

A Proposed Hybrid Book Recommender System

Suhas Patil¹, Dr. Varsha Namdeo²

¹M.Tech Scholar, Department of Computer Science & Engineering, RKDF IST, Bhopal, RGPV University, India

²Associate Professor, Department of Computer Science & Engineering, RKDF IST, Bhopal, RGPV University, India

Abstract - Now-a-days, many major e-commerce Websites are using recommendation systems to provide relevant suggestions to their customers. The recommendations could be based on various parameters, such as items popular on the company's Website; user characteristics such as geographical location or other demographic information; or past buying behavior of top customers. In this paper, a book recommendation engine is proposed which uses data mining techniques for recommending books. The proposed recommender system will give its users' the ability to view and search books as well as novels which will be used to draw out conclusions about the stream of a user and the genre of the books liked by that user. The system will analyze the user behavior by using the features of various recommendation techniques such as content based; collaborative and demographic. Thus, in this paper a hybrid recommendation system is proposed which satisfies a user by providing best and efficient books recommendations.

Keywords: Collaborative Technique, Content Based Technique, Recommendation Engine, User's Interest.

I. INTRODUCTION (10pt, Caps, normal)

A Recommendation Engine, in actual definition can be referred to as a system that can run on clustered / non-clustered environment taking user online footprint as one of its input set and generating a probable footprint for the user thereby providing its users a prediction closer to reality. Recommendation Engines require a large dataset and a fast computing system that can perform analytics on the same within fraction of seconds. Recommendation Engines, in simpler terms are programs that are data intensive and involve complex pattern matching on a set of predefined parameters and they become efficient with the increase in the size of the content being fed to them. Recommender systems represent user preferences for the purpose of suggesting items to purchase or examine. They have become fundamental applications in electronic commerce and information access, providing suggestions that effectively prune large information spaces so that users are directed toward those items that best meet their needs and preferences. A variety of techniques have been proposed till today for performing recommendations. The techniques such as content-based, collaborative, knowledge-based and demographic are used for recommendations. Sometimes, the features of these techniques are combined in hybrid recommenders to improve the performance of recommendation engine.

In the proposed book recommendation engine, books will be displayed according to the readers' preferences in a hierarchical way to categorize readers' interest in different genres, the users' pattern of searching different books and to form an effective set of rules based on that. New books will be appropriately presented according to users' needs. Based on users' interest and books properties, a book recommendation system will be generating best and efficient book recommendations.

II. LITERATURE SURVEY

Following are some of the existing book recommendation engines used by the top-rated book purchasing websites. The existing engines make use of conventional algorithms for recommendations.

In Content based Recommendation Engine, system generates recommendations from source based on the features associated with products and the user's information. Content-based recommenders treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on product features. In Collaborative recommendation engines, suggestions are generated 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. In Context based Recommendation Engine, system requires the additional data about the context of item consumption like time, mood and behavioural aspects. These data may be used to improve the recommendation compared to what could be performed without this additional source of information.

III. PROPOSED SYSTEM

In above figure 1, the architecture of proposed system is shown. The main module in this system is Recommender system. The registered user logs in to the system. The user can view books of different categories. The issuing and searching history of books for each individual is stored in the database. The proposed system has two user roles. First is student / staff and second one is admin. The user can view all books, request a particular book, return a book, view recommended books. The system admin can perform all the crude operations such as adding new books, updating existing books and maintaining books database. In recommender system module, mainly two

techniques are used for recommendations to develop a hybrid set of recommended books. Collaborative based filtering and content based filtering techniques are performed on the data which is present in user's history. The results from both the recommender techniques are combined and the set for recommended books is generated. Hence the proposed system is called a hybrid book recommendation system.

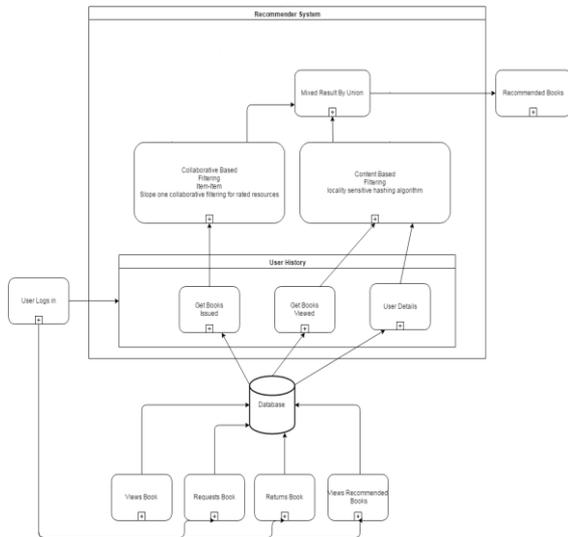


Fig. 3.1 System Architecture

IV. TECHNIQUES USED

Recommendation techniques have a number of possible classifications. The classification is based on the sources of data on which recommendation is based and the use to which that data is put. In general, recommender systems have (i) background data, the information that the system has before the recommendation process begins, (ii) input data, the information that user must communicate to the system in order to generate a recommendation, and (iii) an algorithm that combines background and input data to arrive at its suggestions. The recommendation techniques are classified into five types: 1] Collaborative. 2] Content based. 3] Demographic. 4] Utility based. 5] Knowledge based. In this paper, for the proposed book recommendation engine collaborative, content-based are used.

TABLE 1. RECOMMENDATION TECHNIQUES

Technique	Background	Input	Process
Collaborative	Ratings from U of items in I.	Ratings from u of items in I.	Identify users in U similar to u, and extrapolate from their ratings of i.
Content-based	Features of items in I	u's ratings of items in I	Generate a classifier that fits u's rating behavior and use it on i.

Following Table 1 gives overview of the recommendation techniques. Assume that I is the set of items over which recommendations might be made, U is the set of users

whose preferences are known, u is the user for whom recommendations need to be generated, and i is some item for which we would like to predict u's preference.

1. Content Based Technique

This approach relies on creating a plethora of parameters to describe a product 'P'. Considering a smart phone as an example the possible parameters could be screen size, image quality, Wi-Fi protocols, brand names, operating systems etc. The larger the parameter set the better and easier it is to match patterns with user profile and his online footprint. The parameters can then be assigned weights and hence a relative priority is set for each of the parameter. All these parameters are then used to create a user profile and each time a prospective user checks out another product, his profile gets updated. Hence we see that the system learns about the user preferences and selection patters by his online footprint. Popular platforms that use such an approach are IMDB and Pandora.

For Content based technique, Locality-sensitive hashing method is used. Locality-sensitive hashing (LSH) is a method of performing probabilistic dimension reduction of high-dimensional data. The basic idea is to hash the input items so that similar items are mapped to the same buckets with high probability. In LSH the goal is to maximize probability of "collision" of similar items. Jaccard similarity is used along with LSH method.

Locally Sensitive Hashing (LSH) Technique or MinHash Algorithm has following steps:-

1. The Recommendation is done on the basis of the mapping between the attributes of user profile and the attributes of book profile stored in the database to compute the Jaccard's Coefficient score.
2. Based on the Jaccard's Coefficient score, the user is provided with the appropriate list of books.
3. First we start by placing user's attributes in set A and set B consisting of university attribute.
4. Then we need to merge set A and set B into one single list.
5. Next step is generate a hash array using self defined hash function. The hash array will have size equal to the size of merge list.
6. A matrix is generated whose size of row is equal to number of sets and the size of column is equal to the size of merge list. Initial the values in the matrix will be a MAX integer.

7. Now filling of first row of the matrix is done by comparing the first element of merge list with the elements of the set A. We are performing content based comparison. If the first element of merge list matches with any element of set A then the hash value corresponding to the first element of merge list is placed in the place of first MAX value of first row. Then for filling second row of the matrix, again the merge list elements and set B elements are compared and corresponding hash values are placed if a match is found.

8. The next step is to compare column-wise hash values if two same hash values are found then the match counter is incremented.

9. Finally Jaccard's Coefficient (JC) score is calculated as:

$JC = \frac{\text{the value of match counter}}{\text{by the size of the column in matrix}}$

2. Collaborative Technique

Collaborative recommendation is probably the most familiar, most widely implemented and most mature of the technologies. Collaborative recommender systems aggregate ratings or recommendations of objects, recognize commonalities between users on the basis of their ratings, and generate new recommendations based on inter-user comparisons. A typical user profile in a collaborative system consists of a vector of items and their ratings, continuously augmented as the user interacts with the system over time. Some systems used time-based discounting of ratings to account for drift in user interests. In some cases, ratings may be binary (like/dislike) or real-valued indicating degree of preference. Some of the most important systems using this technique are GroupLens/NetPerceptions, Tapestry and Recommender. These systems can be either memory based, comparing users against each other directly using correlation or other measures, or model-based, in which a model is derived from the historical rating data and used to make predictions. The greatest strength of collaborative techniques is that they are completely independent of any machine-readable representation of the objects being recommended, and work well for complex objects such as music and movies where variations in taste are responsible for much of the variation in preferences.

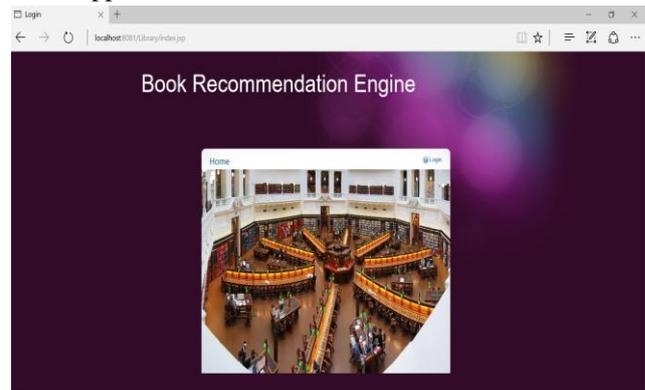
Slope One is a family of algorithms used for collaborative filtering. It is the simplest form of non-trivial item-based collaborative filtering based on ratings. Their simplicity makes it especially easy to implement them efficiently while their accuracy is often on par with more complicated and computationally expensive algorithms. When ratings of items are available, such as is the case when people are given the option of ratings resources (between 1 and 5, for

example), 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.

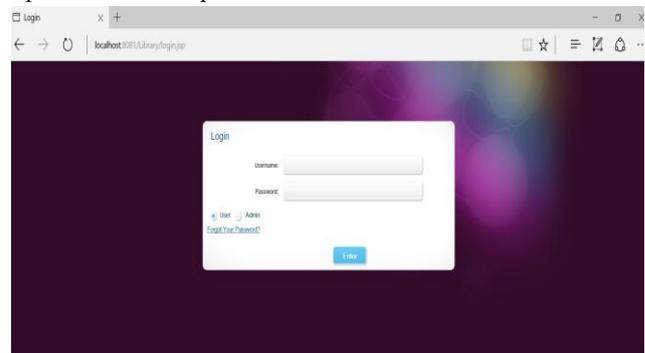
V. SYSTEM RESULTS

Following are some of the screenshots of the implemented hybrid book recommendation system.

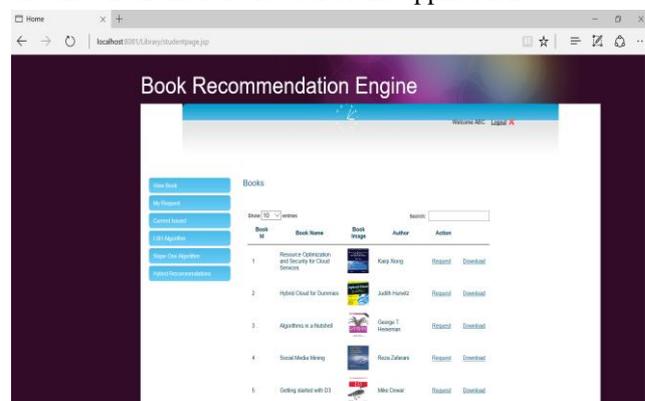
1] The screenshot shows the first page i.e. welcome page of the application.



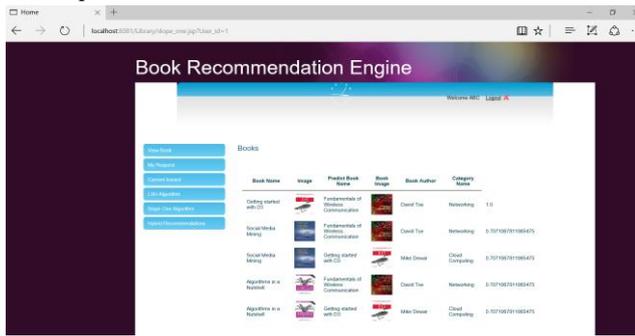
2] The screenshot shows the login screen of the application. Depending upon the role of the user login operation will be performed.



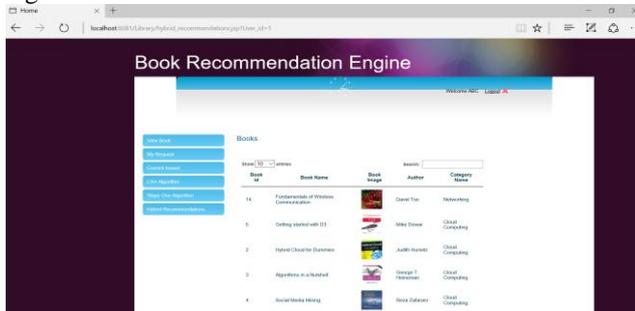
3] The screenshot shows the landing page after successful login of student. The different services available to student are shown on the left side bar of the application.



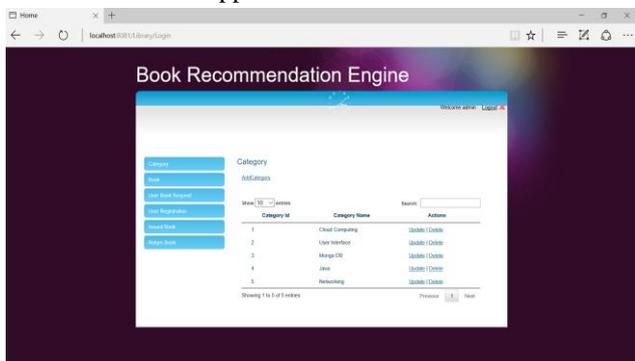
4] The screenshot shows the list of recommended books generated by applying slope one recommendation technique.



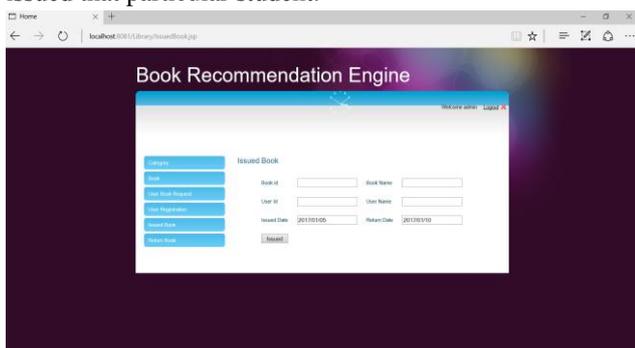
5] The screenshot shows the list of books obtained by combining the results of slope one and min hash algorithms.



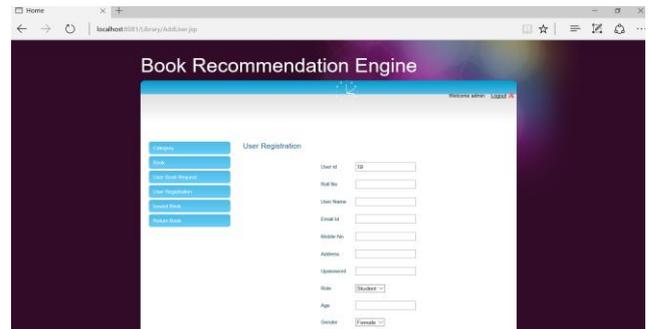
6] The screenshot shows the landing page of Admin. The different operations available to admin are shown on the left side bar of the application.



7] The screenshot shows the form for book issue which the admin uses to issue a requested book to the student. On successful submission of this form the requested book is issued that particular student.



8] The screenshot shows the form for new registration of student. The admin will register the details of new student in the application and will share the login credentials with student.



VI. CONCLUSION

Recommender systems are an extremely potent tool utilized to assist the selection process easier for users. The implemented book recommendation engine is a competent system to recommend Books for e-users. This recommender system will definitely be a great web application implemented in Java language. Such type of web application will be proved beneficial for today's high demanding online purchasing web sites. This hybrid recommender system is more accurate and efficient as it combines 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.

REFERENCES

- [1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Trans. Knowl. Data Eng.
- [2] G. Linden, B. Smith, and J. York, "Amazon recommendations: Item-to-item collaborative filtering," IEEE Internet Comput., Feb. 2003.
- [3] Michael Hashler, "Recommender Lab: A Framework for Developing and Testing Recommendation Algorithms" Nov. 2011.
- [4] R. Bell, Y. Koren, and C. Volinsky, "Modeling relationships at multiple scales to improve accuracy of large recommender systems" KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, New York, NY, USA, 2007, ACM.
- [5] O. Celma and P. Herrera, "A new approach to evaluating novel recommendations", RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems, New York, NY, USA, 2008, ACM.
- [6] C. N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification": Proceedings of the 14th international conference on World Wide Web, New York, USA, 2005, ACM.

- [7] Robin Burke, “Hybrid Recommender Systems: Survey and Experiments”, California State University, Fullerton Department of Information Systems and Decision Sciences.
- [8] Dharmendra Pathak, Sandeep Matharia and C. N. S. Murthy, “NOVA: Hybrid Book Recommendation Engine”, IEEE, 2012.
- [9] Ahu Sieg, Bamshad Mobasher, Robin Burke, “Ontology-Based Collaborative Recommendation”..