Association for Information Systems

AIS Electronic Library (AISeL)

ICEB 2010 Proceedings

International Conference on Electronic Business (ICEB)

Winter 12-1-2010

Solving Cold Start Problem in Collaborative Filtring Method of Recommender Systems

Zainab Khanzadeh

Mehrgan Mahdavi

SOLVING COLD START PROBLEM IN COLLABORATIVE FILTERING METHOD OF RECOMMENDER SYSTEMS

Zainab Khanzadeh, Department of Computer Research Studies, Islamic Azad University, Arak, Iran.
E-mail: e.z.khanzadeh@gmail.com

Mehrgan Mahdavi, Department of Computer Engineering, University of Guilan, Rasht, Iran.
E-mail: mehrgan.mahdavi@gmail.com

Abstract

Recommender systems try to recommend articles of potential interest to a user with respect to the user's individual preferences. Such recommender systems are the focus of current interest in part because of their importance for e-business. Collaborative Filtering is the most promising technique in recommender systems. It provides personalized recommendations according to user preferences. But one of the problems of Collaborative Filtering is cold-start. Here, we provide a novel approach for solving this problem through the attributes of items in order to recommend articles to more people for improving e-business.

Keywords- Cold-start; Collaborative Filtering; Recommender Systems

Introduction

Recommender Systems

Recommender Systems were introduced as a computer-based intelligent technique to alleviate the challenge of information and product overload [1]. They are designed to understand users' interests, learn from them based on past buying behavior and recommend items (whether they be products, books, movies etc) that will be interest of them [4]. They can be utilized to efficiently provide personalized services in most e-business domains, benefiting both the customer and the merchant. Recommender Systems will benefit the customer by making to him find items they wish to purchase. At the same time, the business will be benefited by the increase the amount of sales which will normally occur when the customer is presented with more items he would likely find appealing [1].

Type of Recommender Systems

There are four major types of recommender systems:
1) Collaborative Filtering
2) Content-based
3) Knowledge-based
4) Hybrid systems

Collaborative Filtering

Collaborative recommendation (CF) makes recommendations to a user by matching him/her with other users of the system and finding similarities and dissimilarities between this user and other users. Generally, it tries to find a group of people whose opinions are similar with the given or active user and recommends items they like to that user. The similarities or dissimilarities between users are calculated from analysis of their past buying behavior [2]. For example, if Sue and Jerry have liked many of the same movies, and Sue liked *Titanic,* which Jerry hasn't seen yet, then the system may recommend *Titanic* to Jerry [5]. Gustos, GroupLens, FireFly and Jester are all examples of collaborative recommender systems [2]. Figure1 show the work of recommender system based on collaborative filtering.

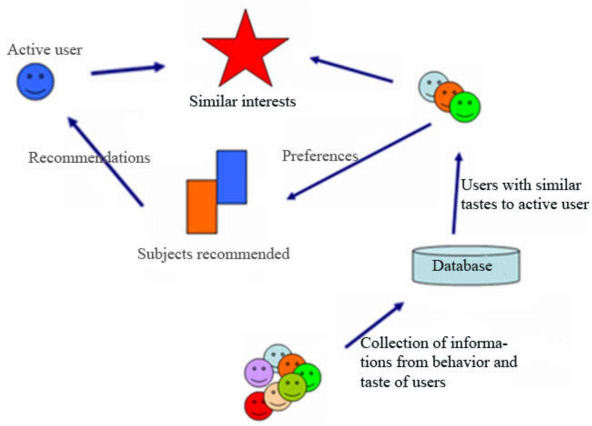

Figure 1. Simple description work of collaborative filtering

Some disadvantages of this method are as follow:

- Cold-start: This problem has two aspects. First, when a new item enters to the system, that item has not been rated by users. Therefore collaborative filtering method cannot provide recommendations for items with no or very few ratings data. Secondly, when a new user enters to the system and has been rated no items or items that no one rated them, collaborative filtering method cannot understand what the user interested in.

Therefore cannot recommend item to that user [4].

- Sparsity: Considering that, there is no possible that all products by all users are rated, so that the matrix to show preference for goods customers will be formed for many existing products have been empty, so probability of finding congenial customers is low. This problem occurs in store with too article. Also, this problem occurs too when system is in early and learning stages [3].

- Popularity Bias: Collaborative filtering method cannot recommend items to someone with unique tastes. It tends to recommend popular items [3].

- Scability: When number of users is hundreds or thousands person, algorithms of collaborative filtering work well. But nowadays, e-business is growing rapidly and number of user is grown more than millions persons. Millions of user data that is called neighbor, for providing recommendation should be calculate in real time that makes response time is very long. Searching among millions neighbor is a time consuming process and thence, these systems cannot work well [3].

Content-based

Content-based (CB) recommendation tries to recommend items based on information on the content of items the user has liked. Generally, a user's profile is constructed by analyzing and extracting useful features from the content of the items that the user has bought. Then recommendations are made based on this user's profiles. Content Advisor and LIBRA is a content based web page recommender, which stores the features of items in databases and responds to users' searches for the content they desire [2].

LIBRA (Learning Intelligent Book Recommending Agent) is a book recommender system [6]. It uses a database of book information extracted from web pages at Amazon.com [8]. LIBRA, originally built by Mooney as a content based recommender system, has a database of books categorized by six genres: literature, mystery, romance, science fiction, science and technology, and children's books. Each genre has approximately 8000 books. The books are stored in a semistructured representation of ISBN, Author, Title, Description, Subject, Related Authors, and Related Titles [7]. Figure 2 show work of LIBRA system in Amazon site for recommending book.

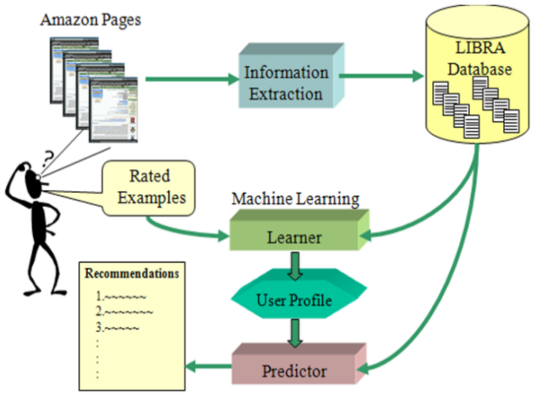

Figure 2. LIBRA System

Sample Extracted Information from Amazon's pages:
Title: <The Age of Spiritual Machines: When Computers Exceed Human Intelligence>
Author: <Ray Kurzweil>
Price: <11.96>
Publication Date: <January 2000>
ISBN: <0140282025>
Related Titles: <Title: <Robot: Mere Machine or Transcendent Mind>
Author: <Hans Moravec> >
…
Reviews: <Author: <Amazon.com Reviews> Text: <How much do we humans…> >
…
Comments: <Stars: <4> Author: <Stephen A. Haines> Text:<Kurzweil has …> >
…
Related Authors: <Hans P. Moravec> <K. Eric Drexler>…
Subjects: <Science/Mathematics> <Computers> <Artificial Intelligence> …

Advantages of this method are as follow [8]:

- No need for data on other users. So it has not cold-start or sparsity problems.
- Able to recommend to users with unique tastes.
- Able to recommend new and unpopular items.

Disadvantages of this method are as follow:

- Requires content that can be encoded as meaningful features [3].
- Unable to exploit quality judgments of other

users. Unless these are somehow included in the content features [3].

- User can receive only recommendations based on his/her previous purchases that their content saves in his/her profile. So, if the user has tastes that are not represented in his/her profile, items talking to the unrepresented taste will not be recommended [7].

- Content-based system can offer the user reliable recommendation when user has rated enough items and system can identify user interests. Therefore, for a new user with low or fewer ratings, system is unable to provide the right recommendation [3].

Knowledge-based

Knowledge-based (KB) systems make use of knowledge about users and products to generate recommendations. They use a reasoning process to determine what products meet a user's requirements [9]. The Personal Logic recommender system offers a dialog that effectively walks the user down a discrimination tree of product features. Others have adapted quantitative decision support tools for this task [10].

.

Hybrid systems

Often, designers of these systems combine two or three types of triple types that already said, for two purposes:

- Increasing efficiency of system
- Decreasing the effect of weak points, when each systems to employ alone

Our approach

As we have mentioned above, one of the problem of collaborative filtering method is cold-start. This problem has two aspects. One aspect occurs when a new item enters into the system. So the item probably has not been rated by too many users. Therefore collaborative filtering method cannot provide recommendations for items with no or very few ratings data. The second aspect occurs when a new user enters into the system. he/she has rated no items or has rated items that no one has rated them. Here, collaborative filtering method cannot understand what the user interested in. Therefore cannot recommend items to that user [4].

Our work focuses on the first aspect of cold-start. When a user (can be a new user) has very few ratings in his/her profile and the items has rated are part of cold-start items (no one has rated them), recommender systems may fail to provide recommendations that user interests [4], because collaborative systems failing to find similar users.

The proposed algorithm

In our algorithm, we try through attributes of items find similar users with cold-start user. Then, items that they are interested, are recommended to cold-start user. Of course, users who percentage of their beloved items -which their attributes are similar to attributes of cold-start user beloved items - is more, they are in priority. In the other word, at first their beloved items are recommended to cold-start user and then beloved items of the others.

Input: The amount minattr (percent of similarity one item with another item, based on their attributes), the amount minsim (percent of similarity one user with another user), the value of n (number of recommendations).

Output: n recommendations for each cold-start user

1. First, we should find cold-start users. If user has rated items that they have not been rated by no one else, he/she will be spotted as a cold-start user.
2. **For** each cold-start user u_i **do**
3. Put in A set all the items have selected by cold-start user u_i.
4. $A = \{a_1, a_2, \dots\}$.
5. **For** each user u_j except of cold-start user u_i **do**
6. Put in B set all the items that have been selected by user u_j.
7. $B = \{b_1, b_2, \dots\}$.
8. Find all possible association rules from A and B itemsets on the form of $B \rightarrow A$. (Of course, here instead of B and A, we should put elements of these itemsets.)
9. Now, we should specify which of the rules obtained is correct. For this, we should calculate the value of *crule* for each rule. So, we should note to the features of the items A and B and see what percentage of features of these two items are equivalent together. As follow:
10. $crule = \frac{number\ of\ the\ same\ attributes\ of\ two\ items}{number\ of\ total\ attributes\ of\ two\ items} \times 100$
11. **If** $crule \geq minattr$ **then**
12. This rule can be considered as a correct rule. So, we conclude that two items are similar.
13. **End if**
14. Now, we should understand that user u_j is similar to cold-start user u_i. For this, we should calculate the value of *sim* for user u_j. As follow:

15. $sim_{u_j} = \frac{\text{number of correct rules obtained for user } u_j}{\text{number of total rules obtained for user } u_j} \times 100$
16. **If** $sim_{u_j} \geq minsim$ **then**
17. We can conclude that the user u_j is similar to cold-start user u_i.
18. **End if**
19. **End for**
20. Now, we want to provide n recommendations for cold-start user u_i. From all users similar to cold-start user u_i, respectively, we choose users with a higher value of *sim* and then recommend n items selected by them but not selected by cold-start user u_i to cold-start user u_i. (Because we assume that, users similar to cold-start user u_i are in a group. So, we act like as collaborative filtering method of recommender systems.)
21. **End for**

The point that should be consider is that at above algorithm whatever the values of *minattr* and *minsim* be more, the accuracy of recommendations to be more. Of course must be careful that if the value of these numbers is too high, possibility it cannot offer any recommendation.

An example of work of algorithm

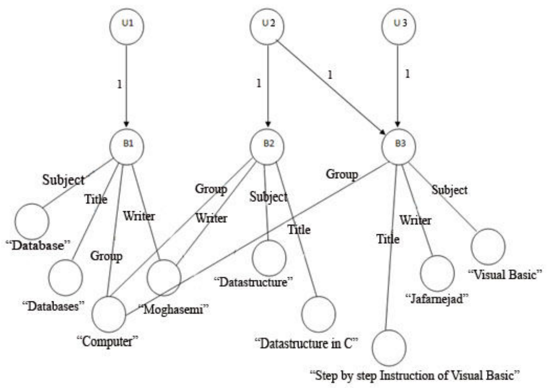

Figure 3. The proposed preference model in book recommender system

Figure 3 shows a simple example of the proposed preference model in a book recommender system.

It consists of three layers, namely the User layer, the Item layer, and the Item Attribute layer. An edge connecting a User node and an Item node means that the user has rated the item, and is labeled with a fuzzified rating in the range [0, 1] to reflect the degree to which the user had liked the item. But this model only captures membership degrees obtained by user ratings with respect to the Like fuzzy set. This is because the goal of CF is to recommend to users items they may like. Besides, precedent work on ARM (Association Rule Mining) -based CF already proved that negative preferences are not useful for generating recommendations [11].

An edge connecting an Item node and an Item Attribute node represents a particular attribute of the item. The label of such an edge describes the name of the attribute while the value of an Item Attribute node describes the value of the attribute [11].

Input: Minattr=50, minsim=50, n=2

1. Here U1 is a cold-start user. Because he has selected B1 that has not been selected by no one else.
2. **For** cold-start user U1
3. $A = \{B_1\}$
4. **For** user U2
5. $B = \{B_2, B_3\}$
6. Mining association rules from A and B itemsets, on the form of $B \rightarrow A$.
7. $B_2 \rightarrow B_1$
8. $B_3 \rightarrow B_1$
9. For $B_2 \rightarrow B_1$
10. $crule = \frac{2}{4} \times 100 = 50 \geq minattr \Longrightarrow$ $Correct\ rule$
11. For $B_3 \rightarrow B_1$
12. $crule = \frac{1}{4} \times 100 = 25 < minattr \Rightarrow$ $is\ not\ correct$
13. $sim_{U2} = \frac{1}{2} \times 100 = 50 \geq Minsim \Longrightarrow$ $U2$ is similar to U1
14. **For** user U3
15. $B = \{B_3\}$
16. Mining association rules from A and B itemsets, on the form of $B \rightarrow A$.
17. $B_3 \rightarrow B_1$
18. $crule = \frac{1}{4} \times 100 = 25 < minattr \Longrightarrow$ is not correct
19. $sim_{U3} = \frac{0}{1} \times 100 = 0 < Minsim \Longrightarrow$ U3 is not similar to U1
20. Now, we want to provide two recommendations for U1. Here, just U2 is similar to U1. So, we recommend two items that has been selected by U2 but has not been selected by cold-start user U1. Thus, B2 and B3 are recommended to U1.

Conclusion

The rapid development of Internet technologies in recent decades has imposed a heavy information burden on users. This has led to the popularity of recommender systems, which provide advice to users about items they may like to examine. Collaborative Filtering is the most promising technique in recommender systems, providing personalized recommendations to users based on their previously expressed preferences and those of other similar users, but it cannot recommend items to someone with unique tastes. This problem is often referred to as the cold-start problem. This paper describes how we should solve this problem through the attributes of items.

References

[1] Emmanouil Vozalis, Konstantinos G. Margaritis, “Analysis of recommender systems’ algorithms”, Department of Applied Informatics, University of Macedonia.

[2] Weiyang Lin, " Association rule mining for collaborative recommender systems ", Submitted to the Faculty of the worcester polytechnic institute, In partial fulfillment of the requirements for the Degree of Master of Science in Computer Science, May 2000.

[3] Farshad Akbari, “Designing an improved hybrid recommender system for predicting customer buying behavior in mcommerce applications”, Thesis Submitted for M.Sc. Degree to University Postgraduate Education Office, University of Shiraz, Spring 2008.

[4] Gavin Shaw, Yue Xu, Shlomo Geva,” Investigating the use of association rules in improving recommender systems”, Proceedings of the 14th Australasian Document Computing Symposium, Sydney, Australia, 4 December 2009.

[5] Nathaniel Good, J. Ben Schafer, Joseph A. Konstan, Al Borchers, Badrul Sarwar, Jon Herlocker, and John Riedl , “Combining collaborative filtering with personal agents for better recommendations”, GroupLens Research Project Department of Computer Science and Engineering University of Minnesota Minneapolis, MN 55455.

[6] Stuart Edward Middleton,” Capturing knowledge of user preferences with recommender systems”, A thesis submitted for the degree of Doctor of Philosophy, Faculty of Engineering and Applied Science , Department of Electronics and Computer Science, May 2003.

[7] Mustafa Bilgic, Raymond J. Mooney, Elaine Rich “Explanation for recommender systems: satisfaction vs. promotion”, May 19, 2004.Gunther, O. and Spiekermann, S. RFID and the perception of control: the consumer’s view, *Communications of the ACM*, 48(9), September 2005, pp.73-76.

[8] Raymond J. Mooney, Loriene Roy, “Content-based book recommending using learning for text categorization”, Appears in Proceedings of the Fifth ACM Conference on Digital Libraries, pp.195-240, San Antonio, TX, June 2000.

[9] Cane Wing-ki Leung, Stephen Chi-fai Chan and Fu-lai C, “A collaborative filtering framework based on fuzzy association rules and multiple-level similarity”, Department of Computing, The Hong Kong Polytechnic University, To appear in Knowledge and Information Systems.

[10] Robin Burke, “Knowledge-based recommender systems”.

[11] Cane Wing-ki Leung, Stephen Chi-fai Chan, Fu-lai Chung, “An empirical study of a cross-level association rule mining approach to cold-start recommendations”, Department of Computing, The Hong Kong Polytechnic University, Hong Kong, Knowledge-Based Systems 21 (2008) 515–529.