International Journal of Recent Research in Science, Engineering and Technology

Vol. 1, Issue 1, April 2015

RANK BOOSTING APPROACH ON HADOOP FOR CONSUMER PREFERENCE RECOMMENDER SYSTEM

V.Padmapriya[1], A.Appandairaj[2]

P.G. Scholar, Department of CSE, Ganadipathy Tulsi's Jain Engineering College, Vellore, Tamilnadu, India[1]

Assistant Professor, Department of CSE, Ganadipathy Tulsi's Jain Engineering College, Vellore, Tamilnadu, India[2]

ABSTRACT: In recent years, the growth of number of customers, number of services and online information are increasing rapidly. Recommender systems provide recommendations about the services to the users. In this paper, we propose a recommender system which provides service recommendation based on user preferences. User Preferences are identified by the keyword. Based on the keyword, service recommendations are provided to the user. For this process, we use a user based Collaborative Filtering algorithm and implement it in Hadoop environment. To improve the efficiency and scalability of the process, we propose "combined preferences" using Rank Boosting algorithm. This algorithm processes the similarities with the reviews of existing users based on the preferences. Then it provides ranking to the services and generate output recommendations. Finally, recommender system generates high similarity matching results as the recommendation list to the end users for their combined preferences.

KEYWORDS: Recommender system, keyword, Hadoop, combined preference, rank boosting algorithm.

I. INTRODUCTION

Although several recommender systems are there, user preference recommender system becomes popular because of its personalized recommendation to each user. With the growing amount of data, effectively recommending services that users preferred have become an issue. The amount of information and items got profoundly and immensely large, leading to an information overload. It became a big problem to find what the utilizer is genuinely probing for. For doing a right decision, customers still encounter a very time consuming process and get worthless information by themselves.

Sometimes the contents of internet documents that customers browse have nothing to do with those that they require indeed. In parallel and distributed systems, variety of nodes connected within the network. Both parallel and distributed systems may be outlined as a group of processing elements that communicate and collaborate to attain a typical goal. The node to node communication is done through messages. Green computing also deals with the energy consumption during information transfer between nodes to nodes in the network. The main objective of the green computing is to reduce the cost by reducing the energy consumption during information transfer between nodes to nodes.

Big data brings new opportunities and critical challenges to industry. Big data management stands out as a challenge in IT sectors. Big data tendency also provide heavy influence on service recommender systems. To overcome these problems, we implement it in hadoop. Hadoop is a Big Data management tool to handle large amount of data. In this paper, we design a recommender system which is user friendly and provide more accurate information than the other related works. It gives more personalized recommendations to consumers. Therefore, this system becomes an interesting and successful recommender system.

II. EXISTING SYSTEM

Recommender systems like Keyword-Aware Service Recommendation method provides service recommendations based on user preferences. It presents a customized service recommendation list and recommending the foremost applicable services to the users effectively. Here the keyword shows the preference of the user. For this process here we use a user-based collaborative filtering algorithm. The recommendation process starts with the specification of the initial set of ratings that is either explicitly provided by the users or is implicitly inferred by the system. To enhance the efficiency of this process it is implemented in Hadoop environment. However this fails to point out accurate recommendation when the consumer enters two or three related keywords and requires personalized recommendation list.

III. PROBLEM STATEMENT

Existing system provide personalized recommendation list based on single keyword and concentrates on that single numerical rating. Through this system, only users who can enter single keyword will get accurate recommendations. This fails to give accurate recommendations when the user enters two or three related keywords and requires personalized recommendation list for those combined preferences.

IV. WORKING OF PROPOSED SYSTEM

To overcome the problems in the existing recommendation system, here we have a tendency to propose combined preferences using Rank Boosting algorithm. Recommender system tends to suggest the top services of the returned result list to the user. The services in higher position, especially the first position, should be more satisfying than the services in lower position of the returned result list. To evaluate the quality of Top-K service recommendation list, MAP and DCG are used as performance evaluation metrics. And the higher MAP or DCG presents the higher quality of the predicted service recommendation list. Evaluating a service through multiple criteria and taking into account of user feedback will facilitate to make more effective recommendations for the users.

To improve the efficiency and scalability of the process, Rank Boosting algorithm gets the input as combined preferences, based on the preferences it process the similarities with the reviews of the existing users then it provides the ranking to the services. The weighting strategy of Boost is to increase weights of misclassified samples and decrease weights of correctly classified samples until the weighted sample distributions between misclassified samples and correctly classified samples are even on each round. Based on the ranking provided to the services we generate the output recommendations. Finally it generates high similarity matching results as the recommendation list to the end users for their combined preferences.

A. Loading and preprocessing of data

In this section , we tend to describe the process of collected information. We initially load the data. Once loading, we tend to analyze the data. After analyzing process, we view the information present in the dataset. After that we begin the preprocessing step. In the preprocessing step, we remove the null value, missing tuples etc. Here we are going to process three set of data. First one is user information. It consists of user id, user age, profession, gender, zipcode etc. Second one is ratings information. The ratings data consists of user id, item id, ratings and timestamp. Third one is movie dataset. It consists of movie id, name, release date, imdb url category of the movie etc. We first load all the data into the hadoop distributed file system and preprocessing all the data.

B. Analysis of user reviews

After preprocessing , we view the cleaned or processed data. At the same time we analyze the user reviews. The user review contains the information about the place or hotels or transportation etc. Using the review we further continue our process. In this process we are going to analyze the user reviews i.e. the users who all are watch the movies already and provide the ratings to that movies. In this processing we compare the relevant results to the ratings dataset and retreive all the ratings of the relevant results movie.

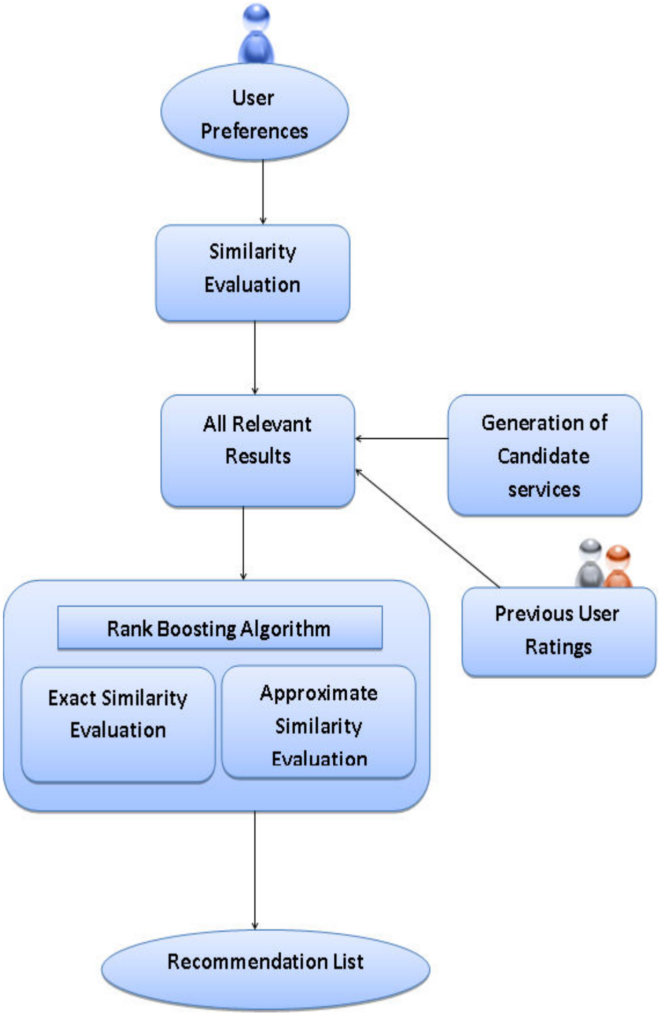


Fig 2.1: Framework of our approach

C. Mapper and reducer process

In this module we first collect the user preferences in the form of query model. We implement the query model to get the user request, here it is user preference. The user preference is processed using the map reduce mechanism. The process is performed by splitting the preferences i.e.it is done using mapper process. After the processing, the results are aggregated**.** In this process separation of ratings takes place. Based on the ratings the movies also categorized. The categorization take place based on the user preferences. The review processing and relevant results are analyzed based on the user preferences.

D. Prediction of recommendation list

After map reduce process execution, we aggregate the result to generate the recommendation list. The recommendation list is generated using user collaborative filtering algorithm. This algorithm generates the output, recommendation list. A keyword-aware service recommendation method, named in this paper, which is based on a rank boosting algorithm. Keywords extracted from reviews of previous users are used to indicate their preferences. We select the ratings from the frame, from the ratings data we provide the results. The input is given in the form of combined preferences, and then using the rank boosting algorithm we generate the recommendation list.

E. Evaluation process

The evaluation process takes place to predict the accuracy of the recommendation list. The evaluation is in the form of graphs. To evaluate the performance of KASR in accuracy, we compare KASR with other two well-known recommendation methods: Rank Boost algorithm using Pearson Correlation Coefficient (PCC) and item-based algorithm using PCC, which are called as UPCC and IPCC respectively.

1) Algorithm: Hadoop Rank Boosting Algorithm

Input : User Preferences, previous user reviews.

Output : Recommendation List

```
Load the user information into HDFS
Load the item information into HDFS
Load the rating information into HDFS
For each categories C1,C2…….Cn
        Categorize the movies with the category.
        Load into HDFS
Get the user preference→combined user preferences   CUP
For each item category
    Check
      if  CUP == item Category
        Retrieve all the relevant item results
        Count the relevant results
          Load results into HDFS
    End if
End loop
For each relevant result item id
    Check
      if item id == rating.item id
      Similarity →sim(prev,curr);
      Extract the relevant result and load into HDFS
      End if
End loop
Ranking the rating results based on the highest rating.
Select the K-Value which is a integer.
Generate the recommendation list based on the K-value.
```

2) Similarity Calculation

Similarity is calculated using

i) Jaccard similarity
ii) Cosine similarity

Jaccard Similarity

Jaccard Similarity = a / (a+b+c);

Where,

a→Number of keywords present in both previous and active keyword list

b→Number of keywords present in previous user keyword list.

c→Number of keywords present in active user keyword list

Cosine Similarity

Cosine Similarity = |a|intersection|b| / (|a||b|)

Where,

|a|intersection|b|→number of keywords common in both set.

Ial→number of keywords in a set.

Ibl→number of keywords in b set.

V. RESULTS & EXPERIMENTS

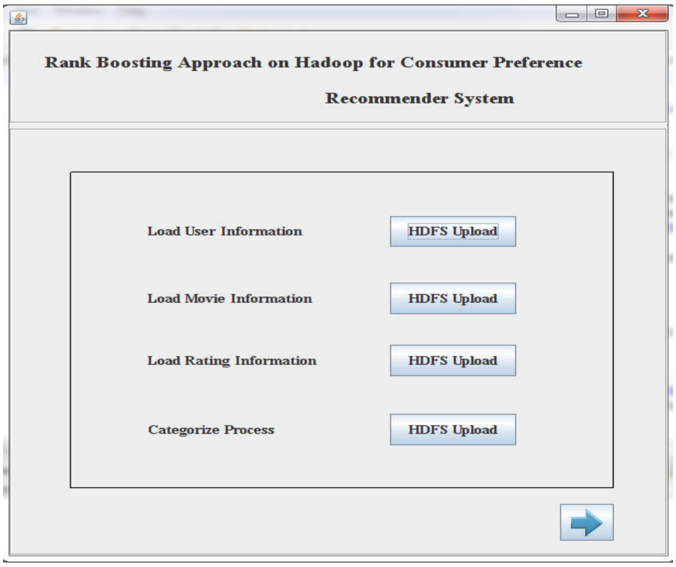
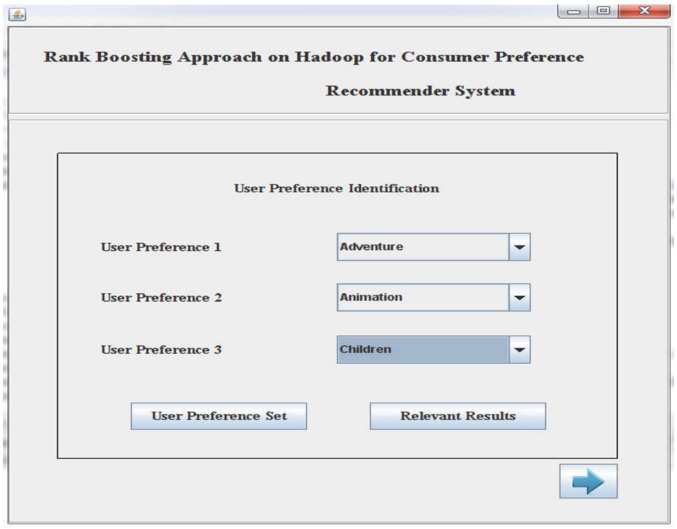


Fig 5.1 Loading data in HDFS

Fig 5.1 represents the process of loading collected data in Hadoop Distributed File System. Here, three datasets are uploaded into Hadoop Distributed File System. They are User information, Movie information and Rating information. Finally Categorize process takes place to sort out preferences of each user.


Rank Boosting Approach on Hadoop for Consumer Preference
Recommender System
User Preference Identification
User Preference Set
Relevant Results

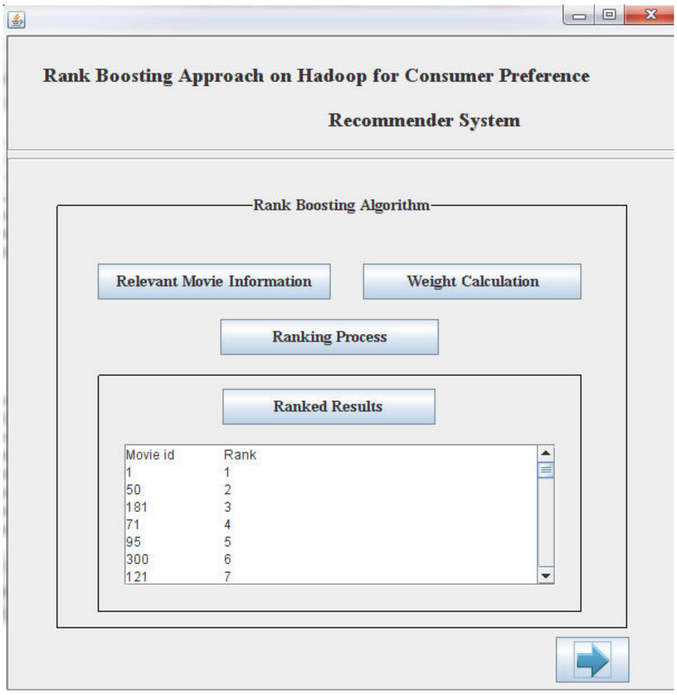
User Preference 1 Adventure

User Preference 2 Animation

User Preference 3 Children

Fig 5.2 Obtain User Preferences

Fig 5.2 represents the process of obtaining user preferences. This process is done using map reduce mechanism. Here the combined preferences of a user are collected as User Preference Set. Then the relevant results are recognized by mapping the preferences with the list of movies. Finally, reducing process retrieves the number of relevant results.



```
Movie id    Rank
1           1
50          2
181         3
71          4
95          5
300         6
121         7
```

Fig 5.3 Ranked Recommendation List

Fig 5.3 represents the generated ranked results of our recommender system. Ranking process takes place based on Similarity calculation.

VI. CONCLUSION

In this paper, our work described how explicit ratings can be utilized in order to implicitly obtain user's preference to specific categories. First, it aims at presenting a personalized service recommendation list and recommending the most appropriate service(s) to the users. Second, to improve the scalability and efficiency in "Big Data" environment, we have implemented it on a MapReduce framework in Hadoop platform. Finally, rank boosting algorithm is used for combined preferences. Experimental results demonstrate that combined preferences based on rank boosting algorithm gives the better results than other recommender system. It also significantly improves the accuracy and scalability of service recommender systems over existing approaches.

ACKNOWLEDGEMENT

First and foremost, I thank the Lord Almighty who paved the path for my walk which lifted me to pluck the fruits of success and who is the torch for all my endeavors and engagements. I record my deep sense of indebtedness and whole heated gratitude to my friends, for their active involvement, encouragement, caretaking and valuable suggestions in all the way and also shaping project work and include report. Home is the backbone of every success and our humble salutations to our beloved parents T.Vadivelu and V.Lakshmi and my lovable brother V.Thanigaivel, who inspired, motivated and supported us throughout the course of the project. I also extend my sincere thank to my guide A.APPANDAIRAJ and the Faculty Members of CSE Department, Friends who have render their valuable help in completing this project successful.

REFERENCES:

[1] J. Manyika, M. Chui, B. Brown, et al, "Big Data: The next frontier for innovation, competition, and productivity," 2011.
[2] C. Lynch, "Big Data: How do your data grow?" Nature, Vol. 455, No. 7209, p. 28-29, 2008.
[3] F. Chang, J. Dean, S. Ghemawat, and W. C. Hsieh, "Bigtable: A distributed storage system for structured data," ACM Transactions on Computer Systems, Vol. 26, No. 2 (4) , 2008.
[4] W. Dou, X. Zhang, J. Liu, J. Chen, "HireSome-II: Towards Privacy-Aware Cross-Cloud Service Composition for Big Data Applications," IEEE Transactions on Parallel and Distributed Systems, 2013.
[5] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, Vol. 7, No.1, pp. 76-80, 2003.
[6] M. Bjelica, "Towards TV Recommender System Experiments with User Modeling," IEEE Transactions on Consumer Electronics, Vol. 56, No.3, pp. 1763-1769, 2010.
[7] M. Alduan, F. Alvarez, J. Menendez, and O. Baez, "Recommender System for Sport Videos Based on User Audiovisual Consumption," IEEE Transactions on Multimedia, Vol. 14, No.6, pp. 1546-1557, 2013.
[8] Y. Chen, A. Cheng and W. Hsu, "Travel Recommendation by Mining People Attributes and Travel Group Types From Community-Contributed Photos," IEEE Transactions on Multimedia, Vol. 25, No.6, pp. 1283-1295, 2012.
[9] Z. Zheng, X Wu, Y Zhang, M Lyu, and J Wang, "QoS Ranking Prediction for Cloud Services," IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 6, pp. 1213-1222, 2013.
[10] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and Evaluating Choices in a Virtual Community of Use," In CHI '95 Proceedings of the SIGCHI Conference on Human Factors in Computing System, pp. 194-201, 1995.

BIOGRAPHY

AUTHOR 1
V.PADMAPRIYA
M.E. (CSE)
GTEC, VELLORE,
TAMILNADU,
INDIA

AUTHOR 2
A.APPANDAIRAJ,
ASSISTANT PROFESSOR,
GTEC,VELLORE,
TAMILNADU,
INDIA.