A Collaborative Approach for Web Personalized Recommendation System

Nirav M. Khetra\(^1\) Shruti B.Yagnik\(^2\)
\(^1\)P.G Student\(^2\)Assistant Professor
\(^1,2\)Department of Computer Engineering
\(^1,2\)LJJET, Ahmedabad

Abstract— Collaborative filtering (CF) is an important and popular technology for recommender systems. However, current CF methods suffer from such problems as data sparsity, recommendation inaccuracy and big-error in predictions. A distinct feature of typicality-based CF is that it finds ‘neighbors’ of users based on user typicality degrees in user groups (instead of the co-rated items of users, or common users of items, as in traditional CF). To the best of our knowledge, there has been no prior work on investigating CF recommendation by combining object typicality. Further, it can obtain more accurate predictions with less number of big-error predictions.

Key words: Recommendation, Typicality, Collaborative Filtering

I. INTRODUCTION

Recommender systems or recommendation systems are a subclass of information filtering system that seeks to predict the ‘rating’ or ‘preference’ that user would give to an item.\(^3\) Recommender systems have become most common in a few years and are applied in lots of applications. The most popular ones are generally Bollywood - Hollywood movies, songs, news, books, research articles, search queries and products. However, there are also recommender systems for comedy, hotels, financial services, insurance services (life insurance), persons (online dating) and twitter.\(^8\)

An information filtering technology, commonly used on e-commerce websites, that uses a collaborative filtering to present information on items and products that are likely to be of interest to the reader. The recommender system will use details of the registered user's profile and opinions and habits of their whole community of users and compare the information to reference characteristics to present the recommendations.\(^9\)

An example of a recommender system is WhatShouldIReadNext.com, a site where users can enter a title of a recent book they have read and enjoyed to see recommended books that they are likely to also enjoy.

One approach to the design of recommender systems that has seen wide use is collaborative filtering. Collaborative filtering techniques are based on gathering and analyzing a huge amount of information on users’ behaviors, activities or preferences and getting predicted on what users will like based on their similarity to other users.

A main advantage of the collaborative filtering approach is that it does not depend on machine analyzable content and therefore it is capable of accurately recommending lots of complex items.

II. ANALYSIS OF CURRENT SYSTEM

Collaborative filtering (CF) is an important and popular technology for recommender systems. However, current CF methods suffer from such problems as data sparsity, recommendation inaccuracy and big-error in predictions.

Data Sparsity. The data sparsity problem is the problem of having too few ratings and hence it is difficult to find out correlations between users and items.\(^1\)

It occurs when the available data are insufficient for identifying similar users or items. It is a major issue that limits the quality of CF recommendations.\(^2\)

Recommendation accuracy. People require recommender systems to predict users’ preferences or ratings as accurately as possible.

Recommender systems are widely used to recommend products to the end users that are most appropriate online book selling websites now-a-days are competing with each other by many means. Recommendation system is one of the stronger tools to increase profit and retaining buyer. The book recommendation system must recommend books that are of buyer’s interest.\(^6\)

III. PROPOSED METHOD

Recommender engines are the most immediately recognizable machine learning technique in use today.

Recommender systems apply knowledge discovery techniques to the problem of making personalized recommendations for any kind of products, information or services during a live interaction.

The User based collaborative filtering technique can be further improved considering the scalability issue. The computation of User similarity is barrier to the system. The method with applying user trust performs worst when data sparsity level is high. The methods here work for
improving the user based similarity measure, by improving the accuracy of the recommendations.

A. Flowchart:

![Proposed Flowchart](image)

**Fig. 3.1: Proposed Flowchart**

B. Proposed Solution:
The steps in the proposed algorithm are as below.

1. Movie-rating dataset would be the input of the algorithm.
2. From the given dataset, a User-item matrix will be generated.
3. MapReduce will be used for parallel computing of the User-based similarity.
4. The output will be processed with Apache mahout applying machine learning approach to generate the recommendations.
5. The output of the recommender system will be evaluated with the base system.

C. Tools and Technology:
Following table shows the tools and technologies used in the research work.

<table>
<thead>
<tr>
<th>Research Approach</th>
<th>User- Based Collaborative Filtering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>Apache Mahout with Hadoop plug-ins configured in Eclipse</td>
</tr>
<tr>
<td>Language</td>
<td>Java</td>
</tr>
<tr>
<td>Dataset</td>
<td>MovieLens Dataset with 1,00,000 ratings MovieLens Dataset with 1,00,00,00 ratings</td>
</tr>
</tbody>
</table>

**Table 3.1: Tools and Technology**

D. Advantages of Algorithm:
- User-based techniques hold allowing CF-based algorithms to scale to large data sets and at the same time produce high-quality recommendations.
- The user will get more precise (accurate) and optimum recommendation.

IV. IMPLEMENTATION

A. Tools for Setting up System Environment:
Recommendation applications have different configuration, composition and deployment requirements. Tools required to implement the Collaborative Approach for Recommendation System are given below.

- Hadoop [10] – Hadoop is an open source software framework that is suited for large datasets (can be of terabytes or petabytes) across large clusters of computers (hundreds or thousands of nodes). Hadoop is an implementation of Google’s simple programming model MapReduce.
- Hadoop Architecture – Figure 4.1 shows the architecture of Hadoop which consists of two main components distributed file system (HDFS) and Execution engine (MapReduce).

![Hadoop Architecture](image)

**Fig. 4.1 Hadoop Architecture**

B. Implementation of Proposed Technique:
The proposed approach is implemented by Java using Eclipse on Hadoop platform under the environment of 2.3 GHz Intel(R) Core i3 processor, GB RAM and Ubuntu 12.0.4.

V. RESULT AND ANALYSIS

A. 1. Result
Following graph shows the comparison of state-of-the-art methods on MAE.
B. Analysis

From the research work some analysis between different techniques have been made. The following table shows the comparison with state-of-the-art methods on MAE which includes SCBPCC, WLR, CBT, SVD++, TyCo and my method nCo.

<table>
<thead>
<tr>
<th>Training Set</th>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML100</td>
<td>SCBPCC</td>
<td>0.874</td>
</tr>
<tr>
<td></td>
<td>WLR</td>
<td>0.915</td>
</tr>
<tr>
<td></td>
<td>SVD++</td>
<td>0.925</td>
</tr>
<tr>
<td></td>
<td>TyCo</td>
<td>0.830</td>
</tr>
<tr>
<td></td>
<td>nCo</td>
<td>0.802</td>
</tr>
<tr>
<td>ML200</td>
<td>SCBPCC</td>
<td>0.871</td>
</tr>
<tr>
<td></td>
<td>WLR</td>
<td>0.941</td>
</tr>
<tr>
<td></td>
<td>SVD++</td>
<td>0.881</td>
</tr>
<tr>
<td></td>
<td>TyCo</td>
<td>0.830</td>
</tr>
<tr>
<td></td>
<td>nCo</td>
<td>0.831</td>
</tr>
<tr>
<td>ML300</td>
<td>SCBPCC</td>
<td>0.870</td>
</tr>
<tr>
<td></td>
<td>WLR</td>
<td>1.018</td>
</tr>
<tr>
<td></td>
<td>SVD++</td>
<td>0.885</td>
</tr>
<tr>
<td></td>
<td>TyCo</td>
<td>0.814</td>
</tr>
<tr>
<td></td>
<td>nCo</td>
<td>0.960</td>
</tr>
</tbody>
</table>

Table 5.1: Comparison With State-Of-The-Art Methods on MAE

VI. CONCLUSION

In this paper, we investigate the collaborative filtering recommendation from a new perspective and present a novel typicality-based collaborative filtering recommendation method. In this method, a user is represented by a user typicality vector which can indicate the user’s preference on each kind of items. A distinct feature of this method is that it selects ‘neighbours’ of users by measuring users’ similarity based on their typicality degrees instead of corated items by users. Such a feature can overcome several limitations of traditional collaborative filtering methods.

There are several possible future extensions to our work. In this method, we do not specify how to cluster resources so as to find out item groups and the corresponding user groups. One possible future work is to try different clustering methods and see how the recommendation results are affected. How to using parallel computing methods (e.g., MapReduce) to handle the large scale applications is also one of the possible future works.

REFERENCE