Design and Implementation of Semantic and Content based Hybrid Recommender System for Java Programs

Rajesh K. Shukla
Research Scholor
RGPV, Bhopal, India,

Sanjay Silakari, PhD
Professor and Director
UIT, RGPV Bhopal

P.K. Chande, PhD
Ex Professor (IS)
India Indian Institute of Management, Indore

ABSTRACT

During the past few years the World Wide Web has emerged as the mainstream medium of communication and information dissemination. With the rapid growth of the WWW and the advent of eservices for online shopping, social networking, email and more; The Web personalization[10,11] and recommendation system has now become one of the most important tool for both Web-based organizations as well as for end users in order to extract the “right” and “interesting” information from the World Wide Web. Recommendation system (RS) is one of the most advanced approaches which are widely used for personalization of information on the web and information retrieval systems. Recommendation systems are now popular commercially as well as in Research community. Many major e-commerce Websites are already using recommendation systems to increase their customers by providing relevant suggestions to their customers and providing them better recommendation for purchasing of products. The recommendations could be based on various parameters, such as customer’s behavior of purchasing, rating and commenting; user characteristics such as geographical location or other demographic information.. In this paper we are proposing the design and implementation of a computer programs recommender system that recommends the user; Java programs; which are similar to program that a user is currently interested in. In order to achieve this, we have prepared a tag list, which is a list of keywords, packages and classes available in Java that have been used to match the program similarity with each other. With each program in database, a heading is associated which is displayed before user to choose one. Feature extraction is achieved by identifying tags available in program heading as well as in contents of a program. A threshold value (t) is also available which determines how similar a program should be in order to be recommended to the user. The proposed system can work in three different modes: Heading based recommendations, Content based recommendations and Mixed recommendations.

Keywords

Tag Based, Content Based, Recommendation, Systems

1. INTRODUCTION

Due to the exponential growth in volume of information available on the internet it has become more difficult to access relevant information from the Web. The explosive growth of the World Wide Web and the advent of e-services for online shopping, social networking, e-mail and more has made the Web personalization and Recommendation System as the indispensable tool for both Web-based organizations and for the end users in order to extract the “right” and “interesting” information from the World Wide Web. The Web Personalization and Recommendation system attempt to reduce information overload and it learns from the customer’s behavior of purchasing, rating and commenting and then recommends product(s) in which a user may be interested. It helps to build up a long lasting relationship with loyal users of website which helps in retaining the customers. The impact of personalization and recommendation system can be experienced by the rapid popularity that this area has gained in the last few years. Many of the Web portals are using Recommender system. Customers preferably choose to visit those websites, which understand their needs, provide them rapid value added customized services and easy access to required information in simple understandable format. Web personalization and recommendation system plays a major role to meet these goals. The Amazon.com, CD-NOW, eBay, Levis, Moviefinder.com, and Reel.com, Netflix etc are real world examples of the operation of industry-strength recommender systems 2 This paper is organized as follow: section.2 presents a brief overview of different approaches for recommendation systems. Proposed recommendation approach is described in section.3. It includes the outline, description and the algorithm of proposed approach for Java Program Recommendation System. Section.4 shows the experimental setup for evaluating the effectiveness of proposed recommendation system. Major findings, results and discussions are presented in section.5. Finally, section.6 concludes the paper

2. RECOMMENDER SYSTEM APPROACHES

The two main approaches that have been used in building the recommender system [8] are collaborative filtering (CF) [25] and content-based filtering. Now most of the current research involves employing a hybrid approach by combining the content-based and collaborative filtering approaches in order to leverage the strengths of both methods. Figure 1 illustrates the different types of recommendation approaches.

2.1 Collaborative Based Recommendations

Collaborative Based Recommendations Collaborative filtering technique was introduced by Goldberg et al (1992)[1]. It is considered to be one of the most successful approaches in recommendation systems. Collaborative Based Recommender systems accept collaborative filtering as one of the widely executed technique [2]. It analyzes the historical interactions by collecting the user feedback as database of preferences for items by users. A matrix of distances between different users or different items is created in the form of ratings for items in a given domain and a then the item is recommended to the user based on exploiting the similarities among the past ratings of all users collectively. The collaborative filtering technique uses the user ratings matrix either directly, or indirectly to induce a collaborative model. The user ratings matrix consists of rating data along with a set of users and items. Collaborative recommendation system identifies other users whose choices are similar to those of the given user and recommend the items they have liked. Therefore the users are recommended items similar to the user preference and interest in the past. The performance of the collaborative filtering approach relies on the available user preference data. The collaborative filtering technique can essentially be divided into three major categories [3, 4].

2.1.1 Memory-based approaches

This method is based on the assumption that if two users A and B act similarly on predicting the behavior of 'n' items then they will rate the other items similarly. This approach uses the entire collections or a sample of the previous ratings of user-item to generate the prediction for the other items. This method is also known as neighborhoodbased method because in this method every user is grouped with the people having similar interests. Memory based approaches can be classified into two User-based CF where the similarity between users are calculated by comparing their ratings on the same item so Recommendations are made based on users with similar characteristics Item-based CF where the similarity between two items is determined by comparing the rating made by same user on the items so Recommendations are based on similar items. 2.1.2 Model-based approaches In this method, the users' preferences are modeled by constructing a special prediction model [5]and the system is trained using various machine learning and data mining algorithms. It represents the user preference by a set of rating scores and the collection of ratings is used to learn a model of user preferences. The system provides recommendations by estimating parameters of statistical models for user ratings and based on the known model, makes prediction for test and real-world data. 2.1.3 Hybrid Collaborative Filtering These methods are used to make prediction by combining different Collaborative Filtering techniques such as collaborative and content-based methods in order to leverage the strengths of both methods. PHOAKS (People Helping One Another Know Stuff) is a purely collaborative recommender system. GroupLens[6] The GroupLens project is another purely collaborative approach to recommendation

2.2 Content Based Recommendation

Content-based Recommendation system is based on analysis and classifying content of the considered objects and its relation to the user's preferences so content based recommendation systems recommend an item similar to the ones that each user liked in the past, taking into account the description of the item previously rated by a user and a profile of the user's preferences The recommendation process basically consists of the description of items, user profiles and techniques to compare the attributes of the user profile against attributes of a items to identify what is the most suitable recommendation for a particular user. Content-based recommendation systems is used in various fields such as recommending web pages, television programs, news articles , items for sale etc. Landsend.com, the leading clothing company of the US, is a purely content based recommender system. 2.3 Hybrid approach: The hybrid recommendation system [9] combine collaborative and content-based methods in order to leverage the strengths of both methods and mitigating inherent limitations of either paradigm. In content-based approach, similar items to the ones the user preferred in past will be recommended to the user while in collaborative filtering, items that other people with similar tastes and preferences like will be recommended. NewsWeeder [7] is a Netnews filtering system that uses both content-based and collaborative filtering.

3. PROPOSED WORK

i. Outline

We have implemented a computer programs recommender system that recommends the user, Java programs which are similar to program that a user is currently interested in. For example if a user searches for a program on how to build up a JFrame in Java then the program related to SWING and GUI should be recommended to him because JFrame is a part of SWING library and is used to develop GUI applications in Java. In order to achieve this, we have prepared a tag list, which is a list of keywords, packages and classes available in Java that have been used to match the program similarity with each other. With each program in database, a heading is associated which is displayed before user to choose one. Feature extraction is achieved by identifying tags available in program heading as well as in contents of a program. A threshold value (t) is also available which determines how similar a program should be in order to be recommended to the user.

ii. Description

The proposed system can work in three different modes: Heading based recommendations, Content based recommendations and Mixed recommendations. In heading based recommendations, the tags available in the heading of program currently being searched by user are determined. These tags are then matched against the tags present in heading of all the other programs. If the number of tags matched is greater than t then the program is included in recommendation list. In content based recommendations, the tags available in the contents of program currently being searched by user are determined. These tags are then matched against the tags present in contents of all the other programs. If the number of tags matched is greater than 't' then the program is included in recommendation list. In mixed recommendations, the tags available in the contents as well as heading of program currently being searched by user are determined. These tags are then matched against the tags present in heading as well as contents of all the other programs. If the number of tags matched is greater than t then the program is included in recommendation list.

iii. Algorithm

Feature Extraction Algorithm (i) Read the heading and contents of each program (ii) Discard common words from headings and comments from programs (iii) Match each of the remaining word against the tags list (iv) If it matches with any tag then Check whether it is a part of heading or content of a program If it is a part of heading then Add the word in Heading_ Features of that program Else Add the word in Content_Features of that program Recommendation Algorithm (i) Let the user search a program of his interest (ii) Set the Recommendation_List ={Φ}

(iii) Check the mode of running, which can be either Content_Based, Headin_ Based or Mixed (iv) Get the threshold value and store in variable t (v) If mode is Heading based then Add all those programs to Recommendation_List whose at least t Heading_Features matches with that of program searched by user Else If mode is Content based then Add all those programs to Recommendation_List whose at least t Content_Features matches with that of program searched by user Else If mode is Mixed then Add all those programs to Recommendation_List whose at least t (Content_Features ∪ Heading_Features) matches with that of program searched by user (vi) Recommend all the programs present in Recommendation_List to user

4. EXPERIMENTAL SETUP

The recommendations heavily depend upon threshold value and a wrong threshold may disturb the whole recommendations therefore we conducted a survey to find out the optimal value of t. The survey was done with 200 computer science students in order to find out at what value of t, the students find recommendations valuable.

5. FINDINGS, RESULTS AND DISCUSSIONS

From survey, we found that when t= 1, 2, 6, 7 and 8, more than 75% of users were not satisfied. It is because when t=1 or 2, the recommendation list contains too many recommendations as there are some basic tags which can be found in almost every Java program. These tags always match with other programs making the recommendations worthless. Thus when threshold is too low (i.e. 1 or 2) then the algorithm suggests too many recommendations out of which, many are irrelevant. Similarly, when threshold is too high (i.e. 6, 7 or 8) then the algorithm may suggest too few recommendations because it is hard to match so many tags in small sample programs. According to the survey that we carried out, ideal value of t were between 3 and 5 as with these values, most of the users were satisfied and found the recommendations valuable. The percentage of users satisfied with different similarity filters when t=3, 4 and 5 as well as the percentage of users who were found unsatisfied with all the three filters are shown in Table.1. Table.1. Percentage of users satisfied with different recommendation criteria when threshold t=3, 4 and 5

Threshold Value (t)	Percentage of users satisfied with Heading Based Filter (%)	Percentage of users satisfied with Content Based Filter (%)	Percentage of users satisfied with Mixed Filter (%)	Percentage of unsatisfied users (%)
3	36	73	76	24
4	27	86	88	12
5	18	93	94	6

It should be noted that the percentage of satisfied users is maximum for mixed filter as compared to heading based and content based filters. The pictorial representation of table 1 is given in figure 1(a), 1(b) and 1(c)

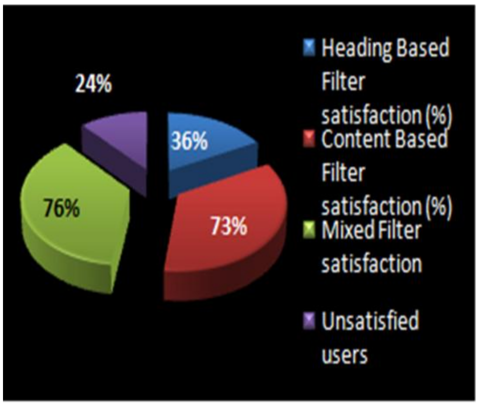

Figure 1 (a)

When the Threshold Value (t)=3 36% of users satisfied with Heading Based Filter 73% of users satisfied with Content Based Filter 76% of users satisfied with Mixed Filter 24% of unsatisfied users

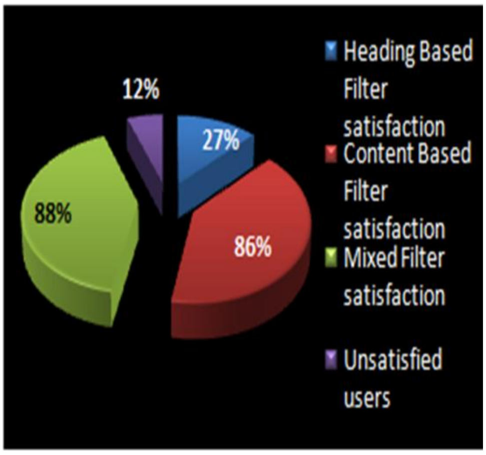

Figure 1 (b)

When the Threshold Value (t)=4 27% of users satisfied with Heading Based Filter 86% of users satisfied with Content Based Filter 88% of users satisfied with Mixed Filter 12% of unsatisfied users

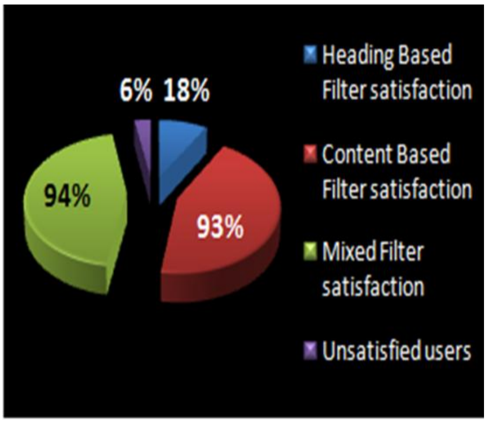

Figure 1 (c)

When the Threshold Value (t)=3 18% of users satisfied with Heading Based Filter 93% of users satisfied with Content Based Filter 94% of users satisfied with Mixed Filter 6% of unsatisfied users.

6. CONCLUSION

Recommender System is a most advanced mass customization for web portals that increase Ecommerce. It is increasingly adapted in future, as globalization increases value to the customers for their Business. Recommending previously was a very complex way but as internet is making whole world small such techniques are highly effective for users in judging products according to their tastes. In this paper, we implemented a “Java Program Recommondation System” based on three different modes. When the threshold is too low (i.e. 1 or 2) then the algorithm suggests too many recommendations out of which, many are irrelevant. Similarly, when threshold is too high (i.e. 6, 7 or 8) then the algorithm may suggest too few recommendations because it is hard to match so many tags in small sample programs. According to the survey that we carried out, ideal value of t were between 3 and 5 as with these values, most of the users were satisfied and found the recommendations valuable. Here it is to be noted that the percentage of satisfied users is maximum for mixed filter as compared to heading based and content based filters It also discussed the classification of different Recommendation system and their application in web portals. 7.

7. REFERENCES

[1] Goldberg, D., Nichols, D., Oki, B.M., Terry, D. “Using collaborative filtering to weave an information tapestry” Commun.ACM 35(12), 1992, pp. 61–70.

[2] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, andJ. Riedl. GroupLens “An Open Architecture for Collaborative Filtering of Netnews” In Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work, 1994

[3] Badrul Sarwar, George Karypis, and Joseph Konstan “Item-based Collaborative Filtering Recommendation Algorithms” Proceedings of the 10th, pages 285–295, 2001.

[4] Xiaoyuan Su and Taghi M. Khoshgoftaar “A Survey of Collaborative Filtering Techniques. Advances in Artificial Intelligence”, 2009 (Section 3):1–19, 2009.

[5] G. Adomavicius and A. Tuzhilin. “Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions” IEEE Transactions on Knowledge and Data Engineering, 17(6):734–749, June 2005. [6] “GroupLens: Applying Collaborative Filtering to Usenet News” Communications of the ACM,40(3), pp. 77-87.

[6] Lang, K. (1995) “NewsWeeder: Learning to filter Netnews” Proceedings of the 12th International Conference on Machine Learning, pp: 331-339.

[7] Resnick, P., Varian, H.R. “Recommender systems”Communications of the ACM 40(3), 1997, pp. 56– 58.

[8] Robin Burke “Hybrid Recommender Systems: Survey and Experiments, User Modeling and UserAdapted Interaction” 12, 331-370 (2002)

[9] Anand, S.S., Mobasher, B. “Intelligent techniques for web personalization.” In: Intelligent Techniques for WebPersonalization, 2005, pp. 1–36., Springer.

[10] Anand, S.S., Mobasher, B. “Introduction to intelligent techniques for web personalization”, ACM Trans. Interet Technol. 7(4), 2007

[11] Zi-Ke Zhang, Tao Zhou and Yi-Cheng Zhang "Tag-Aware Recommender Systems: A State-of-theArt Survey" Springer, Journal of Computer Science and Technology, Vol.26, No.5, 2011, pp 767-77