Book Recommendation System

Shuchit Gandhi*, Ananta Soneji†, Ankul Jain‡, Shreyansh Jain§
Information and Communication Technology
minors in Computational Sciences
DA-IICT
Email: {201301441*, 201301434†, 201301444‡, 201301443§}@daiict.ac.in

***Abstract*—An important activity undertaken by many individuals is literary reading. Choosing to read a book can be a long time commitment and hence making book choice becomes an important task for book lovers and public library users. In this paper, we try to present a model for a personalized hybrid book recommendation system which exploits varied aspects of giving recommendations apart from the regular collaborative and content-based filtering approaches. Temporal aspects for the recommendations are incorporated. Also for users of different age, gender and country, personalized recommendations can be made on these demographic parameters. Scraping information from the web and using the information obtained from this process can be equally useful in making recommendations. The overall architecture of the proposed system is presented and its implementation with a prototype design is described which satisfies a user by providing best and efficient books recommendations.**

I. Introduction

Literary reading is an important task for people all around the world. Public libraries make it possible to exercise this activity for free, by letting users borrow books for one or two weeks. In this context, selecting the right book for reading purpose becomes an important task, because it can save the reader unnecessary trips to the library to exchange books. Recommendation systems is one such technology that help users by providing them with relevant suggestions based on their profile.

A Recommendation Engine, in actual definition can be referred to as a system that can run on clustered / non clustered environment taking users' on-line footprint as one of its input set and generating a probable footprint for the user thereby providing its users a prediction closer to reality. A user profile is generated on the basis of user navigation history and his similarity with other users. Recommendation System gives a list of recommendations to the user which is an attempt of predicting user's preferences. A variety of techniques have been proposed till today for performing recommendations. The techniques such as collaborative filtering, content-based filtering, knowledge-based and demographic filtering are used for recommendations. Sometimes, the features of these techniques are combined in hybrid recommendation system to improve the performance of recommendation engine.

The system we propose takes into account the personalization of the recommendations. Demographic recommendations are a good way of giving personalized recommendations. Filtering the results of a collaborative approach is a good way of making better recommendations. Recommendations suited to the user's age, region, gender can be made to make them more personalized. The cold start problem is a major issue in many recommendation systems. In such a scenario, the system is unable to give appropriate predictions until it has a better idea about the user's preferences. Demographic recommendations could help alleviate this problem to some extent, if not entirely in case of a newly added user. A user always would like to stay abreast of the most popular books in a particular category. The traditional filtering techniques may not always be able to keep a user updated about the recent trends in books. Web scraping, could be of major help to users with such preferences. Temporal aspects are of equal importance when it comes to recommendations for books. Old ratings and recommendations often become obsolete and lead to false predictions. Hence, a time-stamp attribute for every rating the user gives is necessary. Hence the ratings repository remains updated and obsolete recommendations are filtered out of the system over a period of time.

II. Existing Book Recommendation System

Following two are some of the existing book recommendation engines that make use of conventional algorithms for recommendations.

In Content based Recommendation Engine, system provide with recommendations based on the features associated with books such as genre, author, etc and the user's ratings. Content-based recommenders treat recommendation as a classification problem and learn a classifier for the user's likes and dislikes based on product features. In this, system requires the additional data about the context of item consumption like time, mood and behavioral aspects. These data may be used to improve the recommendation compared to what could be performed without this additional source of information.

In Collaborative recommendation engines, suggestions are given on the basis of ratings given by group of people. It locates peer users with a rating history similar to the current user and generates recommendations for the user. The two commonly accepted techniques are Item-based collaborative filtering and User-based collaborative filtering. In Item-based recommendations, the similarity between items is taken into account and then predictions are made. Whereas in the latter, users with similar tastes are found and on the basis of their ratings, predictions are made. Different algorithms like Cosine Similarity Measure, Pearson Correlation Similarity Measure are used for the same.

III. Proposed Recommendation System

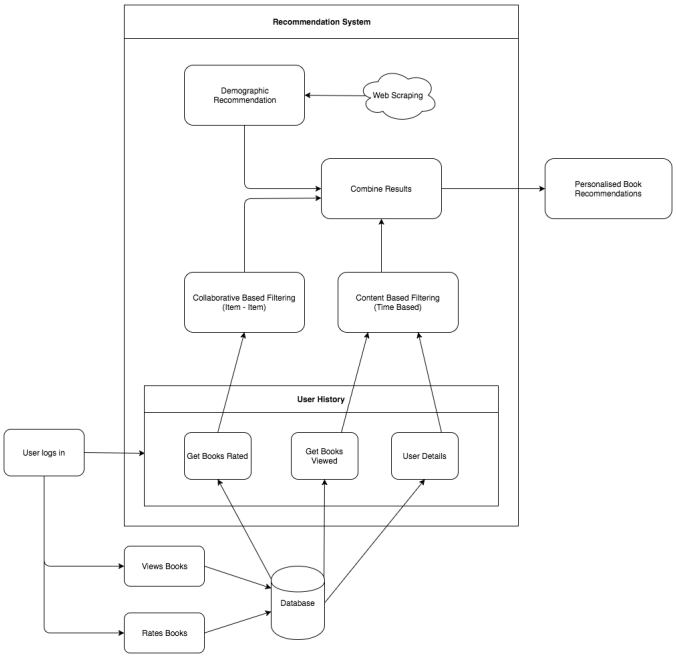

Fig. 1. Architecture of the Recommendation System

The above Figure 1 describes the architecture of proposed system. As shown, the main module in this system is Recommendation system. The registered user logs in to the system. The user can view books of different categories. The user can also rate books as per his/her likings. The rating and searching history of books for each individual is stored in the database. In recommendation system module, mainly three techniques are used for recommendations. Collaborative based filtering and content based filtering techniques are performed on the data which is present in user's history. To address the problem of slow start that arises because of these techniques, demographic recommender is used. The results from all the recommender techniques are combined and the list of recommended books is generated.

Let us consider a scenario where the users' ratings is already stored in the database. Now it's time to provide him with certain recommendations. Two major kinds of filtering techniques would be applied to give recommendations: Collaborative and Content. The collaborative filtering uses item-item similarity which is explained in detail further in the next section. Content-based filtering incorporates an another dimension so as to give recommendation that gets updated with time. This technique uses the temporal dimension for further curating the results. Next, a filtering procedure based on the demographic aspects of the user like his age, gender etc. is applied that helps in making the recommendations more personalized. Also, a popular set of books is obtained using web scraping techniques while applying the demographic filtering procedure.

IV. Filtering Techniques

The Recommendation techniques are classified into five types:

1) Collaborative.
2) Content based.
3) Demographic.
4) Utility based.
5) Knowledge based.

For the proposed book recommendation engine, collaborative, content-based and demographic techniques are used. An overview of these recommendation techniques is presented in the following Table I. For this, assume that I is the set of items over which recommendations are to be made, U is the set of users whose preferences are known, u is the user to which recommendations need to be provided, and i is some item for which we would like to predict u's preference.

TABLE I
Filtering Techniques

Network	Information Required	Input	Process
Collaborative	U's ratings for items in I	u's ratings for items in I	Identify users in U similar to u, and predict from their ratings, u's rating for i.
Content-based	Feature vectors for items in I	u's ratings for items in I	Obtain a classifier that fits u's ratings and thus, predict rating of i.
Demographic	User profile of U	Demographic information of u	Identify users similar to u demographically, and from their ratings, predict u's rating for i.

A. Collaborative filtering

Collaborative filtering is mainly of two major types: Memory-based and Model-based. Memory based is further of two types: Item-Item Similarity and User-User Similarity. The similarity values between items are measured by observing all the users who have rated both the items. As shown in the diagram below, the similarity between two items is dependent upon the ratings given to the items by users who have rated both of them.

Slope One is a family of algorithms used for collaborative filtering. It is the simplest form of non-trivial item-based collaborative filtering based on ratings. They are extremely easy to implement because of their simplicity. Moreover, their accuracy is often on par with algorithms that are both, more complicated and computationally expensive. When item ratings are available, such as is the case when people are given the option of rating books, collaborative filtering aims to predict the ratings of one individual based on his past ratings and on a (large) database of ratings contributed by other users.

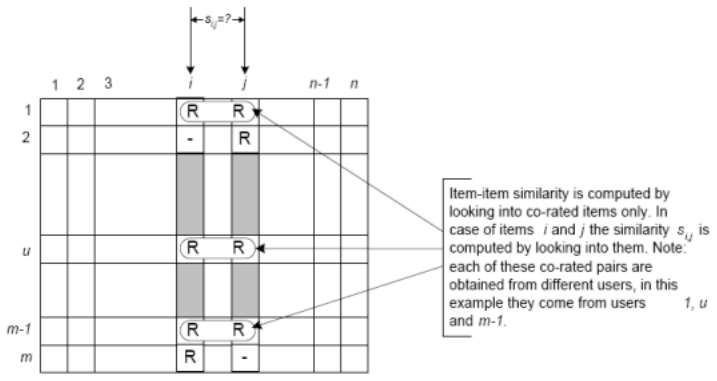

Fig. 2. Item-item Similarity

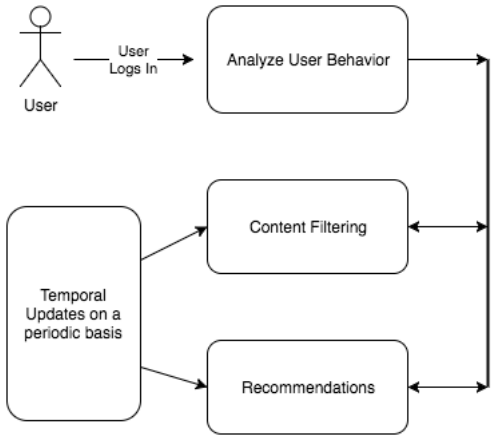

Fig. 3. Time based content filtering

B. Content-based filtering

As a matter of fact, collaborative filtering is a widely used methodology in most of the recommendation systems. Even though, content-based technique can have leverage over collaborative filtering when there are some characteristic values that represent the contents of an item and this is the case with books.

The proposed system adds another important dimension to content-based filtering which is the temporal dimension. This dimension takes into account the number of time an item is liked/viewed by the user over a period of time. A counter is stored for each item which is updated whenever a user checks that items in its favourite list. The greater the counter for any particular book, the more are the chances of it being recommended. Moreover instead of using counter rating of similar user, the algorithm uses counter ratings of all users which indirectly improves diversity of the recommendation process. Over a period of time, the counter for different items will change in relation to other items which would either become more favourite or less favourite in comparison to each other. Thus, the overall system reflects an updated favourite list with respect to a particular category at any given point of time. User preferences are incorporated through content based filtering where only a particular category of book will be recommended pertaining to that particular user.

Figure 3 describes time based content filtering.

The above mentioned process ensures that recommended items remains relevant to the changing user preferences. This approach combines user choices with the most widely visited links over a period of time and thus gives a set of diverse as well as most widely preferred books. The timely updation of recommendation is a very important aspect of the recommendation system as user needs as well as the system content keeps changing with time.

C. Demographic-based filtering

As stated in [3], multiple levels of filtering help give better recommendations. Demographic filtering combined with results of content and collaborative filtering, results in a better experience. Demographic recommender systems aim to categorize the user based on personal attributes and make recommendations based on demographic classes.

Clustering techniques prove to be extremely advantageous in improving the efficiency of recommending algorithms. We cluster users based on their demographic aspects instead of taking the entire set of users registered with the website as a training set. For example, men and women usually tend to have their own respective choices of books which are often different from each other. Hence, it would be wise to create different clusters based on their gender. Also, age can be another attribute that can be considered for clustering purpose. This will help suggesting books to kids, teenagers and adults pertaining to their age groups.

Another parameter that can be helpful for filtering is location. This can be mainly done in two ways - Clustering of users on the basis of their location and clustering of books according to the origin of the author. Native authors are usually preferred over foreign ones due to the ease of understanding the language. As a result, filtering can then be performed only on the clusters rather than on the entire data-set. Thus the demographic parameters can be of great use in personalization of recommendations and reducing the time spent on processing by use of clusters.

Further, data for demographic-based filtering can be scraped from the web. The following Figure 4 describes the process of demographic filtering through data scraped from the web.

One of the major issues faced by the recommender systems is the cold start problem. This technique comes in handy while addressing this issue. The cold start problem is faced by users when they are new to the system. Being new to the platform, the engine has almost no knowledge about the user and hence it is unable to recommend books to them at such an early stage. The system needs to be patient and wait until the user has rated at least a few books. The benefit of a demographic approach is that it may not require a history of user ratings of the type needed by collaborative and content-based techniques. Later on as time progresses, the other filtering techniques come into picture as the system has enough knowledge about the user by then.

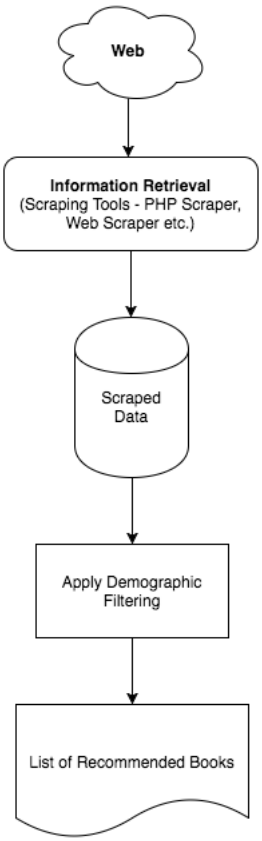

Fig. 4. Demographic-based filtering

D. Combining Results

Finally the results from above mentioned three filtering techniques can be combined in one of the following ways -

- Weighted: The score of different recommendation components are combined numerically.
- Switching: The engine chooses among recommendation components and applies the selected one.
- Mixed: Recommendations from different recommenders are presented together.

V. Conclusion and Future Scope

Apart from just the conventional Collaborative and Content-based filtering techniques, hybrid algorithms that combine results from two or more techniques give better and more personalized recommendations to the users. The proposed system tries to give diverse recommendations to the users which are also relevant to his current information needs. This is achieved through time based content filtering. In conventional filtering techniques, after a certain amount of time, the accuracy of the recommendation process gets stagnant and users keep on receiving similar books as recommendations. Further accuracy can only be increased by taking into account the temporal dimension along with content-based.

Next, demographic filtering through web scraped data helps solve the problem of slow-start faced by major recommendation systems. This process helps suggest books to the new users on the basis of user's demographic details like age, gender, location, etc and also recommends trending books in the region.

This hybrid recommender system is more accurate and efficient as it takes into consideration the features of various recommendation techniques. The book recommendation engine will reduce the overhead associated with making the best choices of books among the plenty.

In the future, list obtained through collaborative filtering can be aggregated with a book list expanded on the basis of author preferences i.e. Authors are used to improve the book recommendations through a fusion approach [4].

References

[1] Chhavi Rana, Sanjay Kumar Jain Maria Schuld, Ilya Sinayskiy and Francesco Petruccione *Building a Book Recommender system using time based content filtering*. WSEAS Tansactions on Computers

[2] Francesco Ricci, Lior Rokach and Bracha Shapira *Introduction to Recommender Systems Handbook* Recommender Systems Handbook, Springer, 2011, pp. 1-35

[3] Robin Burke *Hybrid Web Recommender Systems* Lecture Notes in Computer Science, Springer-Verlag, Vol. 4321, May 2007, pp. 377-408

[4] Paula Cristina Vaz, David Martins de Matos, Bruno Martins, Pavel Calado *Improving an Hybrid Literary Book Recommendation System through Author Ranking* Information Retrieval, 23 Mar 2012.