Scalability Improvement using Map Reduce Algorithm in Recommender Systems

***T. Primya*[1], *G. Kanagaraj*[2]**

[1]Department of CSE, Dr.N.G.P.Institute of Technology, Coimbatore, India

[2]Department of CSE, Kumaraguru College of Technology, Coimbatore, India

E-mail: primya@drngpit.ac.in

Abstract

The proficiently-liked technology for recommender system is collaborative filtering. The current CF methods struggle problems with recommendation inaccuracy, data sparsity and errors in prediction. For curt data retrieval, the implementation of cluster along with map reduce algorithm can glide exactness in prediction and scalability. Clustering of all items into a group is made and then the formation of user group corresponding to each item group is done. By now all the users having swing typically degrees in each of the user group is made. The user typicality matrix to sham the adherent similarities is built. This fanatic typicality matrix based approach will lead to pick a set of neighbours of each user. The prediction of everyday rating of a user concerning the order of an item based upon the ratings of neighbours at adherent upon the item.

Keywords: *Collaborative filtering, clustering, map reduce algorithm, typicality matrix, data sparsity*

INTRODUCTION

Recommendation systems found their application in the ring of e-commerce and internet where items suggest to an action of user almost the basis of their requirement based in the region of their union. A recommendation system is a type of opinion filtering system that built a model from the characteristic of an item according to the rating or prediction, append by a user to an item. Collaborative filtering (CF) is an important and popular technology for recommender systems. There has been plenty of analysis done each in trade and educational level. These strategies are a unit classified into user-based CF and item-based CF [1 –4]. The essential plan of user-based CF approach is to seek out a collection of users WHO have similar favour patterns to a given user. But, these cooperative filtering strategies have facing some issues taking into thought.

1. Data Sparsity.
2. Recommendation Accuracy.

A differential feature of the typicality-based Collaborative Filtering recommendation system is that it selects the "neighbours" of users by measuring users' similarity based on user typicality degrees in user groups, which differentiates it from previously used methods [1]. Our Proposed system provides a new perspective to investigate recommendations by using clusters. This method has the advantages:

1. It works well even with sparse data sets.
2. It can reduce the number of big-error predictions.
3. It improves the accuracy of predictions.

Recommendation System

There has been many works upon recommendation systems and most of these works focus on developing new methods of recommending items to users [1].

Content-based Recommendation Systems

The inspiration of these easy to use hint methods comes from the fact that people had their unreliable evaluations on some items in the late accrual and will have the related evaluations almost new related items in the higher. These easy to use recommendation methods predict the preferences of swift users on the subject of items based upon the preference of supplement linked users or items [1].

Collaborative Filtering Recommendation Systems

This kind of recommendation method predicts the preferences of active users on items based on the preferences of other similar users or items. For the marginal note that collaborative filtering methods do not require neatly-structured item descriptions, they are more often implemented than content-based methods and many collaborative systems are developed in academia and industry [1].

Hybrid Recommendation Systems

Several recommendation systems use a hybrid approach by combining collaborative filtering methods and content-based methods, which helps to avoid some limitations of content-based and collaborative systems. A naive hybrid approach is to take happening collaborative and content based methods separately, and later include their predictions by a combining act, such as a linear inclusion of ratings or a voting plot or supplementary metrics [1].

PROPOSED SYSTEM

This paper proposes an idea of typicality-based CF recommendation proposal named Tyco [1]. The operation of typicality-based CF recommendation is as follows: First, we cluster all items into many item groups. Second, we form a user group corresponding to each item group (i.e., a set of users who like items of a specific item group), with all users having different typicality degrees in each of the user groups. Third, we develop a user-typicality matrix and measure users' similarities based on users' typicality degrees in all user groups it select a set of "neighbours" of each user. Then, we predict the unknown rating of a user on an item, based on the ratings of the "neighbours" of a user on the item [5].

We propose an error-correction technique to recommend similar terms for the question keywords and come back answers of the similar terms. To assist users formulate high-quality queries (keywords), we have a tendency to propose a question expansion-based technique to advocate users of relevant keywords.

Advantages of Proposed System

1. It improves the exactness of predictions when compared with previous recommendation methods.
2. It can shorten the number of big-error predictions.
3. It works competently even with sparse training data sets.
4. Users find relevant patents more easily and improve user search experience.
5. Provide users with gratifications.

PROPOSED ALGORITHM

Clustering Algorithm

1. Cluster models divide the customer base into many segments and treat the task as a classification problem.
2. The algorithms goal is to assign the user to the segment containing the most similar customers.
3. It then uses the ratings of the customer in the segment to generate recommendations.
4. Using a similarity metric a clustering algorithm groups the most similar customers together to form clusters or segments.

Map Reduce Algorithm

1. Map Reduce algorithm is mainly useful to process huge amount of data in parallel, reliable and efficient way in cluster environments.
2. It uses Divide and Conquer technique to process large amount of data.
3. Map Reduce Algorithm uses the following three main steps:
   1) Map Function - Map ().
   2) Shuffle Function – Shuffle ().
   3) Reduce Function – Reduce ().

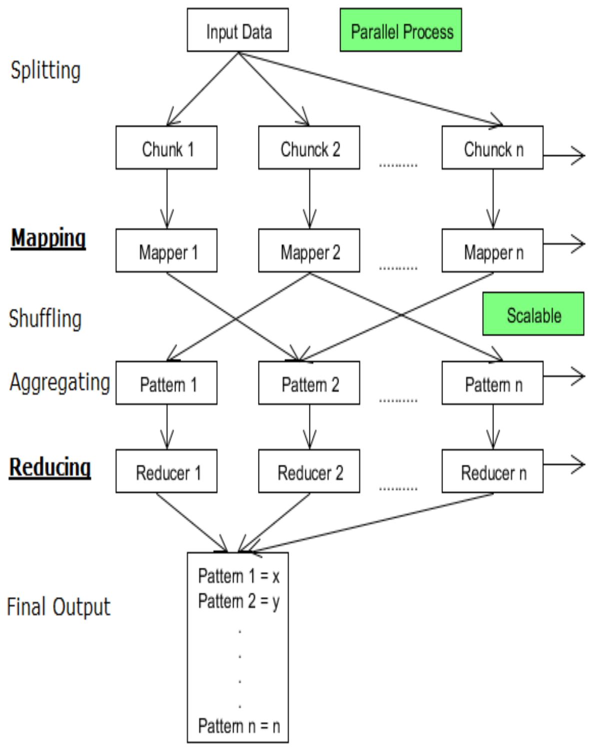


Fig. 1: *Map Reduce.*

Pseudo-Code for Map-Function

1: **Input**
   Moviedataset, Name
   Contents of Movie datasets, Contents
2: **Output**
   A list of (user ratings, 1) pair, one for each movie in the dataset, instances
3: For each movie in movie data set loop
4: emit (user rating, 1)
5: end loop

Pseudo-Code for Reduce-Function

1: **Input**
   A Movie
   List of 1s, Instances
2: **Output**
   The movie and the No.of.Ratings, Result
3: Integer Result=0
4: for each int 'i' in rating loop
5: result+=i
6: end loop
7: emit (rating, result)

SYSTEM DESIGN

The idea of this system is to develop a recommendation engine that can recommend movies to the users with increased accuracy by analyzing the interest of the user and features of the movies. A hybrid recommender system is developed that gets its input from the user in the form of ratings. This ratings list and the profile of the user are the key terms used to predict the interest of the user. The data set considered is a large set of movies which is a big data.

Building Recommendation System

A recommender system is created as GUI to make the user interact with the system in an efficient way. The user can login and logout of the system, can rate the movies, can view the ratings of the movies from the system. The privileges available in this system are:

1) admin
2) user

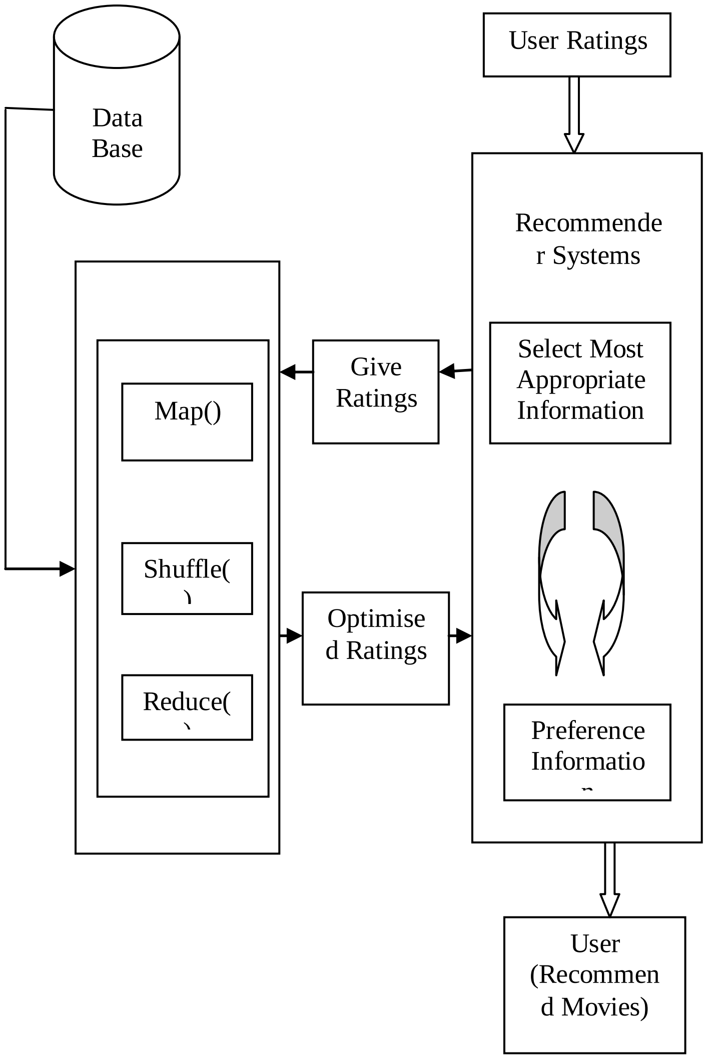


Fig. 2: *Architecture Diagram.*

Admin's Rights
1. Authenticate the new users who sign up.
2. View, edit and upload the movies and corresponding details.

User's Rights
1. To view the ratings of the movie.
2. Rate the movies.

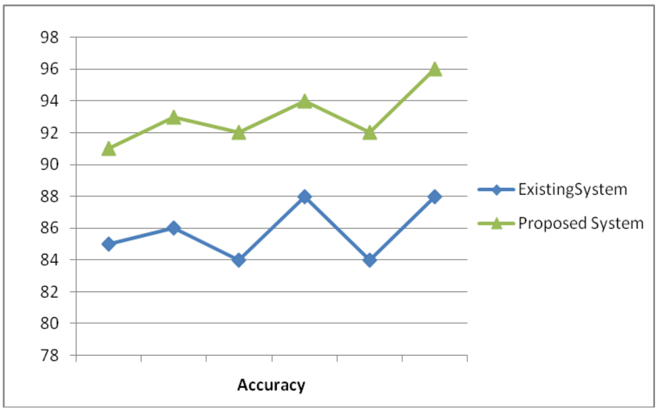
EXPERIMENTS

Data Set

Big Data, i.e., a large set of movies. These movies are collected from the website named Movie Lens. From the MovieLens data set, we obtain 150,000 ratings, assigned by 950 users on 1,700 movies. Each user can give rating to different movies. The ratings can be from 1 (bad) to 5 (excellent) numerical scale.

Expected Experimental Results

Basically, performance of a recommender system can be measured using accuracy. In this work, performance of proposed system is evaluated in terms of calculating accuracy and precision.



The above graph illustrates the expected outcome of enhancing TYCO by improving the accuracy in prediction. It also focuses to reduce big error prediction.

CONCLUSION

In this paper, we examine the movie recommendation system with a new perspective of improving the scalability and reducing the big error predictions by using the Map reduces Algorithm.

FUTURE ENHANCEMENT

One of the possible future works is trying different clustering methods and analyse how the recommendation results are affected. Using parallel computing methods to handle the large scale applications is also one of the possible future works.

REFERENCES

1. Yi Cai, Ho-fung leung, Qing Li, et al. Typicality-based collaborative recommendation. *IEEE Transactions on Knowledge and Data Engineering*. 2014; 26(3).
2. Jyoti Pandey, Manisha R Patil. A survey on recommender system using collaborative filtering techniques; 2015.
3. Z. Hung, H. Chen, D. Zeng. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Trans. Information Systems*. 2004; 22(1): 116–142p.
4. J. Wang, A.P. de Vries, M.J.T. Reinders. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. *Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in*

Information Retrieval (SIGIR '06). 2006; 501–508p.

5. B. Sarwar, G. Karypis, J. Konstan, J. Reidl. Item-based collaborative filtering recommendation algorithms. *Proc.10th Int'l Conf. World Wide Web (WWW '01)*. 2001; 285–295p.