Recommender Systems Built on Hadoop Using Mahout

Monah Evangeline B.

Department of M.E, Alpha College of Engineering, Thirumazhisai, Chennai

Abstract - World Wide Web has been a boon in the fast growing section of social entertainment and commerce. The main process is receiving a help from the recommendation system .In a world of E-Commerce and World Wide Web a recommendation system is the catalyst that induces easiness during the decision making process.The recommendation system helps people in decision making regarding the particular product they are interested in. Collaborative Filtering (CF) is the technique that is being used, which is an algorithm that finds the nearest neighbors among the user with similar decisions. In the vast world of computers, scalability being the problem and in order to manage Big Data the tool called Hadoop is used which enables handling of Big Data. Mahout is a machine learning software that enables Collaborative Filtering (CF) and Clustering which completes the process of a recommendation system. The recommendation system eases decision making, provides handling of Big Data through Hadoop and also has an open source java library, Mahout that enables CF on large sets of data.

Keywords - Recommendation system, Collaborative Filtering, Mahout, Hadoop, Big Data, Recommender algorithms.

I. INTRODUCTION

Recommendation systems has been in use since the year 2000 where it was first used by Pandora Internet Radio a music streaming and automated music recommendation service powered by the Music Genome Project, California, US.

The word “recommendation System” itself, speaks about the meaning of the process that is done in it. In this fast moving world where people deny spending time in commercial and entertainment sectors, the further more hesitation comes when they have to make decisions.

Then came the concept where products were sold Online, which only reduced the time taken to walk around the city from one shop to the other. In order to enhance this, the concept of recommendation system came into the picture.

Recommendation Systems does the process of decision making on behalf of the users which, not only reduces time consumption but on the other hand, suggests the products that are more suitable for the users,

Even though the recommendation system was built to suggest products in online marketing, it has been widely used in almost all sectors right from recommending movies, music, news, books, research articles, search queries, social tags, restaurants, financial services, life insurance, persons (matrimonial), and Twitter followers.

Recommendation System is built based on two ways mainly Collaborative Filtering (CF) and Content-Based Filtering; Collaborative Filtering finds products of similar users and Content-Based Filtering finds similar products with same properties.

Building a Recommendation System depends on the particular field in which the recommendation system will be used. Hence the type of recommendation system also differs according to the type of users and products given. The recommendation system in earlier days where done with help of Data-mining algorithms.

With the growth of technology and the increase in data that is globally affecting the performance of any software, the recommendation system has taken a newer route which makes use of Hadoop and Mahout on Hadoop.

Hadoop is a tool that is being used to enables distributed processing of large sets of data that not only makes the process efficient but enables handing of Big Data, thereby making makes the process scalable with very high fault tolerance.

Mahout, on the other hand also plays a vital role since it is the one that contributes the algorithm to enable Collaborative Filtering (CF). Mahout does two main processes in general, Clustering and Categorizing. Where Clustering is the process of putting together similar items and Categorizing is the process of providing labels to the items that are encountered.

The recommendation system that is presently built for suggesting movies to the users involves CF which finds the interests of similar users according to the ratings they have given for that particular movie.

II. RELATED WORKS

Manos Papagelis, Dimitris Plexousakis[4] suggested that the growth of the sector only led to findings of more accurate and predictable algorithm in order to satisfy the customers completely. Prediction can be defined as the

probability of a user "like" a product which in turn defines recommendation as the "n" list of items with top-n predictions.They also said that there are two main methods or algorithms help in building a recommendation system namely, Collaborative Filtering Algorithm and Content Based Filtering algorithms

The internet usage in the present years has increased enormously leading to an explosion of data. In 2016 the estimated amount of users who use internet in their daily life will be around 3.4 Billion which approximately 45% of the world population. Web search engines like Google, Amazon and Yahoo are the first to face the problem of handling huge set of data says, Fatima EL Jamiy, Abderrahmane Daif, Mohamed Azouazi and Abdelaziz Marzak[7].

A study on improved Collaborative filtering algorithm done by Hee Choon Lee, Seok Jun Lee, Young Jun Chung[8], led to a finding that suggested the preference prediction performance of CMA(Correspondence Mean Algorithm) is way better than NBCFA(Neighborhood Collaborative Filtering Algorithm). Even though these reduce the storage requirement they have scalability issues.

Alexandros Karatzoglou, Alex Smola and Markus Weimer[1] compiled a CF algorithm with hashing using Ït-intensive loss function and Huber loss function. This model is scaled to bigger data-set on large server and also to still data-set over small machines, but the only problem being time complexity.

Dohyung Park ,Joe Neeman ,Jin Zhang ,Sujay Sanghavi ,Inderjit S. Dhillon[6] proposed a paper that talks about the collaborative rank setting: a pool of users each provides a small number of pairwise preferences between d-possible items; from these we will have to predict each users preferences for items they are yet to see and rate. There are two main events depicted in this paper which shows an algorithm based on convex optimization provides good generalization and pairwise comparisons – essentially matching the sample complexity required in the related matrix completion, using actual numerical.

According to Carlos E. Seminario , David C[5]. Wilson in their proposed theory, Apache Mahout is an enabling platform for the research and have faced both of these issues in implying it as part of the work in Collaborative Filtering. This paper gives views on accuracy and coverage evaluation metrics in Apache Mahout, a recent platform tool that provides support for recommender system application development and also the functional changes made to Mahout's collaborative filtering algorithms.

Mahout recommenders support various similarity and neighborhood formation calculations like recommendation prediction algorithms which include user-based, item-based, Slope- One and Singular Value Decomposition (SVD), and also incorporates Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) evaluation methods.Mahout library is readily extensible and provides a wide range of Java classes for customization and has an active community.

III.PROPOSED SYSTEM

The proposed system constitutes an entirely different approach which does not involve any of the Data Mining Technique. Instead it makes use of the Collaborative Filtering algorithm that uses Pearson's Correlation Co-efficient to find the similarity and also uses Cloud environment like Hadoop using Mahout, a java library. The user gives the ratings for a movie or books a product online and this is stored in database which in turn is transformed as data sets.

These data sets are then gathered together and run in a recommendation system, thereby finding the similar products and user's interests.

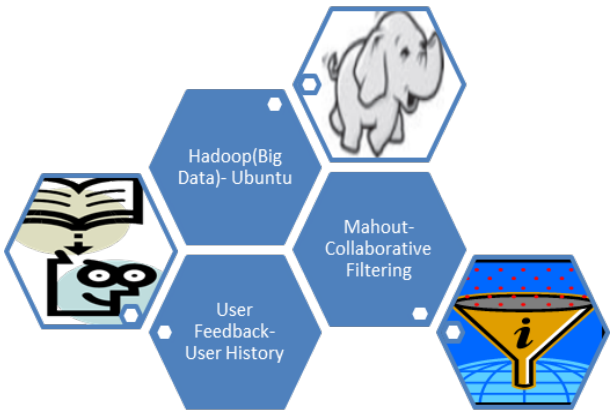

Fig 1: Recommendation System

Hadoop uses HDFS file system, which stores data as flat files. HDFS follows write once read many ideology which doesn't support random read write problem. Thus there will not be any problem regarding the performance of the recommender system providing the most accurate results.

Hadoop can evaluate and generate large sets of data, since it supports HDFS. The CF algorithm involved in the process is supported by Mahout which is a machine learning library. It also provides lower latency access to even small amount of data within a large data set.

Therefore system will provide more accurate result for large data sets also. Mahout overcomes issues with

scalability and Hadoop provides the distributed environment to improve efficiency.

The Hadoop tool composes of three tiers namely, Distributed computation that uses a framework called Mapreduce; Distributed Storage which is nothing but a distributed file system called HDFS providing storage; Server Cloud which runs on commodity hardware.

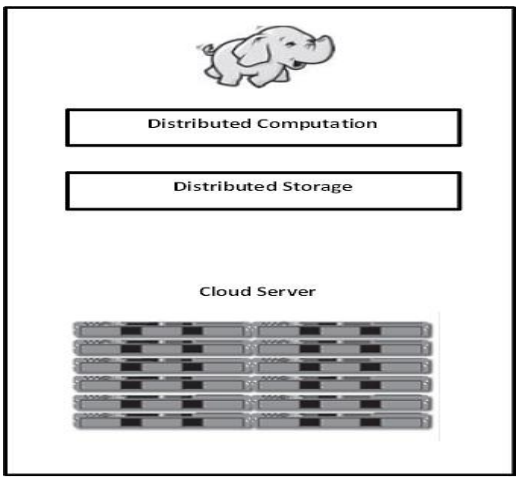

Fig 2: Hadoop Basic Structure

Apache Hadoop provides the distributed environment to a process which only increases the performance of a project. Hadoop runs on HDFS(Hadoop Distributed File System) which is why it handles Big Data exclusively.

Apache Mahout is another machine learning software that helps in clustering and classifying big amount of data. Mahout also supports Collaborative Filtering (CF), thereby making the classification needed to recommend products more easier and extensive. The recommendation system gathers the information from the user history or the feedback ratings given for that particular product and then it is sent to the recommender.

The Collaborative Filtering (CF) algorithm finds the similar users among the user profiles and finds the recommended products This is the simple diagrammatic representation for the proposed system which shows how the flow of control is passed on one section to another

A. The Recommendation Problem:

The first step to build a recommender system, is to find the rough estimation or the prediction on how a user will like an item or on what basis? Thus a utility function is made use; this is based on past behavior, relations to other user, item similarity, context, demography etc;

We assume certain variables in order to form an equation i.e, let C be the users, S be the set of all possible recommendable items. Let U be the utility fuction to measure the usefulness of the item set S to user C, i.e, U:C*S->R, where R is the totally or dered set. Thus for each cϵC chosen items sϵS it maximizes U.

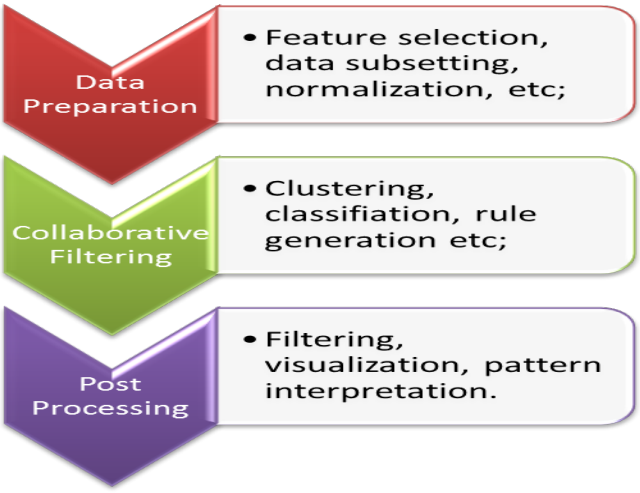

Fig 3: The Recommendation Problem

Thus the process can be explained as a two-step process that is done in offline and online. The learning process and the building of Models and Clusters can be done offline where the decision making process according to the Context given by the user and to recommend we have to be online.

B. Collaborative Filtering Based

The dataset, like a movie dataset is loaded into Hadoop distributed file system (HDFS). Then User-based Collaborative Filtering(CF) using Mahout is enabled.

A rating matrix is given, in which each row represents the users and each column represents the items, corresponding row-column value represents ratings given by the customer to an item. Any absence of rating value indicates that the user has not rated the item yet.

There are many similarity measurement methods used to compute the nearest neighbors among the given set of data. The Pearson's correlation coefficient is used to find similarity between two users. Mahout is used to calculate the similarity.

C. Content Based Filtering

Dataset is loaded into file system, then Mahout performs Item-based CF which is usually done by collecting the Past information of the user, i.e. the ratings given to items that are bought by the customer.

By doing this the similarities between items are brought out and inserted into the item to item matrix. The algorithm selects the items which are most similar to the items rated by the user in their past rating history.

Thus the quick peek of both the collaborative and content based filtering has been explained and here follows what the combination of both the methods will lead to; as the next step, based on top-N recommendations, the target items are selected.

In case of user-based CF, if the nearest neighbors are inadequate, i.e. the interest of the target users does not match that many users then recommending an item for that particular user will be not be accurate.

Item-based CF is based on the past information of the user, so it works well in such cases. The user-based and item-based results that are stored in HDFS are taken and then we combine these results based on threshold value.

D. HDFS

The HDFS framework has a Master node and a Slave node.

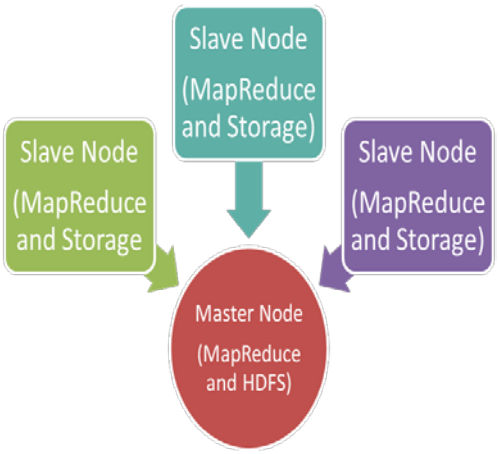

Fig 4: The HDFS framework

The Master node does computation and storage, while the slave node also does computation and storage but in a faster processing level. The master node comprises of MapReduce for computation and the main HDFS for storage. The HDFS does the work of partitioning the storage across the slave nodes and keeping track of the data location. The work of MapReduce is to organize where the computational work has to be scheduled across the slave nodes. The slave node can be given additional nodes for increased storage and processing capabilities.

IV. EXPERIMENT AND FINDINGS

Thus for the experiment, we took a MovieLens dataset of size 1M. The dataset contains 10000054 ratings and 95580 tags applied to 10681 movies by 71567 users. There are three files, movie.dat, rating.dat and tag.dat. For movie recommendation, the most important factor is an immediate list of recommended items.

Since the usage of Hadoop, speedup and efficiency varies as number of nodes varies. To analyze this, the number of movies are obtained which are recommended as threshold changes, Speedup and efficiency also changes according to the number of nodes. While running algorithm in Hadoop framework, speedup varies as numbers of nodes vary.

As the no of nodes increases, speedup increases. if the number of processors is one, the efficiency increases since that processor is fully utilized for the particular program. When the processors increases the efficiency decreases, which means the processors can be utilized for different purposes. This is the one main advantage in the distributed system.

V. CONCLUSION AND FUTURE WORK

The recommendation systems nowadays have their systems build in a way to find more suitable recommendations based on past search queries on-site & off-site, website visiting trend, preference, on past browsing & purchase history and cross checking e-mails.

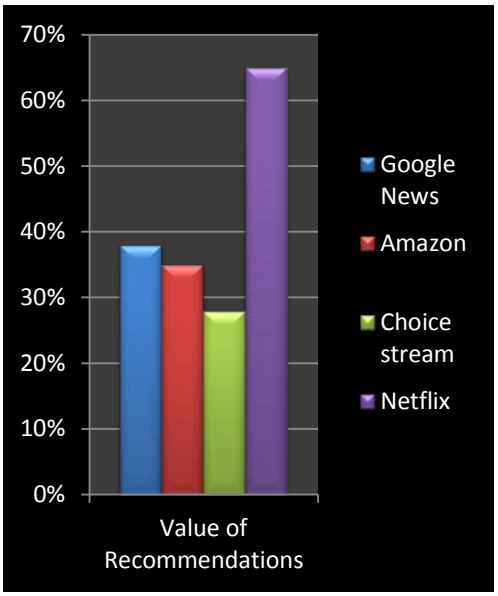

Fig 5: The Value of Recommendations

The collaborative filtering algorithm is used to find the similarity and the present system handles large amount of data which is made possible by using Apache Hadoop.

The value of recommendations has grown in these years and also has proved to be the most sought after by the society. The chart above simply explains the increase in

the usage of recommendation system that has been adapted by the organization famously known all around the world. It goes like this, in Netflix 2/3 of the movies watched, are recommended; Google News generated 38% more clickthrough by recommendations; Amazon had 35% sales from recommendation; Choicestream simply has 28% of music listened and bought only by recommendations.

Thus, the system of recommendation has only been a boon in this growing society and will continue to grow in the years to come. The future enhancement of recommendation system will be the ability in handling enough data to create the correlations that sophisticated engines needed, in which case ultra fine personalization becomes the only focus, to support inhouse marketing efforts and promotions (the latter is largely manual and always evolving), to provide pricing and merchandising decisions. Adding more importance, recommendation engines need to expand to get at least two of the three right for a visitor to convert well, the three being the three P's : Product, Price and Promotion.

REFERENCES

1. Alexandros Karatzoglou, Alex Smola, Markus Weimer, "Collabrative filtering on a budget", Appear in proceedings of the 13th international conference on Artificial Intelligence and Statistics 2010, Italy.

2. Ludlow D, Khan Z (2012) Participatory democracy and the governance of smart cities. In: Proceedings of the 26th Annual AESOP Congress, Ankara,Turkey

3. Khan Z, Kiani SL (2012) A cloud-based architecture for citizen services in smart cities. In: ITAAC Workshop 2012. IEEE Fifth International Conference on Utility and Cloud Computing (UCC), Chicago, IL, USA. pp 315–320. IEEE

4. Manos Papagelis and Dimitris Plexousakis, "Qualitative analysis of user based and item based prediction algorithms for recommendation agents", Science Direct 2005.nb

5. Carlos E. Seminario and David C. Wilson, "Case study evaluation of mahout as a recommender plateform", presented in workshop on recommendation utility evaluation: Beyond RMSE, held in conjunction with ACM in Ireland, 2012.

6. Dohyung Park ,Joe Neeman ,Jin Zhang ,Sujay Sanghavi ,Inderjit S. Dhillon, "Preference Completion: Large-scale Collaborative Ranking from Pairwise Comparisons"

7. Fatima EL Jamiy, Abderrahmane Daif, Mohamed Azouazi and Abdelaziz Marzak, "The potential and challenges of big data - recommendation systems next level application"

8. Hee Choon Lee, Seok Jun Lee, Young Jun Chung, "A study on improved collaborative filtering algorithm for recommendation system", IEEE 2007.

9. J. B. Schafers, J. Konstan and J. Riedi, "Recommendation Systems in e-commerce", 1st ACM conference on Electronic Commerce ACM press, pp. 158-166, 1999.

10. Sarwar B. and Karypis, "Item based collaborative filtering algorithms" in 10th International World Wide Web conference, 2001

11. Allen, R.B., 1990. User models: theory, method and Practice. International Journal of Man–Machine Studies. Balabanovic, M., Sholam, Y., 1997. Combining content-basedand collaborative recommendation. Communications of the ACM40 (3).

12. Breese, J.S., Heckerman, D., Kadie, C., 1998. Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence.

13. Cosley, D., Lawrence, S., Pennock, D. M., 2002. REFEREE: an open framework for practical testing of recommender systems using ResearchIndex.

14. M. Papagelis, D. Plexousakis / Engineering Applications of Artificial Intelligence 18 (2005) 781–789

15. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J., 1999. An algorithmic framework for performing collaborative filtering. Proceedings of the 22nd ACM SIGIR Conference on Research and Development in Information Retrieval.

16. Herlocker, J.L., Konstan, J.A., Riedl, J., 2000. Explaining collaborative filtering recommendations. Proceedings of the ACM Conference on Computer Supported Cooperative Work.

17. Herlocker, J., Konstan, J.A., Terveen, L., Riedl, J., 2004. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) 22 (1).

18. Hofmann, T., 2003. Collaborative filtering via Gaussian probabilistic latent semantic analysis. Proceedings of the 26th Annual International ACM SIGIR

Conference on Research and Development in Information Retrieval.

19. Huang, Z., Chen, H., Zeng, D., 2004. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. ACM Transactions on Information Systems 22 (1).

20. Kalles, D., Papagelis, A., Zaroliagis, C., 2003. Algorithmic aspects of web intelligent systems. In: Zhong, N., Liu J., Yao,Y. (Eds.), Web Intelligence. Springer, Berlin, pp. 323–345.