topics, specifically, p_j is the direct super topic for p_{j+1} where $0 \le j < q$.

Let C be the set of all taxonomy topics such that $C = \{p \mid p \in d, d \in D(t_i), t_i \in T\}$. In order to differentiate between super topics and sub-topic as well as impose the tree structure from the topic set, we define mapping $E : C \to 2^C$ that retrieves all direct sub-topics $E(p_a) \subset C$ for $p_a \in C$ and $E(p_a) \cap E(p_b) = \varnothing$ for any $p_a, p_b \in C$, $a \ne b$. With the mapping E we can recursively extract the taxonomy tree structure from the set C. Moreover, as in standard tree structures, the taxonomy tree has exactly one top-most element with zero indegree covering the most general topic, it is denoted by Γ in this paper. By contrast, for these bottom-most elements with zero outdegree, they are denoted by set $\perp$ and cover the most specific topic concepts. In our

0-7695-3028-1/07 $25.00 © 2007 IEEE
DOI 10.1109/WI-IATW.2007.59

system, for any item descriptor $d = (p_0, p_1, \ldots, p_q)$, it is required $p_0 = \Gamma$ and $p_q \in \perp$.

We define two functions to calculate the topic popularity. The function $user\_topic\_count(p,u)$ returns the frequency of a given topic p appearing in an user u 's past ratings. For example, assume u has rated only one book and this book contains two descriptors. If the given topic p appears in both descriptors, then $user\_topic\_count(p,u)$ returns 2. Another function $topic\_count(p,U')$ returns the total frequency of the topic p within a given user set $U' \subseteq U$. This function is simply the sum of topic frequencies (i.e. $user\_topic\_count(p,u)$) of users in U'.

3. User cluster taxonomy profile

In order to improve the efficiency of the system, we cluster the user set U into $UC = \{uc_1, uc_2, \ldots, uc_k\}$, a set of clusters based on users' explicit ratings, where $\bigcup_{uc \in UC} uc = U$ and $\bigcap_{uc \in UC} uc = \varnothing$. Moreover, for convenience, a user u 's cluster is denoted as $cluster(u)$.

3.1. Hot Topics

For each cluster $uc \in UC$, we build a local cluster based taxonomy tree similar to the global taxonomy tree defined in previous section. Let

$$C_{uc} = \{p \mid p \in d, d \in D(t), t \in RT_{\exp l}(u), u \in uc\}$$

be the topic set for cluster uc and $E_{uc}(p) \subset C_{uc}$ be the sub topic set of $p \in C_{uc}$. We use the following equation to measure the distinctness of a topic in a local cluster with the global user set:

$$topic\_score(p, C_{uc}) = \frac{topic\_count(p, uc)}{topic\_count(p, U)} \quad (1)$$

The higher the topic score, the higher the possibility that the topic is interested by the users in this cluster. We call the topics which have high scores hot topics as defined below:

$$hot\_topics(uc, \varsigma) = \\ \{p \mid p \in C_{uc}, topic\_score(p, C_{uc}) > \varsigma\} \quad (2)$$

where ς is a user defined threshold.

Items related to hot topics should be preferably recommended to the users in this cluster.

3.2. Topic Associations

For the topic taxonomy dataset, each transaction is a set of topics related to the items rated by a user. By applying data mining techniques to the topic taxonomy dataset, we can derive association rules among topics which describe how topics associate with each other within a cluster. By applying the association rules, we could be able to recommend items which are not rated by previous users based on user ratings but might be interested by the target user based on the topic taxonomy associations. For each user cluster uc, $rules(uc)$ is the set of association rules derived from the topic transactions related to the users in uc. $rules(uc)$ is defined below:

$$rules(uc) = \\ \{(\rho_a \to \rho_b) \mid \rho_a \subseteq C_{uc}, \rho_a \subseteq C_{uc}, \\ \rho_a \cap \rho_b = \varnothing, P(\rho_b \mid \rho_a) > \xi\}$$

where $P(\rho_b \mid \rho_a)$ is the confidence of the rule, ξ is the confidence threshold. The meaning of the rule $(\rho_a \to \rho_b)$ can be interpreted as "the set of topics ρ_b might be interested by a user $u \in uc$ if the user u is interested in the set of topics ρ_a ".

Based on the association rules $rules(uc)$ discovered from uc, we can then obtain a set of novel topics for a given user u . We firstly obtain the topic set from the user's past ratings

$$user\_topics(u) = \\ \{p \mid p \in d, d \in D(t), t \in RT_{ALL}(u), p \in \perp\} \quad (3)$$

Next, we observe how topics are associated with each other within u 's belonging cluster uc by using $rules(uc)$. We loop through all topic association rules returned via $rules(uc)$, and check if there are any rule patterns matches u 's topic interests (i.e. $user\_topics(u)$). Specifically, let $(\rho_a \to \rho_b)$ be a rule, then this rule matches u 's topic interests, if $\rho_a \subseteq user\_topics(u)$. Based on the definition of a rule $(\rho_a \to \rho_b)$ described previously, it can be easily observed ρ_b can be a potential set of topics that might be also interested by u .

Specifically, we can compute the weight of a potentially associated topic (i.e. $p \in \rho_b$, $(\rho_a \to \rho_b)$ matches $user\_topics(u)$) by:

$$score(p,u) = \frac{conf((\rho_a \rightarrow \rho_b)) \times user\_topic\_count(p,u)}{\sum_{q \in user\_topics(u)} user\_topic\_count(q,u)}$$

The higher the computed weight indicates the higher the possibility that the user might be interested in the topic. Finally, we collect all these potential topics from the matched rules, and compute their weights.

Formula $novel\_topics(u)$ is used to denote the algorithm described above, where u is a given target user. The output of $novel\_topics(u)$ is a set of pairs $(p, score)$, where p is a proposed novel topic and $score$ is the topic score computed by $score(p,u)$.

4. Taxonomy-based recommender

.

4.1. Item-based Collaborative Filtering (CF)

The basic idea of item-based CF is to predict item t to user u based on the item similarity between t and the items that have been rated by u [9]. The similarity between two items is computed based on user explicit ratings as defined below:

$$item\_sim(t_i,t_j) = \frac{\sum_{u \in U_{ij}} (r_u^i - \overline{r_u})(r_u^j - \overline{r_u})}{\sqrt{\sum_{u \in U_{ij}} (r_u^i - \overline{r_u})^2} \sqrt{\sum_{u \in U} (r_u^j - \overline{r_u})^2}} \quad (4)$$

where r_u^i represents user u 's rating to item t_i, $\overline{r_u}$ is user u 's average explicit ratings: $\overline{r_u} = \frac{\sum_{t \in RT_{\exp l}(u)} r_u^t}{|RT_{\exp l}(u)|}$, U_{ij} is the set of users who have rated both t_i and t_j. U_{ij} is defined as below:

$$U_{ij} = \{u \in U \mid \{t_i, t_j\} \subset RT_{\exp l}(u)\}$$

Note, it is possible that two items are never rated by more than one user, i.e., $U_{ij} = \varnothing$. In such case, $item\_sim(t_i,t_j)$ returns a special value NC which is a label indicating "Not Computable".

As mentioned above, the prediction of t to user u is based on the similarities between t and the items $x \in RT_{\exp l}(u)$ rated by the user u, where $t \neq x$. In order to achieve it, we need to find the target user's rated items which are computable with the target item t. That is,

$$cItems(u,t) = \{x \in RT_{\exp l}(u) \mid item\_sim(t,x) != NC\}.$$

The prediction of t to u is computed as follows:

$$\eta_{u,t} = \frac{\sum_{y \in cItems(u,i)} (item\_sim(y,t) \bullet r_u^y)}{\sum_{y \in cItems(u,i)} |item\_sim(y,t)|} \quad (5)$$

where $0 \leq \eta_{u,t} \leq 1$.

4.2. Topic Preference

As discussed in Section 3, users in the same cluster have similar topic interests and most likely prefer items that relate to the hot topics of this cluster. Under this assumption, in this paper we propose to take topic preferences into consideration in predicting items to a given user. The prediction is computed not only based on item similarities (i.e. equation (5)) but also based on the given user's topic preference which will be discussed below.

Let $\psi_{u,t}$ denote the degree of user u 's preference towards to item t 's topics. The value of $\psi_{u,t}$ is computed differently according to three different conditions. Firstly, if the topic of t is considered very popular in u 's cluster, that is

$$\delta(t) \cap hot\_topics(cluster(u), \varsigma) \neq \varnothing$$

then $\psi_{u,t}$'s value indicates the degree of popularity of t 's topics in $cluster(u)$. Specifically,

$$\psi_{u,t} = \max_{p \in (\delta(t) \cap hot\_topics(cluster(u), \varsigma))} topic\_score(p, C_{uc})$$

In the case that t 's topics are not popular in u 's cluster, we then check if the topics are considered novel in the cluster, that is

$$\{p \in \delta(t) \mid (q,s) \in novel\_topics(u), p = q\} \neq \varnothing$$

. If the predicates returns true, we then assign $\psi_{u,t}$ the degree of novelty for t 's topics in $cluster(u)$. Specifically,

$$\psi_{u,t} = \max_{\substack{(p,s) \in novel\_topics(u), \\ \{p\} \cap \delta(t) \neq \varnothing}} s \, .$$

Finally, if t 's topics are neither popular nor novel in u 's cluster, we will simply not recommend this item, so that $\psi_{u,t} = 0$.

4.3. Incorporation of Topic Preference with Item-based CF

In order to recommend a set of k items to a target user $u \in U$, we firstly form a candidate item list containing all items rated by u 's neighbors but not yet rated by u . Next, for each item t in the candidate list, we compute the item preference score (i.e. $\eta_{u,t}$) and topic preference score (i.e. $\psi_{u,t}$) for them. The proposed preference ranking for each candidate item can then be computed by combining the item preference score and topic preference score together. Finally, k candidate items with highest preference rankings are recommended to the user u , and these recommended items are sorted by the ranking values in descending order. The completed algorithm is listed below:

```
Algorithm $taxonomy\_recommender(u,k)$
  where $u \in U$ is a given target user
        $k$ is the number of items to be recommended

  1) SET $\gamma_u = [\bigcup_{w \in cluster(u)} RT_{ALL}(w)] \setminus RT_{ALL}(u)$ , the
     candidate item list
  2) FOR EACH $t \in \gamma_u$
  3)     SET $rank_{u,t} = \alpha\eta_{u,t} + (1-\alpha)\psi_{u,t}$
  4) END FOR
  5) Return the top $k$ items with highest $rank_{u,t}$
     score to $u$ .
```

It can be seen in line (3) of the algorithm, the predicted ranking for an item is computed based on the linear combination of item preference score $\eta_{u,t}$ and topic preference score $\psi_{u,t}$. The coefficient α used in the formula is used to adjust the weights of $\eta_{u,t}$ and $\psi_{u,t}$ in the final ranking score, and it can is computed by:

$$\alpha = \frac{\varpi\vartheta}{\varpi\vartheta + (1-\varpi)(1-\vartheta)} \qquad (6)$$

where, $\varpi = \frac{|cItem(u,t)|}{|RT_{\exp l}(u)|}$ and

$0 \leq \vartheta \leq 1$, is a user controlled variable.

In the computation of α , it can be seen ϖ reflects the quality confidence of $\eta_{u,t}$, because the more the target user's past rated items related to the target item, the higher the accuracy of the item preference prediction (i.e. $\eta_{u,t}$) will be. Thus, when ϖ increases α will increase too, and therefore $\eta_{u,t}$ will receive higher weight in the final score (i.e. $rank_{u,t}$). Variable ϑ , on another hand, is used to adjust the weights of ϖ in α , thus, if ϑ is large (e.g. 0.9) $\eta_{u,t}$ will still receive high weight even ϖ is small.

The proposed algorithm can also be used to solve the cold start and data sparsity problems [4, 5]. For datasets with very few explicit ratings, the proposed algorithm can still use the item taxonomy information and the users' implicit ratings to make quality recommendations.

5. Conclusion

A taxonomy-based recommender is proposed in this paper aim for solving the recommendation novelty problem. The proposed recommender utilizes techniques from association rule mining to find how different topics are associated with each other in a given user cluster. Based on the discovered topic associations, the recommender suggests items with topics that are strongly linked to the taxonomy profile of the target user. Besides considering only the topic similarities as suggested by many other systems, the proposed system improves recommendation novelties by recommending items with novel topics strongly associated to the target user profiles.

References

[1] R. Burke, "Hybrid Recommender Systems: Survey and Experiments," *User Modeling and User-Adapted Interaction*, vol. 12, pp. 331-370, 2002.

[2] P. Melville, R. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," presented at Proceedings of 18th National Conference on Artificial Intelligence, 2002.

[3] O. v. Meteren and M. v. Someren, "Using Content-Based Filtering for Recommendation," presented at ECML2000 Workshop, 2000.

[4] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and Metrics for Cold-start Recommendations," presented at Proceedings of 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002.

[5] C.-N. Ziegler, G. Lausen, and L. Schmidt-Thieme, "Taxonomy-driven Computation of Product Recommendations " presented at International Conference on Information and Knowledge Management Washington D.C., USA 2004.