International Journal of Computer Science Trends and Technology (IJCST) – Volume 2 Issue 4, Jul-Aug 2014

RESEARCH ARTICLE OPEN ACCESS

An Efficient Recommender System using Hierarchical Clustering Algorithm

Prabhat Kumar[1], Sherry Chalotra[2]
Research Scholar[1&2],
Department of Computer Science
Guru Nanak Dev Engineering College, Ludhiana
Punjab-India

ABSTRACT
The massive growth of information these days has created the need for information filtering techniques that help users filter out extraneous content to identify the right information they need to make important decisions. The right information they need to make important decisions. Recommender systems are one approach to this problem, based on presenting potential items of interest to a user rather than requiring the user to go looking for them. Recommender system is a subclass of information retrieval system and information filtering system that seek to predict the 'rating' or 'preference' that user would give to an item. The concept of recommender system grows out of the idea of the information reuse and persistent preferences. Recommender systems have recently gained much attention as a new business intelligence tool for e-commerce business. Applying a recommender system for an online retailer store helps to enhance the quality of service for customers and increase the sale of products and services. In order to recommend items for particular requests the system has to perform large searching, sorting, and filtering and huge matrix operations. This will be a very time consuming operation even for smaller searching operations. Therefore there will be a need of an efficient framework to predict or recommend an item within time bounds. Almost all the recommenders proposed earlier uses continuous algorithms but the nature of the items is discrete and for computer systems the performance of discrete algorithms is much better as compared to continuous algorithms. In this paper a User-User based Collaborative Recommender system has been proposed that makes use of discrete cluster algorithms to enhance the recommendations and improve running time.

Keywords: - Recommender System, Collaborative Filtering, Hierarchical Clustering, Jaccard Index.

I. INTRODUCTION

Servers are central to Businesses. In this information age every Business house owns or lease a server for their business logic. With the increase in Business and business logic the load on a server are increasing with enormous rate. Apart from that each server has to perform various other tasks as responding to client, managing enterprise level calculations [6], handling large databases, security and authentication.

But since the domain of these tasks and there algorithms are well known and most of them are standardized thus the time complexity can be improved. This helps the system engineers to cope up with the load issues with the correct optimization and algorithms.

But it has been observed that the recommender systems can increase the sales of a business house by 8-10%. Hence the race to incorporate the recommender system has begun. Now a day's all the business houses are incorporating Recommender Systems to their servers for competitive advantage. But there exists no known algorithm for prediction or recommendation. Instead we use statistical data and then try to come to a conclusion based on certain hypothesis. For this the server has to collect, clean, load, store, parse and perform long computation to enormous set of data.

The calculation includes various very large matrix operations and there is no known efficient algorithm for matrix operations. Hence the computation takes huge amount of time and space. To perform a matrix operation on data that currently amazon has is sufficient to overpower all the computers in the world working together. This is true even if we make computers out of every atom in this world.

Hence there is a craving need of some efficient algorithms. There are various algorithm proposed in the literature most of them are trying to improve the algorithm using K-means algorithm because K-means algorithm [2] is easier to understand and it is very simple to implement. The simpler the algorithm is the better it will perform. It has been observed in many cases such as image compression technique JPEG 2000. But developing a recommender system is somewhat different from other software engineering tasks. K-means is continuous algorithm and works better if data set is continuous in nature. But this is certainly not in case of recommendation because the nature of items is discrete. And researcher tends to use Euclidean Distance [9] which is computationally very expensive since it requires both power and square root functions. And both of these functions require more than 200

cycles to complete. The situation gets worse when they have to perform these functions over a large dataset. And this very large calculation is performed only for one request for recommendation. But large business house has to millions of these requests per day.

The Time complexity of K-means with N-Neighbors is $2^{\Omega(n)}$ where $\Omega(n)$ is distance function. The best known algorithm for that implements K-means is Lloyd's algorithm, Lloyd's algorithm is an Heuristic algorithm and bounded by $O(n^{34}k^{34}d^{8}\log^{4}(n)/\sigma^{6})$. But even this is very large. Hence efforts are being made to develop a new recommender system which is more efficient than the previous recommenders.

This work focuses on developing a user-user collaborative filtering based recommender system [7] whose running time is better than the recommender based on K-means algorithm.

II. STRUCTURE OF PROPOSED SYSTEM

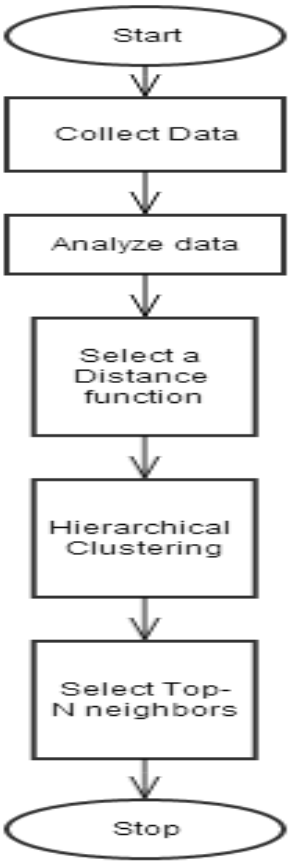

Figure 1 Structure of Proposed System

A. Collecting Data

Collecting or gathering relevant data is an important part of any experimental setup. For this experiment large dataset is required so that evaluation between two different approaches to a recommender system can be achieved with least error.

For this experiment the two chosen datasets are taken from grouplens.org. Among the two datasets the first one contains 100,000 ratings collected from 1000 users on 1700 different movies. And another one contains 1 million ratings collected from 6000 users on 4000 different movies.

B. Cleaning Data

Each dataset consists of various files. The first one contains 5 .dat files along with various other test case files. The other one contains 3 .dat files.

```
1|Toy Story (1995)|01-Jan-1995||http://us.
imdb.com/M/title-exact?Toy%20Story%20(1995)
|0|0|0|1|1|1|0|0|0|0|0|0|0|0|0|0|0
2|GoldenEye (1995)|01-Jan-1995||http://us.
imdb.com/M/title-exact?GoldenEye%20(1995)
|0|1|1|0|0|0|0|0|0|0|0|0|0|0|1|0|0
```

Figure 2 Unclean File

As one can observe the file are barely readable and even harder to input the file into the recommender system. Hence these files need to be translated in a format which is acceptable to the recommender system. There are various ways into which file can be translated some of them are:

- Comma Separated Value (CSV file)
- Tab delimited File
- As a Table
- As a graph

For this purpose Comma Separated Value file is chosen. After Cleaning and merging the file appears.

user_id	movie_title	rating	genre	release_date	age	gender	occupation
1	101 Dalmatians (1996)	2	multiple	27-11-1996	24	M	technician
1	12 Angry Men (1957)	5	Drama	01-01-1957	24	M	technician
1	20,000 Leagues Under the Sea (1954)	3	multiple	01-01-1954	24	M	technician
1	2001: A Space Odyssey (1968)	4	multiple	01-01-1968	24	M	technician

Figure 3 CSV file opened in spreadsheet

C. Analyzing the Data

Analyzing the data is an important part of any recommender system. A true recommender system cannot be built without understanding shape and structure of the data. There are various ways to represent and analyze the layout of the data but the best way to analyze the organization of the data is through a graphical or pictorial representation. The graphical representation for analysis of the data can be achieved by plotting them in N-Dimensional space. N-Dimensions depict

more information than N-1 or lower dimensions. But since human tends to understand 1-D and 2-D world objects better so here 2-D plotting are shown in the figure below

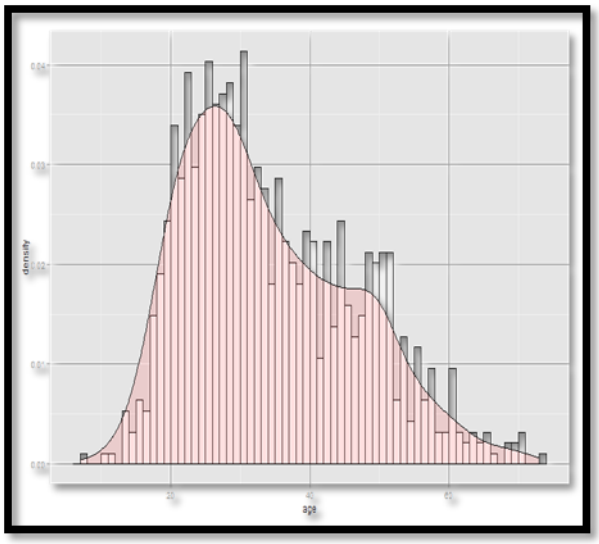

Figure ***Error! No text of specified style in document.****V- Analyzing Age Density*

Figure shows the age group of the users who tends to watch the movie. From the plot it is clear that age group between 20-30 years tends to watch more movies that any other age group.

D. Selecting a Distance Function

In the context of this paper Distance function is a function that depicts how similar or dissimilar two objects under consideration are. There are various different Distance functions. For this system Jaccard Similarity Coefficient [3] is used. Hence it is described in details below.

Table ***Error! No text of specified style in document.****- User-Item using binary variable*

User/ Items	Item 1	Item 2	Item 3	Item 4
User 1	1	0	1	1
User 2	0	1	1	1
User 3	1	1	0	0
User 4	1	1	0	0

Table ***Error! No text of specified style in document.****- Jaccard Similarity Coefficient*

Distance Function	User 1	User 2	User 3	User 4
User 1	X	$^1/_2$	$^1/_4$	$^1/_4$
User 2	X	X	$^1/_4$	$^1/_4$
User 3	X	X	X	1
User 4	X	X	X	x

According to Jaccard Similarity Coefficient,

$$S_{ij} = \frac{\sum_{k=1}^{n} A_{ijk}}{\sum_{k=1}^{n} B_{ijk}}$$

Where,
S = Similarity Function
i = ith row
j = j^{th} row
k = cell under consideration
A = Common Cell

Table ***Error! No text of specified style in document.****I- N-ary variable table*

User / Item	Item 1	Item 2	Item 3
User 1	1	2	2
User 2	1	4	2
User 3	1	2	2

Table ***Error! No text of specified style in document.****- Jaccard similarity function for N-ary Variables*

User / Item	User 1	User 2	User 3
User 1	X	2	0
User 2	X	X	2
User 3	X	X	x

E. Perform Clustering

The above created distance matrix [12] can be used to generate clusters.

The algorithm for clustering [11] for dataset:

1. Start with a point in the cluster having level $L(0) = 0$ and sequence number r = 0.
2. Find pairs that is closest to each other based upon similarity function, say pair x, y, according to $d_{x,y} = \min(d_{i,j})$ where the minimum is over all pairs of clusters in the current clustering.
3. Increment the sequence number: $r = r + 1$. Merge clusters x and y into a cluster to form another clustering r. Set the level of this clustering to $L(r) = d_{x,y}$
4. Update the distance matrix, D, by reorganizing rows and columns based upon newly formed clusters. The distance between the new cluster, denoted (x,y) and old cluster (p) is defined as $d_{[p,(x,y)]} = \mathbf{min}(d_{p,x}, d_{p,y})$.
5. If all the points are in one cluster then goto step 6 else repeat from step 2
6. Stop

In the proposed system N=4 neighbors are selected because it gave good performance with acceptable accuracy.

The illustration of the algorithm using Jaccard Index similarity function is given below. And create 2 groups of clusters.

Initially, there are 4 clusters
Cluster 1 = {User 1}
Cluster 2 = {User 2}
Cluster 3 = {User 3}
Cluster 4 = {User 4}

Table ***Error! No text of specified style in document.****: User Item Matrix*

User/Item	Item 1	Item 2	Item 3	Item 4
User 1	1	0	0	1
User 2	0	0	0	1
User 3	1	1	0	1
User 4	1	0	0	1

Table VI: Cluster group of 3

User – User Similarity	User 1	User 2	User 3	User 4
User 1	X	½	2/3	1
User 2	X	X	1/3	1/2
User 3	X	X	X	2/3
User 4	X	X	X	X

Since, User 4 has the highest similarity with user 1 we can group them together. Hence now clusters appears

Cluster 1 = {1,4}
Cluster 2 = {2}
Cluster 3 = {3}

Table VII: Clustering using minimum distance

User/User	{User1,User4}	User 2	User 3
{User1,User4}	X	½	2/3
User 2	X	X	1/3
User 3	X	X	X

In this process the distance between two cells are computed. In case of choosing between the similar distances one can select the group with fewer members.

It is apparent from the above table that User 3 is at minimum distance with the cell of cluster 1

Hence now two group of cluster is created.
Cluster 1 = {User 1, User 4, User 3}
Cluster 2 = {User 2}

F. Selecting Neighbors

Selection of Neighbors is necessary because in real life scene the size of table is enormous and computing all the distance function between every pair of object is not feasible

1) Top N neighbors [8] – Selecting Top N-Rows
2) Pick nearest neighbors from the same group as formed by Hierarchal cluster [13].
3) Randomly – Picking the objects from the matrix randomly.

Taking random distributed samples is the essence of any inference but since all the correlation among the data is close to zero. Hence for the proposed model Top N neighbors is chosen.

III. EXPERIMENT RESULT AND COMPARISON

The proposed system is modeled using apache mahout project and R programming language.

Both K-means and Hierarchical clusters give the output in scatter plot. But as we can see for large data set it is difficult to view and also difficult to find the hidden trends and patterns in it. Cutting the scatter plot is not a trivial task especially when outliers are very important.

But Hierarchical clusters can also be represented as dendrogram and it is also very easy to cut without losing the quality of information.

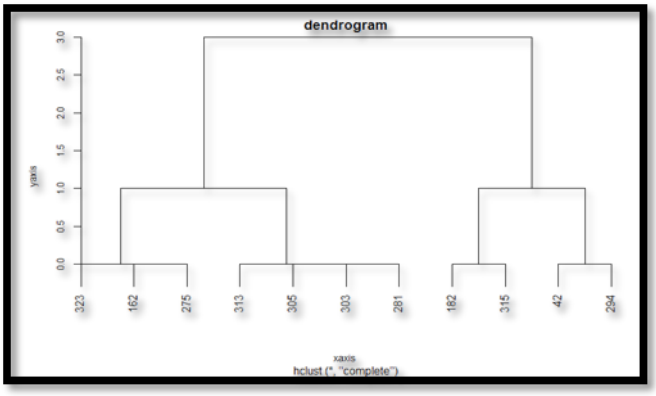

Figure IV: Cluster Output as Dendrogram

The comparison between K-means Recommender System and the proposed Recommender System are based on their running time and accuracy. The accuracy of the recommender systems are evaluated using Mahout Recommender Evaluator [5]. The time obtained is in seconds and the score is the variance between actual prediction and the computed prediction. The actual prediction is represented by zero.

Comparison in Running Time:

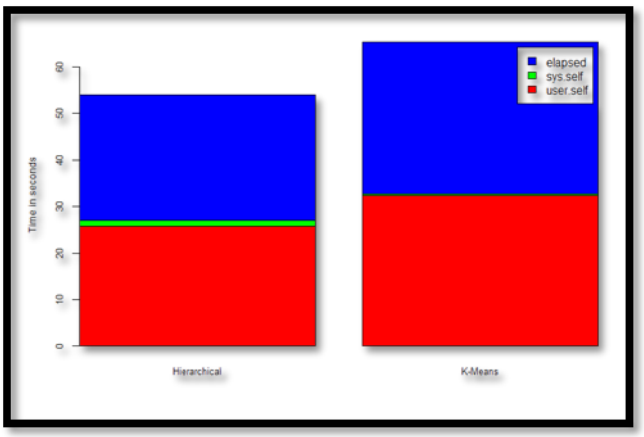

Figure V: Running Time Comparison

The Proposed Recommender Runtime :

```
user  system elapsed
 26.41    0.93   27.40
```

K-means Recommender Runtime:

```
user  system elapsed
 31.27    0.22   31.51
```

Evaluation of Recommender System by Mahout Evaluator.

The lower the score the better it is. It is evaluated between the score of 0 and 1.

- Improved Recommender = 1.1711346617284812
- K-means Recommender = 1.2933345853254894

IV. CONCLUSION

Recommender Systems has emerged as a recommendation tool to assist businesses and users to weed out irrelevant and extraneous information. These systems use a variety of techniques to help users identify the items that best fit their tastes or needs. While popular Collaborative Filtering -based algorithms continue to produce meaningful, personalized results in a variety of domains with better running time results. But still the magnitude of time taken to compute one prediction is much large and outweighs the benefits of recommendation. Though the popular algorithms are trying to minimize the variance between user taste and recommender's output but the hole was still large.

This new approach of the recommender system will efficiently predict or recommend items to the user. This Recommender System framework based on data mining techniques incorporates discrete functions and growing clusters and thus minimizing the variance between user's taste and predictors. Because of discrete algorithmic approach this recommender system has managed to decrease running time of the recommender system for each recommendation little bit. But since corporate business houses have to address thousands of such recommendation per seconds the net effect is quite satisfactory.

As observed from the result the new improved Recommender system based on Jaccard Index [3], Euclidean Similarity function and Hierarchical cluster gives better score and running time.

V. FUTURE SCOPE

The recommender system makes use of very large matrices who's space complexity is $O(mn)^k$.

Where,
m = No. of rows of a matrix.
n = No. of columns of a matrix.
k = Number of Matrices multiplied together.

Hence the calculations require a great amount of storage space and disk space and memory.
Memory operations can be contained up to a limit using Greedy Algorithm [4]. But storage space is required. So we need efficient compression techniques that can store and retrieve the sub-matrices as per demand.

And another problem is lots of online shopping sparcity problem *[1]* where the options are plenty but there is confusion to chosen one because there is very few rating on these items and not all products are rated according. Which means, first of all not all the products are reviewed properly, more over the rating is majorly divided when a lot of voting are given by different people it is easy to make a choice but when there are very few rating are given to all the product in the same category then it is very tough to check the authenticity of the rating there by making the decision to select all the more difficult. Several solutions have been proposed to overcome this problem like Dimensionality Reduction, Implicit ratings.

Dimensionality Reduction - By reducing the dimensionality of the information space we can depict N dimensional information in the lower dimensions this implies that different ratings given to different product can be used as a reference to infer the recommendation of the product to the user. But the process of reducing dimensionality leads to loss of information and visualization. These problem needs to overcome.
Implicit Ratings - Some mechanism should be adopted to increase the implicit ratings by inferring from the user's passive actions without having to let the each user rate explicitly. This problem also needs to be addressed.

ACKNOWLEDGMENT

Special thank you goes to those who contributed to this paper:

Dr. Kiran Jyoti for his valuable comments and sharing his knowledge.
Prof. Akshay Girdhar for making data he collected available.
The grouplens.org for hosting the dataset to be used for research purpose for free.

REFERENCES

1. *Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering.* **Zan Huang, Hsinchun Chen Daniel Zeng.** 1, s.l. : ACM Transaction on Information System, January 2004, Vol. 22.
2. *An Efficient k-Means Clustering Algorithm: Analysis and Implementation.* **Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman and Angela Y. Wu.** 7, s.l. : IEEE Transactgion on Pattern Analysis and Machine Interlligence, 2002, Vol. 34.

3. *Using of Jaccard Coefficient for Keywords Similiarity.* **Suphakit Niwattanakul, Jatsada Singhtongchai, Ekkachai Naenudorn and Supachanun Wanpu.** s.l. : Proceedings of the Interanational MultiConference of Engineers and Computer Scientists, 2013, Vol. 1.
4. *Greedy Algorithms for Classfication - Consistency, Convergence Rates, and Adaptivity.* **Shie Mannor, Ron Meir and Tong Zhang.** Cambridge : Journal of Machine Learning Research, 2003.
5. *A survey of Accuracy Evaluation Metrics of Recommendation Tasks.* **Shani, Asela Gunawardana and Guy.** s.l. : Journal of Machine Learning Research, 2009.
6. *Cloud Computing and Enterprise Resource Planning Systems.* **S L Saini, Dinesh Kumar Saini, Jabar H. Yousif and Sandhya V Khandage.** 2011, Proceedings of the World Congress on Engineerin.
7. *A survey of Collaborative Filtering Techniques.* **Khoshgoftaar, Xiaoyuan Su and Taghi M.** s.l. : Hindawi Publishing Corporation, August 2009, Advances in Artificial Intelligence, Vol. 2009, p. 19.
8. *Item-Based Top-N Recommendation Algorithms.* **Karypis, Mukund Deshpande and George.** 1, s.l. : ACM Transactions on Information System, January 2004, Vol. 22.
9. *Comparative Analysis & Evaluation of Euclidean Distance Function and Manhattan Distance Function Using K-means Algorithm.* **Karambir, Amit Singla and Mr.** 7, Kurukshetra : International Journal of Advance Research in Computer Science and Software Engineering, July 2012, Vol. 2. 2277 128X.
10. *Information Retrieval Perspective to Nonlinear Dimensionality Reduction for Data Visualization.* **Jarkko Venna, Jaakko Peltonen, Kristian Nybo, Helena Aidos and Samuel Kaski.** s.l. : Journal of Machine Learning Research, 2010.
11. *Research Paper On Clustering Techniques of Data Variations.* **Er. Arpit Gupta, Er. Ankit Gupta, Er. Amit Mishra.** 1, s.l. : International Journal of Advance Technology & Engineering Research, November 2011, Vol. 1. 2250-3536.
12. *A Novel Spectral Clustering Method Based on Pairwise Distance Matrix.* **Chi-Fang Chin, Arthur Chun-Chieh Shih and Kuo-Chin Fan.** Chungli : Journal of Information Science and Engineering, 2010.
13. *Efficient Active Algorithms for Hierarchical Clustering.* **Akshay Krishnamurthy, Sivaraman Baladrishnan, Min Xu and Aarti Singh.** Edinburgh : Proceedings of International Conference on Machine Learning, 2012.