E-COMMERCE RECOMMENDER SYSTEM

Chongtham Cha Chingthangkhomba[1], Sanjay Patel K M[2], Madesh Kumar J S[3]
[1,2,3]Student, CSE Dept, JSSSTU, Mysuru, Karnataka, India

Abstract: ***E-commerce provides an easy way to sell products to a large customer base. However, there is a lot of competition among multiple e-commerce sites. A humongous number of products are available at the tip of the customer's finger. When customers land on an e-commerce site, they expect to find what they are looking for quickly and easily. Finding the right product from all these available products is becoming more and more difficult as customers are not sure about the brands or the actual products they want to purchase. They have a very broad idea about what they want to buy. A lot of products which would have been liked by the right consumers often become invisible because of these huge options. For a seller, making her products visible to the right customer is also a difficult task. It is a huge loss of resources to make every product visible to every single customer. These problems are seldom encountered in physical stores as every customer is attended one at a time. In e-commerce, the consumers active at a time are much beyond the capability of a human to manually handle them. To have a successful business, the right customer needs to find the right product. The takeaway from this is that for e-commerce sites to survive in the market, they need a system to suggest or recommend the right products to the right customers effectively.***

Keywords: Data mining, Classification, E-commerce, E-commerce recommender system, Collaborative filtering, Content based filtering, Collaborative filtering, Hybrid algorithm

I. INTRODUCTION

With the development of E-Commerce, it's harder and harder for consumers to find the product they want, and the recommender systems are applied more and more widely. A recommender system includes user model, the recommended model and recommendation algorithm. The improvement considered in this paper mainly refer to the recommendation algorithm. Limited resource, data valid time and cold start problems are not well considered in existing E-Commerce recommender system. According to the problems descripted above, an algorithm based on limited resource and a solution to cold start problem are proposed.

II. LITERATURE SURVEY

In online shopping as well as in real time shopping there arises an issue where the business entity needs to know the items that are being sold frequently. Not only the items frequently sold, but other items bought with that frequently bought item (Customer buying behaviour). This information is required in order to deal with the imports of items by the business entity depending on the sales record as well as placements of items in the Shopping Malls which increases sales.

A collaborative filtering recommendation algorithm based on user interest change and trust evaluation:
Description:
A collaborative filtering algorithm is one of the technologies used in personalized recommender system. The traditional algorithms focus only on user ratings and do not consider the changes of user interest and the credibility of ratings data, which affected the quality of the system's recommendation. Hence this paper presents an improved algorithm to solve this problem. The main idea is based on the assumption that similar users have similar preferences. By computing user's similarity based on the user ratings to find the neighbours who have the similar interest with the active user. Then the active user's preference for an item can be predicted by combining the neighbour's scores for the same item. Finally, top-N items which the active user will most probably like are provided. But the similarity measurement in collaborative filtering algorithm only pay attention to the similarity score rather than the user interest. The user scores at different time will lead to the different recommended result from the user's current information needs. Hence in this paper, a new collaborative filtering algorithm is based on time-weight and credit evaluation. By integrating the time weight based on non-linear forgotten function and user credit evaluation during the course of the user similarity computation, we can find the target user's neighbour with similar interests more accurately, making final recommended results based on neighbour average ratings can better reflect the change of user interest in time.

Cold-start problem in collaborative recommender systems: efficient methods based on ask-to-rate technique
Description:
To develop a recommender system, the collaborative filtering is the best-known approach which considers the ratings of users. It is able to compute the similarity of users when there are enough ratings expressed by users. Therefore, a major challenge of the collaborative filtering approach can be how to make recommendations for a new user that is called cold-start user problem. To solve this problem, they have been started with a method based on making a profile of a new user by integrating information provided by user. This paper is a review of these proposed methods and how to use the ask-to-rate technique. Consequently, they are categorized into non-adaptive and adaptive methods. Then, each category is analysed and their methods are compared. The objective of a recommender system typically is to recommend items that best fit user's personal preferences. Since the system does not have any data about the new user preferences, it could not provide any personalized recommendation for him/her. It has to acquire some data about the new user. In this paper we have reviewed several

methods for dealing with the new user problem via ask-to-rate technique.

III. EXISTING SYSTEM

Retail marketers are constantly looking for ways to improve the effectiveness of their campaigns. One way to do this is to target customers with the particular offers most likely to attract them back to the store and to spend more time and money on their next visit. In existing e-commerce applications such as "flipkart.com", "myntra.com", "amazon.in", "ebay.com", "snapdeal.com" etc. we have many services which helps customer shopping. Difficult for the consumers to find the products they want in existing recommender systems. Limited resource situation, data valid time and cold start problems have not been well considered in existing Ecommerce recommender system.

IV. PROPOSED SYSTEM

Proposed system is an effective E-Commerce recommender system that can give out effective recommendations for customers which can be approved by customers as far as possible. Customers can get benefits, at the same time, the trading volume can be enhanced. Proposed system finds the solution for the existing recommendation problems such as; limited resource situation, data valid time, cold start problems

V. INPUT DATA

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering data entry operators need to know the allocated space for each field; field sequence and which must match with that in the source document. The format in which the data fields are entered should be given in the input form. Here data entry is online; it makes use of processor that accepts commands and data from the operator through a key board. The input required is analysed by the processor. It is then accepted or rejected. Input stages include the following processes

1. Data Recording
2. Data Transcription
3. Data Conversion
4. Data Verification
5. Data Control
6. Data Transmission
7. Data Correction

One of the aims of the system analyst must be to select data capture method and devices, which reduce the number of stages so as to reduce both the changes of errors and the cost. Input types, can be characterized as:

1. External
2. Internal
3. Operational
4. Computerized
5. Interactive

Input files can exist in document form before being input to the computer. Input design is rather complex since it involves procedures for capturing data as well as inputting it to the computer

VI. DEVELOPMENT

1. Recommendation Process:

On the basis of collaborative filtering principle, the recommendation process of customer's attractions can be divided into three steps;

a. The representation of user (customer) information. The purchasing history of attractions by customer need to be analysed and modelled.

b. The generation of neighbour users (customers). The similarity of customers can be computed according to the buying history data and the collaborative filtering algorithm. A neighbour customer list can be calculated on the basis of known similarities.

C. The generation of attraction recommendations. Top-N attractions will be recommended to the customer according to the buying history of his neighbours.

According to above steps, user's basic information and past purchasing history can be used to calculate the user list of neighbours.

2. Generation of Neighbours

a) Neighbour users generated mainly based on the similarity between each user.
b) Suppose that the set of all customers C = {C1, C2...Cn}, for each customer Ci (i=1, 2... n), the system can calculate the neighbours list including the top N customers which similarity is higher than the given threshold.
c) There are mainly three ways to measure the similarity between customers, including Cosine method, Correlation similarity method and Adjusted Cosine method

3. Generation of Recommendations

Recommendations of attractions are computed by the purchasing times of neighbours. According to the calculation above, we know that the neighbours of customer T1 are T2 and T3, so we can list all the purchasing history of all the attractions so as to summary

TABLE I. **Purchasing History of the customer**

Customers	Attractions				
	A_1	A_2	A_3	A_4	A_5
T_1	1	0	1	1	1
T_2	0	1	1	1	1
T_3	1	0	1	1	0
T_4	0	1	1	0	1
T_5	0	1	0	1	0

$$sim(T_i, T_j) = \frac{|S_i| \cup |S_j|}{|S_i| \cap |S_j|} \quad (2)$$

Based on (2) and Table 1, we can calculate the similarity between T_1 and T_2, T_1 and T_3, T_1 and T_4, T_1 and T_5 as follows

$$sim(T_1, T_2) = \frac{|S_1| \cup |S_2|}{|S_1| \cap |S_2|} = \frac{3}{5} = 0.6 \quad (3)$$

$$sim(T_1, T_3) = \frac{|S_1| \cup |S_3|}{|S_1| \cap |S_3|} = \frac{3}{5} = 0.6 \quad (4)$$

$$sim(T_1, T_4) = \frac{|S_1| \cup |S_4|}{|S_1| \cap |S_4|} = \frac{2}{5} = 0.4 \quad (5)$$

$$sim(T_1, T_5) = \frac{|S_1| \cup |S_5|}{|S_1| \cap |S_5|} = \frac{1}{5} = 0.2 \quad (6)$$

If the value of threshold θ is set to be 0.5, then the neighbors of T_1 are T_2 and T_3.

the most popular ones. As listed in Table 2, we can find that the maximal purchasing times of neighbours are attraction A3 and attraction A4.

TABLE II. **Purchasing history of neighbor customers**

Customers	Attractions				
	A_1	A_2	A_3	A_4	A_5
T_1	1	0	1	1	1
T_2	0	1	1	1	1
T_3	1	0	1	1	0
Total	**2**	1	3	3	2

When new customers enter the system, there is usually insufficient information to produce recommendation for them, because there is no purchasing history of the new customers. The usual solution of the cold start problem is similarity calculation between each user by profile information, such as age, sex, professional, income, etc.

VII. IMPLEMENTATION

This web application is implemented using object-oriented programming language. Object oriented programming is an approach that provides a way of modularizing programs by creating partitioned memory area for both data and functions that can be used as templates for creating copies of such modules on demand.

This project is implemented using three tier architecture. ASP.NET is used in the presentation layer, C# classes are used in the Business logic, Table adopter is used in the data tier and MS SQL server 2005 (database) is used as the backend.

Implementation Steps:

- Presentation Layer: It invokes the Business logic through button click or page load event or SelectedIndexChange event of the dropdownlist.
- Business Logic contains the class members and member functions. An object for Business logic class is created and object will invoke the methods.
- The business logic object will call table Adopter method. Table Adopter will open the database connection. Since SQL server 2005 is used as the backend, to interact with the database SqlDataSource is used.

Connection String:

```
<connectionStrings>
<add
name="ASSOCIATION_RULESConnectionStr
ing" connectionString="Data
Source=COMPUTER21;Initial
Catalog=ERSDatabase;User id=sa;pwd=unlock"
providerName="System.Data.SqlClient"/>
</connectionStrings>
```

b) Data set: This represents an in-memory cache of data.

This project is implemented using Data set “DL.xsd”

The Data Set includes the following table adapters:

- Admin_Details
- Customer_Details
- ContactUs
- FAQs
- Item_Category
- Item_Details
- Item_SubCategory
- Customer_Transactions
- Transaction_Details
- Feedbacks
- Problem_Details
- StrongRules

VIII. CONCLUSION

The whole motive of this project is to build a more efficient recommender system which addresses the limited resource situation and the cold start problem. This project has set a recommender system which identifies the similar users i.e. neighbours and recommends products, which are less in stock, to the right customer. It also generates neighbours based on Area of Interest (AOI) and recommends products in case of users with no history.

Future Enhancement:

- Online money payment is not incorporated in this project, this can also be an enhancement to the current application.
- Click data, browsing data can be analysed to provide a much-improved product recommendation.

REFERENCES

Offline references:

[1] Beginning ASP.NET 1.1 in C# -Mathew MacDonald
[2] ASP.net complete reference - Pearson Publication
[3] Build Your Own ASP.NET Website Using C# &VB.NET
[4] Data Mining- Concepts and techniques – Jiawei Han and Michelin K
[5] Taming the Big Data tidal wave - Bill Franks

Referential URL's:

[1] http://msdn.microsoft.com/asp.net
[2] http://www.asp.net
[3] http://www.w3schools.com
[4] http://www.dotnetjunkies.com
[5] http://www.dotnetspider.com
[6] http://www.webservicex.net
[7] http://www.csharphelp.com
[8] http://www.geekpedia.com
[9] http://www.script20.com

