Rec-Tech: A Recommendation System encapsulated in Online Technical Forum

Kujur Preeti Nancy[1], Pal Arnab[2], Pooja[3]

[1,2,3] ESL, Chandernagore Lab
CA-5 Saltlake City, Kolkata - 64

Abstract- In the present days of e-commerce and social engineering the use of recommendation system is one of the most relevant issue. Several IT giants like Amazon, Facebook use highly developed recommender system to serve the users in a better method. So, Recommendation Algorithms are one of the most important topics in Data Science and Social Engineering. Here we used a recommendation system in a Technological Discussion Forum where users can post any relevant question related to different genres like, Programming Languages, Operating System, Artificial Intelligence etc. and we are building a recommendation system based on the user activities. With the advancement in the process of learning via Internet lead learners to an Internet Forums. As they are getting interested in Group Discussions, it's the best and huge platform to share our knowledge among different perspective to make it more lucid. In this paper, we have created a website, Technical Forum where our main objective is to make Recommendation System operative with our own algorithms.

Index Terms- Recommender System, Data Mining, JSP, Web Development, Database, Artificial Intelligence

I. INTRODUCTION

We intent to create a technical forum which is a General Discussion website emphasizes on various topics of Computer Science only accessible to pre-registered user. We have rendered facilities to the registered user as they can create any technical conversational thread, may post their view for any thread. Moreover, we aim on creating a Recommendation System which basically mean we hand over them updates on their area of interest in various ways by saving their searches in a Database.
We have operated on various languages and technology:--
1. HTML (Hyper Text Markup Language)
2. CSS (Cascading Style Sheet)
3. JAVASCRIPT
4. MYSQL
5. JSP (Java Server Page)

HTML:
* The main markup language for creating web pages and other information that can be displayed in a web browser.
* Provide a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, images and other items.
* Embed script written in language such as JavaScript which affect behaviour of HTML web pages.

CSS:
* Style sheet language used for describing the look and formatting of a document written in a markup language.
* Provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content.
* Allow the web page to display differently depending on the screen size or device on which it is being viewed.

JAVASCRIPT:
* Prototype-based scripting language with dynamic typing has first class functions. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.
* Enables programmatic access to computational objects within a host environment.

MYSQL:
* MySQL is open-source relational database management system.

JSP:
* A technology that helps software developers create dynamically generated web pages based on HTML, XML, or other document types. It uses the java programming language.
* To deploy and run Java Server Page, a compatible web server with a servlet container is required and here, we have used Apache Tomcat.
* JSP pages used delimiters for scripting functions. A script is fragment of Java code that is run when the user requests the page.

II. MOTIVATION

As students and teachers are depending more and more on internet for their various educational purpose which involve searches for the website such as forums which may have reliable information for their problems. Furthermore, to interact to the large community of members having same interest which they can easily find in forum websites where people with different location and with different perspective discuss on any topic where the nature of topic depend on type of forum. Thus, we aim to provide them the proper discussion platform as well as recommending them various books and various conversational thread in order to help them to enhance their knowledge.
We also put our conscious effort to make the recommendation system effective and efficient by introducing our own algorithm

behind it. The methodology section describes the algorithm in detail with mathematical notations.

III. METHODOLOGY

In the following section we are going to describe the methodology of the project we have done:
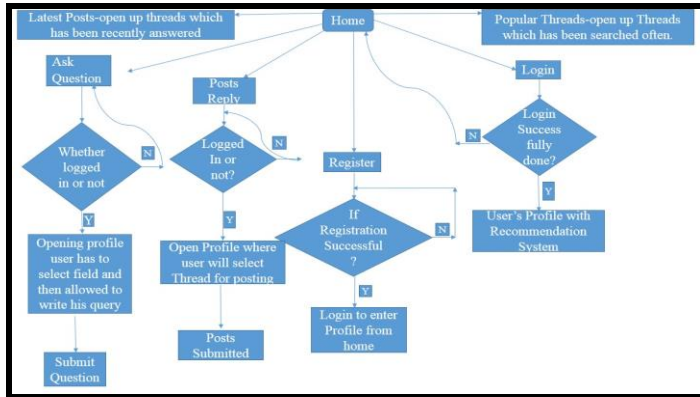
A. Flow Diagram

Figure 1: Flow Diagram for the Home Page

The Figure 1 describes the flow diagrams of the Home Page that comes to the user at the very beginning. At first the system checks that whether any user is logged in or not. If someone is logged in then we are allowing him or her to proceed with the forum. Otherwise it prompts the user to login.
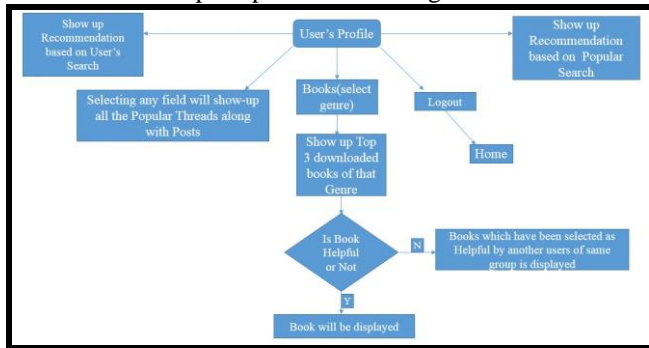

Figure 2: Flow Diagram for the User Profile

The Figure 2 describes the flow diagrams of the User Profile Page.

As the website is completely login-based, visitors have to register for being profited from the features of website. Popular posts are generally the threads which have the highest priorities among all the fields (searched maximum times on the website) whereas Latest Posts are the conversational threads which has been started recently on the website. Besides, When a user have to ask any query or want to post for any conversational thread then he has to get himself logged in, as soon as he is logged in he can proceed further through his Profile. Each time when any user submit his query, a new file is written each time named after genre name together with Question ID.

When user login to his profile he get Recommendation on the basis of his previously performed work ,that is, information about the field he is mostly interested in and additionally, Recommendation on the basis popular searches so that to let the user know about popular discussions going through websites.

Moreover, we are providing books to them as well as we are asking user whether the book is helpful or not, if the user don't find it helpful then we will show him the books which has been found helpful by the user of his own group. Here, group primarily means we have divided all users on the basis of their same area of interest.

B. Algorithms

In the primary level, we recommended the users with the top popular threads from the top popular genres from the table where questions are stored. In that table, there is a column of Priority which is being incremented each time the user click in the thread. So in the cold start level, we used the maximum of the Popularity Index to recommend the users.

At the next level of recommendation, we have used the Clustering Method to throw the recommendations to the users. At first, we have a matrix of size (m×n) where we have stored the interest factors of the users according to the genres. Table 1 demonstrate the structure of the matrix.

Table 1
The Matrix used to store the interests of the users

Genre	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8
Java	4	5	1	0	6	8	4	2
Python	5	6	2	1	7	2	3	1
Linux	3	2	2	2	10	6	5	6
Oracle	1	1	1	0	1	3	5	8
ML	0	1	8	9	1	0	1	0
CV	1	1	10	10	1	1	1	8

Now, from this table, we are interested in finding the right recommendation for the users. We used clustering method to find out the right choice. First, we searched the highest interest factor for a particular user. For example, from the Table 1, U_1 has highest interest in Python. So, we recommended the user U_1 with python threads. In the next level, we saw that users U_1, U_2 and U_5 are in the same group because they have almost same Interest Factor in the field of Python. To calculate this we used Mean Deviation (MD). Where the formula for calculating the MD is given below. Here u represents a particular user and U represents the user group. Here if_u represents the interest factor of the user of a particular genre and $\overline{if}$ represents the Interest factors of the user who is logged in a particular genre.

$$md = (if_u - \overline{if}) \; where \; u \, \epsilon \, U$$

Now if *md* is in a range of *2 – 2.5* we declared the particular user *u* in the group of the particular user we are interested with. We recommended the user with the latest threads which are initiated by the users of his or her group. This calculated this measure robustly. Thus, the recommendation process is dynamic this results in a situation that, the user may not fall in a particular

group each time. As the interest of human being changes frequently, the user who was interested in Python at a particular time may not be interested in Python one week later. He or she may be interested in JAVA currently, then he or she should fall in the cluster of the people who are interested in JAVA. This constraint is also served by our system.

Books are necessary part of the students or working people who are interested in technological discussions. Thus, books are also recommended to the users which are relevant to the thread for which the user is searching. For example user U_1 is searching for a thread on Linux and when he or she once found the particular thread he or she is interested with. We are showing the popular books on Linux. There are two buttons, "Helpful" and "Not Helpful", with each and every book to track the user's preference about the books. When the user is clicking on the "Helpful" button we are transferring him or her to a new page with the description of the book, and possible download link of those books which are available in Online Versions. But, when the user clicks on the "Not Helpful" button we are showing the book, which may be the most helpful one for the user. We are doing this with a new algorithm. Using the MD we are creating a group of users who share their interests. And, then we are selecting those books which has been recommended by the users of that group in that particular genre. According to the aforesaid example, we will show the relevant Linux books to the user. The figure given below describes the algorithm for selecting the relevant book.
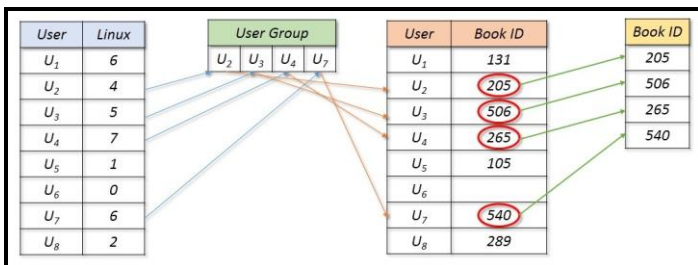

Figure 3: The Process for recommending the books to the users

Now, once we get the Book IDs we are showing the relevant books to the user according to the Book ID one by one when the user is clicking on the "Not Helpful" button.

C. Databases:

MySQL is the most popular Open Source Relational SQL database management system. We have used MySQL for storing and retrieving all the database information both conveniently and efficiently. In a database forum we have used several tables for different operation. Following are the tables:-

1. Table mod_deatil – Used for storing the details of the user. As the data is fetched we store it with a particular user-id. This table have attributes, like-
   i) mod_id (int)
   ii) user_name (varchar).
   iii) Password (varchar)
   iv) email_id (varchar)

2. Table ques_tbl - This table is used for storing queries being asked by user. On submitting the question, the question with particular question id is stored in a table with Priority=0 in the beginning. As many times the Question is being searched, the Priority is increased. Attributes are-
   i) Q_id (int)
   ii) Ques (varchar)
   iii) Priority (int)
   iv) Link (varchar)
   v) Genre (varchar)

3. Table ans_tbl – Mainly used to store the posts submitted for any conversational thread with the date of posting.Attributes are as-
   i) Q_id(int)
   ii) Posts (varchar)
   iii) Date(date)

4. Table rec_new – Chiefly, this table is used for applying Recommendation System to the forum where we have one columns for one user-id with genre and average attribute. The user searching for any thread will increase the corresponding genre along that user's column by value 1.Attributes-
   i) Genre(varchar)
   ii) Average(int)
   iii) User's name as columns(int)

5. Table book_rec – Again, we have another table for recommending books where we have particular column named by user's-id with the book-id for storing value 0 ,if the book is not helpful and 1, if the book is helpful along that user's column corresponding to particular book-id.Attributes are-
   i) Book_id(int)
   ii) Genre(varchar)
   iii) User's name as column(int)

6. Table book_tbl –Used to store the unique book id with different attributes where we have book priority which is increased each time when the book is downloaded.Attributes are as follows:-
   i) B_id(int)
   ii) B_genre(varchar)
   iii) B_priority(int)
   iv) B_link(varchar)
   v) B_name(varchar)
   vi) Img(varchar)
   vii) Description(varchar)

Here, a varchar represents a string and an int represents an integer of length 8 bit.

IV. RESULTS

Here goes some of the web pages of our development, which shows the recommendation and various other features. The Figure 4 is showing the Home Page of our system.
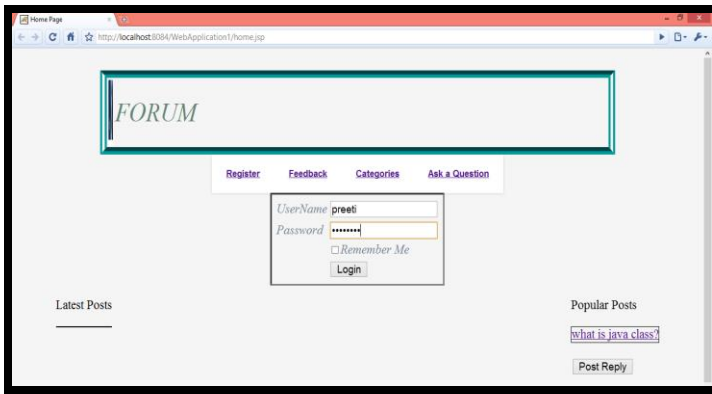
Figure 4: The Home Page displayed to the visitors

The next screenshot displays the User's Profile of our system.
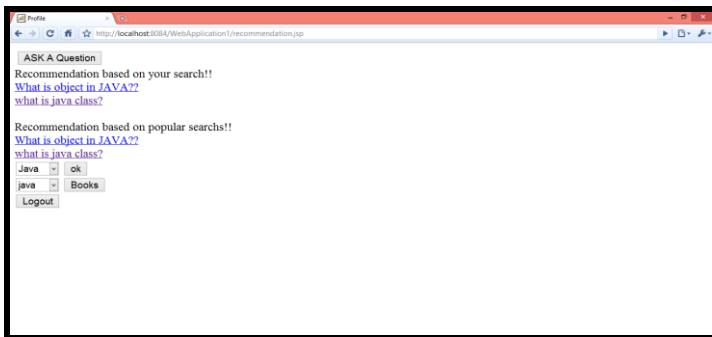

Figure 5: The User Page displayed to the users

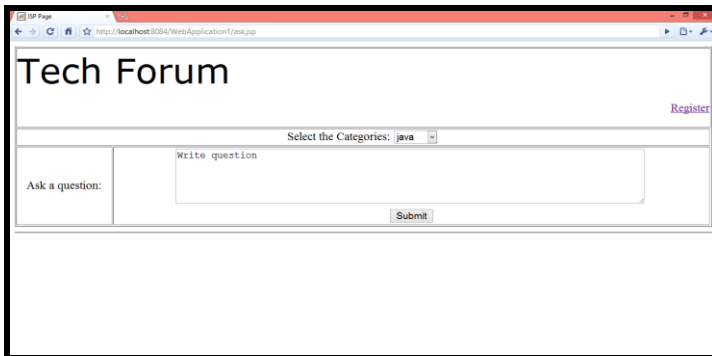The next screenshot displays the page from where an user can ask a question.

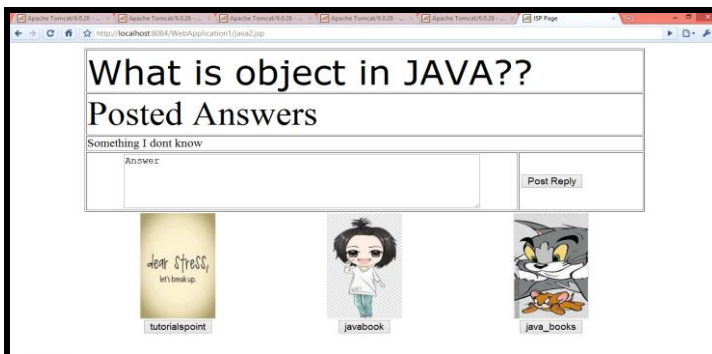*Figure 6: The page for posting new thread by the users*

Figure 7: The page for showing an existing thread and commenting on the thread
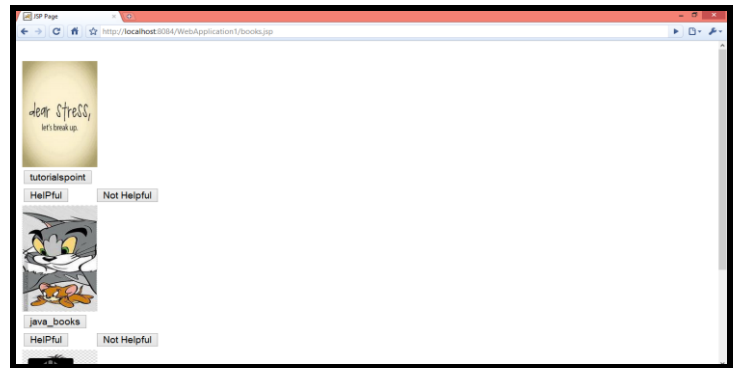

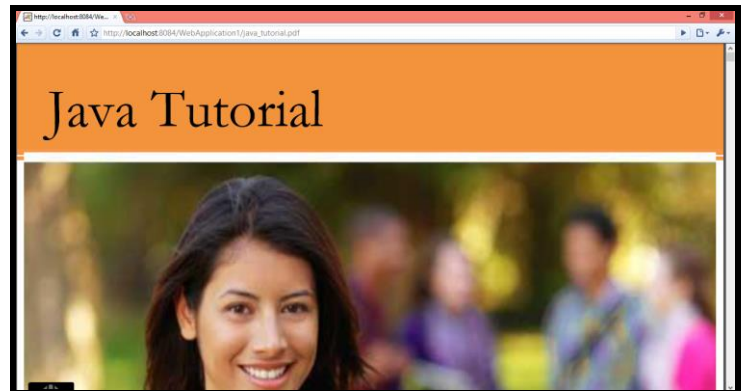*Figure 8: The page displaying the recommended books*

Figure 9: The page where the e-books are being downloaded

The Figures 6, 7, 8, 9 are displaying several screenshots of our system.

V. FUTURE WORK

We will work on this later and we will emphasize on the security of the system. We will use cryptographic algorithms like Hashing, Private Key Encryption intensively to make the system more secure and protect this from serious threats. Moreover, we will work on the Recommender System to make it more efficient.

VI. CONCLUSION

The methodology section describes the implementation of the system in detail and the result section shows that the system is working as per our expectation.