

# A Review Paper on Ranking of Product on Big Data

Smita M. Deshpande

*PG. Student, Department of Computer Engineering  
Sandip Institute of Technology and Research Centre, Nashik, Maharashtra, India*

R. S. Shirsath

*Asst. Professor, Department of Computer Engineering  
Institute of Technology and Research Centre, Nashik, Maharashtra, India*

**Abstract-** Recommendation system has been used by enormous users. It used commonly in recent years, and are used in a variable areas in many popular applications which comprise of movies, songs, bulletin, files, research courses, online shopping, social networking sites, and products recommendation. Gradually, the extent of consumers, products and facts has mature rapidly, the big data scrutiny problem for examination of recommender systems. Here we are evaluating most efficient recommendation system. Conventional recommender systems frequently suffer from deficiency of scalability, efficiency and real time recommendation problems while processing or analysing documents taking place at huge scale. To get rid from these problems, a recommendation system is implemented in Apache Hadoop and Spark including MapReduce paradigm for Bigdata. Proposed Framework can have considerable enhancement in performance associated to traditional tools.

**Keywords –** Bigdata, Collaborative, Hadoop, HBase, MapReduce, Product, Recommendations, Spark, SVD

## I. INTRODUCTION

Product database are repetitively becoming larger, making it progressively difficult for impartial systems to process products data. The large expanse of data is available on the network in the custom of ratings, ranks, appraisals, ideas, complain, explanations, response, and comments about any product which can be any product, event, distinct and services which be capable to use for making correct conclusion [1, 2]. Moreover lots of blog forums are existing on the web where web users can give their judgment, examinations, and comments about the items. The recommendation based on the ranking and summary of appropriate text about the items can be used for decision making. The growth of e-commerce sites and online dealings are improving the requirements of a stout recommendation system. Now a day the many of users buy products from online shopping websites. Approaches of praising innovative things take their boundaries, particularly for information discovery [3, 4]. Recommender systems have convert extremely common in current ages, and are exploited in a various areas some applications which are movies, music, news, books, research objects, search enquiries, social codes, and products. These systems are advantageous substitute to search algorithms as they benefit users to determine stuffs they might not have set up by themselves.

Recommender systems, can be divided into following categories: the content-based filtering, the collaborative filtering and hybrid recommender system [5].

### *1.1. Content-based filtering-*

This filtering method is centred on the item and profile of user's fondness. In the content-based filtering, keywords are used to define the items and a user summary is erected to designate the type of item the user likes. The content based recommender technique is made, using the buyers' profiles based on their pre-experienced item features and the consistency with other items. With the use of the customers' outlines, it can imitate the results of the recommendation or customers' histories of purchases with ease. There are several shortcomings in the content-based filtering. In this filtering, the attributes of recommending goods must be textual and it cannot recommend multimedia files unswervingly and the recommendation is made based only on the customers' pre-experience, it is possible to recommend specific items that are fit to customers' own preference and impossible to recommend unrelated products which are different from what they purchased in the past [6, 7].

### *1.2. Collaborative filtering-*

This filtering method is based on collecting and analysing massive amount of records on user's preferences, activities and behaviour. To make up for limitations of content based filtering, the collaborative filtering is accepted in the recommender system. It predicts based on what user will like built on the resemblance among other users. The collaborative filtering can be said that the most widespread item is recommended for every customer. It is recognized as the most commercially successful recommender technique. This method is based on the assumption that the user who like an item in past will also the like the same or similar type of item in the future. Collaborative filtering practice can be partitioned into the user-based technique using the relations among customers and the item-based technique using the associations between goods [8, 9]. Examples of this filtering are Facebook and LinkedIn recommendations.

### *1.3 Hybrid filtering-*

This filtering is the grouping of content based and collaborative filtering approach. Hybrid approach was presented to handle with a difficulty of traditional recommendation systems. Two foremost problems that are mentioned by researchers in this field are cold-start problem and stability versus plasticity problem. Cold-start problem happens when learning based techniques like collaborative, content-based, and demographic recommendation algorithms are used. Netflix is the example of hybrid recommendation system [10, 11].

The rest of the paper is organized as follows. Proposed embedding and extraction algorithms are explained in section II. Architecture is presented in section III. Concluding remarks are given in section IV.

## II. PROPOSED ALGORITHM

### *A. Proposed System –*

Hadoop has a distributed file structure so it takes the input data from various file systems. Input data is taken from datasets. From this dataset we are taking 70% data for training and 30% data for testing. The data is collected at Hadoop distributed file system (HDFS) as shown in Figure 1. Data comes from various sources such as use metadata, movie metadata and movie ratings. This data is combined in matrix form which goes to the recommendation system over Apache Spark.

Here on Apache spark we are implementing our algorithms. We are using Apache spark over Hadoop system because Apache spark runs 100x faster than Hadoop MapReduce. Then we carry out recommendations using various recommendation algorithms such as collaborative filtering, content-based, hybrid, SVD, trustSVD, etc. We are using linear regression method for providing recommendations to users.

Using this algorithms we are predicting user ratings. To predict user ratings we are factoring our matrix which was in HDFS into vectors and values. Then we associate this factorized data with test user input data, this value which

we get by combining factorized matrix and user test data should approximately match with our original training matrix data. Test user input data comes from the inputs testing data which is stored in HDFS [12].

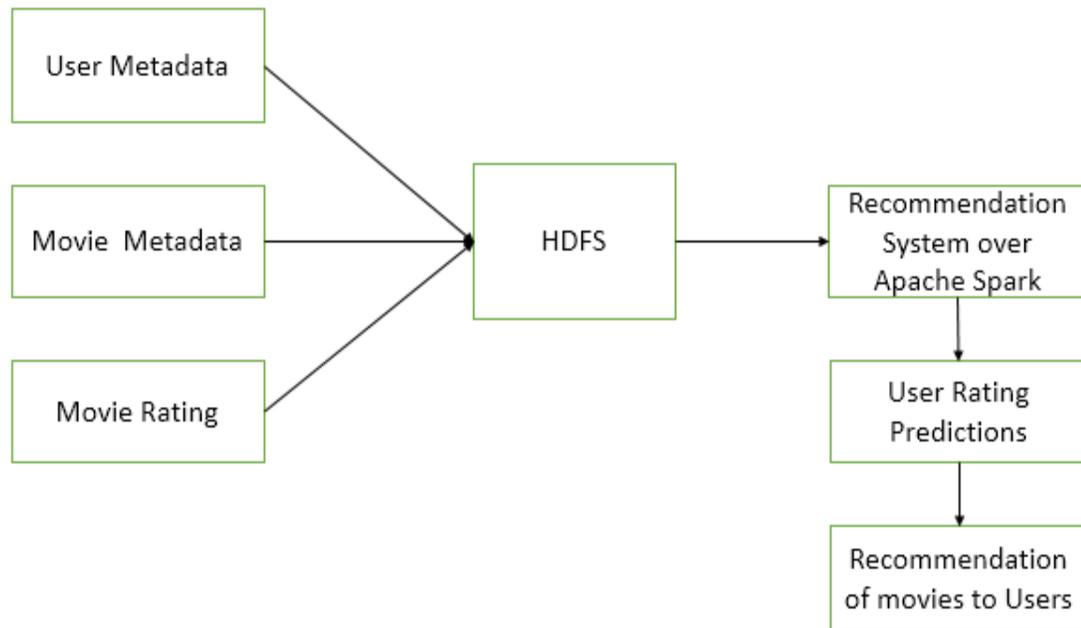


Figure 1. Proposed System Architecture

### III. ARCHITECTURE

Proposed system includes following components they are Hadoop nodes, scattered recommendation engine and Hbase storage shown in Figure 2. Our dataset consists of evaluations. Each rating will have rating associated with each of the evaluations. Every customer in the dataset has a customer Id through which we would be able to know that the associated user has given what rating to the product [13].

Product data is loaded into the Hadoop clusters with MapReduce paradigm. Data is encumbered into clusters so that each node will be able to process data simultaneously. The data is changed in accordance to the customer rather than the product. The user perceptions are compared with other users. So the data is arranged as a customer with list of products he had purchased. Product big data influences MapReduce technique for fast stocking. Data abstraction problems framed as key-value pairs can be capably dispersed with Hadoop as well as HDFS [14, 15].

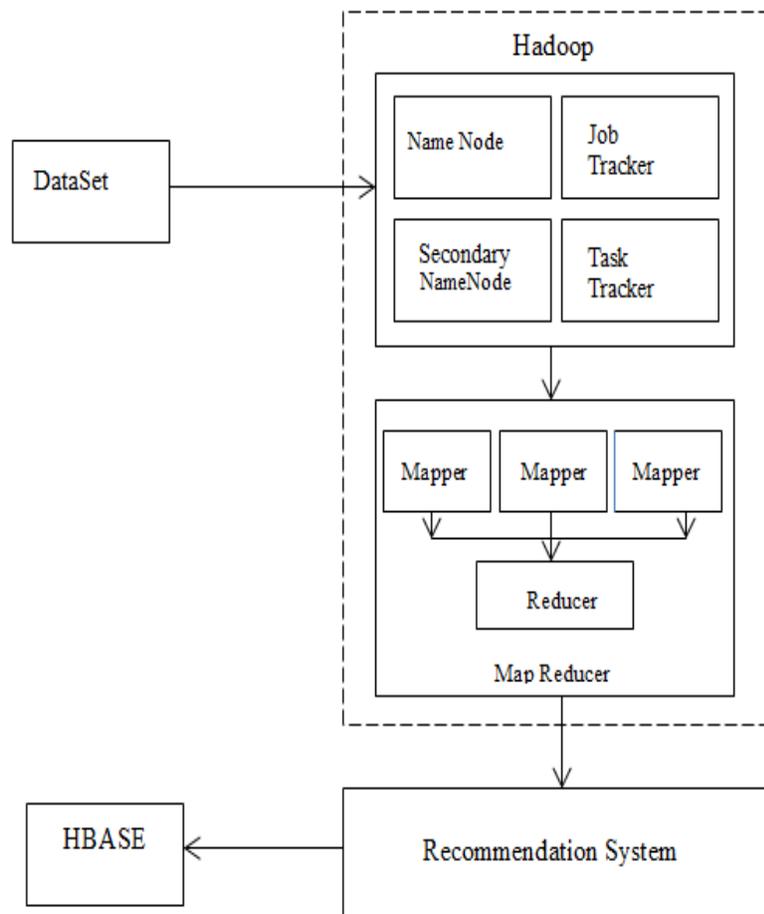


Figure 1 System Flow

#### IV. CONCLUSION

An ascendable product recommendation filtering for Bigdata on Apache Hadoop and Spark can process superior than regular recommendation system based on Hadoop. A boosted HBase gives better presentation. For low expectancy applications HBase is highly preferred as of distributed architecture and leverage the influence of Apache Hadoop. Apache spark is a reckless and universal engine for large scale records processing. It runs programs up to 100x quicker than Hadoop MapReduce in memory, or 10x nearer on disk, so we are realizing our system on Spark to provide scalability and real time commendations. As the size of data escalates the Apache Spark accomplishes well by accumulating additional data nodes for processing.

## V. REFERENCE

- [1] Gejianxin and Liu jiaomin, "Software Test Cases Recommendation System Research based on Collaborative Filtering", IEEE SNPD 2016, May 30-June 1, 2016.
- [2] Bartosz Kupisz and Olgierd Unold, "Collaborative Filtering Recommendation Algorithm based on Hadoop and Spark", IEEE 2015.
- [3] Riyaz P A, Surekha Mariam Varghese, "A Scalable Product Recommendations using Collaborative Filtering in Hadoop for Bigdata", International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST- 2015).
- [4] Dheerajkumar Bokde, Sheetal Girase and Debajyoti Mukhopadhyay, "An Approach to A University Recommendation by Multi-Criteria Collaborative Filtering and Dimensionality Reduction Techniques", IEEE International Symposium on Nanoelectronic and Information Systems, 2015.
- [5] ZhiyangJia,Wei Gao, Yuting Yang, Xu Chen, "User-based Collaborative Filtering for Tourist Attraction Recommendations", IEEE 2015International Conference on Computational Intelligence Communication Technology, 978-1-4799-6023-1/15 2015 IEEE DOI 10.1109/CICT.2015.20.
- [6] Jyoti Gupta, JayantGadge, "Performance Analysis of Recommendation System Based On Collaborative Filtering and Demographics", IEEE International Conference on Communication, Information Computing Technology (ICCICT), 978-1-4799-5521-3, 2015 DOI: 10.1109/ICCICT.2015.7045675.
- [7] Suman Datta, Joydeep Dasy, Prosenjit Gupta and Subhashis Majumder, "SCARS: A Scalable Context-Aware Recommendation System", IEEE 2015.
- [8] FAN Lu, LI Hong, LI Changfeng, "The improvement and implementation of distributed item-based collaborative filtering algorithm on Hadoop", Proceedings of the 34th Chinese Control Conference July 28-30, IEEE 2015.
- [9] Shunmei Meng, Wanchun Dou, Xuyun Zhang, and Jinjun Chen, "KASR: A Keyword Aware Service Recommendation Method on MapReduce for Big Data Applications", IEEE 2014.
- [10] Kunhui Lin, Jingjin Wang, Meihong Wang, "A Hybrid Recommendation Algorithm Based on Hadoop", the 9th International Conference on Computer Science Education (ICCSE 2014) August 22-24, 2014.
- [11] Yingya Zhang, Cheng Yang, ZhixiangNiu, "A Research of Job Recommendation System Based on Collaborative Filtering", Seventh International Symposium on Computational Intelligence and Design, IEEE 2014.
- [12] Poonam Ghuli, Atanu Ghosh and Dr. Rajashree Shettar, "A Collaborative Filtering Recommendation Engine in a Distributed Environment", IEEE 2014.
- [13] Xiao Peng, Shao Liangshan, Li Xiuran, "Improved collaborative filtering algorithm in the research and application of personalized movie recommendations", Fourth International Conference on Intelligent Systems Design and Engineering Applications, IEEE 2013.
- [14] Suyun Wei, Ning Ye, Shuo Zhang , Xia Huang, Jian Zhu, "Item-based Collaborative Filtering Recommendation Algorithm Combining Item Category with Interestingness Measure", International Conference on Computer Science and Service System, IEEE 2012.
- [15] Michael D. Ekstrand, John T. Riedl and Joseph A. Konstan., "Collaborative Filtering Recommender Systems", Foundations and Trends R in Human Computer Interaction Vol.4, No.2 (2010).
- [16] Wu Yueping and ZhengJianguo, "A research of recommendation algorithm based on cloud model", IEEE 2010.