https://dx.doi.org/10.22161/ijaems.4.2.3 *ISSN: 2454-1311*

Recommendation Based On Comparative Analysis of Apriori and BW-Mine Algorithm

Priyanka Makkar, Snehal Shintre

Department of Computer Engineering, Pune University, Pune, India

Abstract— *With The Growth of WWW recommending appropriate and relevant page to the user is a challenging task. In many web Applications, user would like to get recommendation based on their interest of surfing. Web Mining is used to extract relevant information for the user from logs, web content, hyperlinks etc. In this paper we will be using logs to recommend frequent access patterns to the users .This paper aims at using the logs of user ,cleaning logs , identifying users , identifying session , completing sessions from website structure and then using and comparing different recommendation algorithm like Apriori Algorithms and BW-Mine to recommend frequent items to the user. We will also be comparing different recommendations Algorithm with the help of example. The fundamental of finding access patterns with Apriori is that any set that occurs frequently must have its frequent subset. The fundamental of finding access pattern with BW-Mine, it constructs the WB-table, VI-List, and HI-Counter for finding frequent patterns.*
Keywords- Apriori Algorithm, BW-Mine Algorithm, Recommendations, Frequent access Pattern , Path Completion.

I. INTRODUCTION

With the tremendous amount of data available on WWW there is a need to extract useful information from the data and to use the same to improve performance and therefore Mining comes in picture. Mining is a important step in the knowledge discovery in databases .KDD process includes: [8]. a) Selection: Selecting relevant data for analysis from the database .b) Preprocessing: Removing noise and inconsistent data and combining multiple data sources. c) Transformation: Transforming data into appropriate forms to perform data mining .d) Mining: Choosing a data mining algorithm which is appropriate to pattern in the data; Extracting data patterns e)Interpretation/Evaluation : Interpreting the patterns into knowledge by removing redundant or irrelevant patterns; f)Translating the useful patterns into terms that human understandable.

Web mining research which is divided into three categories: (i) Web Content Mining: It can be defined as the scanning and mining of text, graphs and pictures from a Web page to find out the significance of the content to the search query (ii) Web Structure Mining: It is based on the hyperlinks, categorizing the Web pages and generated the information. Web structure mining describes the organization of the content of the web where structure is defined by hyperlinks between pages and HTML formatting commands within a page. (iii)Web Usage Mining: Web usage mining is described as applying data mining techniques on Web access logs to optimize web site for user's interest.[9]

Mining Tasks includes following task: a)Classification is finding models that analyze and classify a data item into several predefined classes. b) Regression is mapping a data item to a real-valued prediction variable. c)Clustering is identifying a finite set of categories or clusters to describe the data. d) Dependency Modeling (Association Rule Learning) is finding a model which describes significant dependencies between variables .e) Deviation Detection (Anomaly Detection) is discovering the most significant changes in the data. f) Summarization is finding a compact description for a subset of data[10]

In this paper we have focused on web usage mining and association rule mining, which provides us with the frequent pattern . This paper focused on the comparative study of two algorithms with the help of an example. The two algorithms used are Apriori -based which mine frequent patterns by repeatedly scanning the database and BW mine algorithm which create WB table, HI counter table level wise. One of the main advantage of BW Mine algorithm is it can run in parallel. Moreover, the algorithm works with both dense and sparse dataset. Furthermore, BW mines the frequent patterns, which can then form association rules, in parallel So works well for large dataset and therefore various real life applications can be solved using this algorithm [3]

The next section describes some related works. Section III presents the overall block diagram of the process and its explanation. Section IV describes BW-mine algorithm with the help of example .Section V provides us with frequent pattern using Apriori algorithm and is explained with the help of example. Section VI Shows Conclusions of Comparative Study of BW-mine and Apriori algorithm.

https://dx.doi.org/10.22161/ijaems.4.2.3 *ISSN: 2454-1311*

II. RELATED WORK

The most basic algorithm to find frequent patterns is the Apriori algorithm [1], It provides frequent pattern by repeatedly scanning the data set. This algorithm mines frequent patterns level-wise. It first finds the patterns of cardinality k and tests if each of them is frequent. Based on these frequent patterns of cardinality k , Apriori then generates candidate patterns of cardinality k+1. This process is applied repeatedly to discover frequent patterns of all cardinalities. Hence, it requires k scans of the dataset to find a frequent pattern of cardinality k [2][3].

The FP-growth algorithm [4] focus on the disadvantage of the Apriori algorithm and improves performance by using the transaction dataset and creating extended prefix-tree structure called FP-tree, from which frequent patterns are mined[5]. FP-growth scans the dataset only twice and removes the disadvantage of Apriori algorithm which scans the database till the cardinality of the discovered patterns.

The various algorithm has been proposed like H-mine algorithm. This algorithm avoids creating multiple FP trees[6]. First, a memory based, efficient pattern-growth algorithm, H-mine, is proposed for mining frequent patterns for the data sets that can fit into the main memory. A simple, memory based hyper-structure, H-struct, is designed for fast mining. Second, H-mine(Mem) has a polynomial space complexity and is thus more space efficient than pattern-growth methods such as FP-growth [7].

After this we have BW mine[3] algorithm which creates various tables by scanning the database only once and can work in parallel to provides us with frequent access pattern.

III. BLOCK DIAGRAM

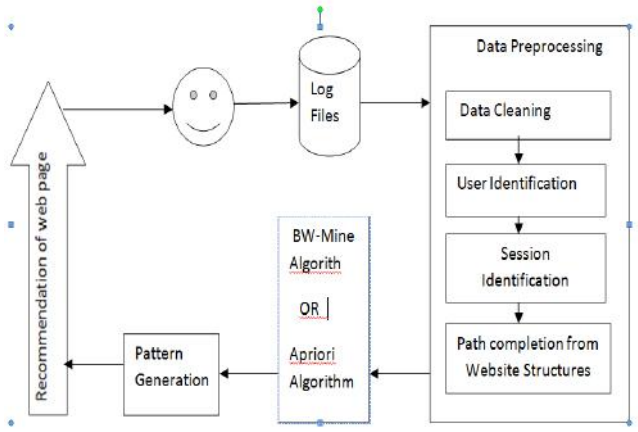

Fig. 1: Block diagram of the system

In this paper we will be using web surfer logs and recommend pages based on BW Mine and the same logs are used for Apriori Algorithm to recommend pages to the user and then we will be doing comparative study of recommendation done by both algorithm. As shown in Fig 1. we will be using Web surfer logs and preprocess them so that we can apply recommendation algorithm on it. Preprocessing includes data Cleaning, User identification, session Identification, Completion of the sessions from website structures. After preprocessing we will be applying recommendation algorithm and will compare the result of both and then recommend the pattern to the users.

Table.1: Sample web logs

WEBUSER	WEB PAGES
A1	2,5,9
A2	1,3
A3	1,5,8
A4	2,3,5,7
A5	1,2,3
A6	1,2,3,5
A7	1

The above table shows the sample web logs .In this example we are considering 7 users form A1-A7 and 10 web pages from 1 -10. We have considered the threshold of 2 i.e if the page is accessed by 2 or more users then that page is frequently accessed

IV. EXAMPLE SOLVED BY BW-MINE ALGORITHM

a) First data structure to be created is WB-table . After scanning log, we identify our pages of interest. We put 1 if i^{th} web page is accessed by user and 0 if the user has not accessed i^{th} page.

b) The next data structure used is Level-0 HI-Counter which stores counter of each webpage For each column c of the WB-table, we create Level-0 HI-Counter by counting number of 1s in the column c and put the count in the c-th position of the HI-Counter.

c) For each row r of the Web Based-table, we create a VI-list by recording the column index. These 3 tables can be built by using only one scan of the web log[3].

d) From each row r of the Level-0 VI- List, BW-mine consider respective rows in WB-table to obtain the database for X . and the same way level -1 H1 counter can be created. This process is repeated until all patterns are generated.

https://dx.doi.org/10.22161/ijaems.4.2.3 *ISSN: 2454-1311*

WB TABLE OF Web Log

USER	1	2	3	4	5	6	7	8	9	10
A1	0	1	0	0	1	0	0	0	1	0
A2	1	0	1	0	0	0	0	0	0	0
A3	1	0	0	0	1	0	0	1	0	0
A4	0	1	1	0	1	0	1	0	0	0
A5	1	1	1	0	0	0	0	0	0	0
A6	1	1	1	0	1	0	0	0	0	0
A7	1	0	0	0	0	0	0	0	0	0

LEVEL - 0 HI- COUNTER

	1	2	3	4	5	6	7	8	9	10
Level 0	5	4	4	0	4	0	1	1	1	0

Level- 0 VI-list

WEBPAGE	WEB USER
1	A2,A3,A5,A6,A7
2	A1,A4,A5,A6
3	A2,A4,A5,A6
4	
5	A1,A3,A4,A6
6	
7	A4
8	A3
9	A1
10	

Fig.2: Level 0 WB table,HI-Counter &VI list table

As the threshold is 2 .The frequent pattern derive from Figure2 are {1}{2}{3}{5}.Now next level can be constructed using above tables. The next level can be constructed in parallel.

Figure 3 show the next level WB table ,HI counter &VI list table. The shows {1,2},{1,3},{1,5} are frequent patterns. These patterns are generated with respect to page 1.

WB Table for webpage{1}

USER	2	3	4	5	6	7	8	9	10
A2	0	1	0	0	0	0	0	0	0
A3	0	0	0	1	0	0	1	0	0
A5	1	1	0	0	0	0	0	0	0
A6	1	1	0	1	0	0	0	0	0
A7	0	0	0	0	0	0	0	0	0

LEVEL - 1 HI- COUNTER FOR {1}

	2	3	4	5	6	7	8	9	10
Level 1	2	3	0	2	0	0	1	0	0

WEBPAGE	WEB USER
1&2	A5,A6
1&3	A2
1&5	A3

Level-1 VI-list

Fig.3: Level 1 for page 1 WB table ,HI-Counter &VI list table

USER	3	4	5	6	7	8	9	10
A1	0	0	1	0	0	0	1	0
A4	1	0	1	0	1	0	0	0
A5	1	0	0	0	0	0	0	0
A6	1	0	1	0	0	0	0	0

WB Table for webpage{2}

	3	4	5	6	7	8	9	10
Level 1	3	0	3	0	1	0	1	0

WEBPAGE	WEB USER
2&3	A4,A5,A6
2&5	A1

Level-1 VI-list for {2}

Fig.4: Level 1 for page 2 WB table,HI-Counter &VI list table

Figure4 show the Level-1 WB table ,HI counter &VI list table for page 2. The above table shows {2,3},{2,5} are frequent pattern .These patterns are generated with respect to page 2.

Figure 4 show the level-1 WB table ,HI counter &VI list table for page 3. The shows {3,5} is frequent pattern.

WB TABLE OF FOR {3}

USER	4	5	6	7	8	9	10
A2	0	0	0	0	0	0	0
A4	0	1	0	1	0	0	0
A5	0	0	0	0	0	0	0
A6	0	1	0	0	0	0	0

LEVEL - 1 HI- COUNTER FOR {3}

	4	5	6	7	8	9	10
Level 1	0	2	0	1	0	0	0

Level-1 VI-list for {3}

WEBPAGE	WEB USER
3&5	A4,A6

Fig.5: Level 1 for page 3 WB table,HI-Counter &VI list table

https://dx.doi.org/10.22161/ijaems.4.2.3 *ISSN: 2454-1311*

WB Table for webpage{5}

USER	6	7	8	9	10
A1	0	0	0	1	0
A3	0	0	1	0	0
A4	0	1	0	0	0
A6	0	0	0	0	0

LEVEL - 1 HI- COUNTER FOR {5}

	6	7	8	9	10
Level 1	0	1	1	1	0

Fig.6: Level 1 for page 5 WB table,HI-Counter

Figure 6 shows that there is no frequent item set with respect to page{ 5}.Figure 7 shows next level of frequent patterns with respect to {1,2}.The results shows that {1,2,3} is frequent item set..By repeating the above process we find that their is no frequent item set for pages{1,3},{1,5}

WB Table for webpage{1,2}

USER	3	4	5	6	7	8	9	10
A5	1	0	0	0	0	0	0	0
A6	1	0	1	0	0	0	0	0

LEVEL - 2 HI- COUNTER FOR {1,2}

	3	4	5	6	7	8	9	10
Level 2	2	0	1	0	0	0	0	0

Level-2 VI-list for {1,2}

WEBPAGE	WEB USER
1,2,3	A5,A6

Fig.7: Level 2 for page {1,2} WB table,HI-Counter &VI list table

WB Table for webpage{2,3}

USER	4	5	6	7	8	9	10
A4	0	1	0	1	0	0	0
A5	0	0	0	0	0	0	0
A6	0	1	0	0	0	0	0

LEVEL - 3 HI- COUNTER FOR {2,3}

	4	5	6	7	8	9	10
Level 3	0	2	0	1	0	0	0

Level-3 VI-list for {2,3}

WEBPAGE	WEB USER
2,3,5	A4,A6

Fig.8: Level 2 for page {2,3} WB table,HI-Counter &VI list table

The results shows that {2,3,5} is frequent item set as shown in Figure 8. By repeating the above process we find that their is no frequent item set for pages{2,3},{2,5} and similarly using same process we find that there is no frequent pattern of next level from {1,2,3},{2,3,5} and therefore process terminate.

Therefore from the above algorithm the frequent pattern generated are {1},{2},{3},{5},{1,2}{1,3}{1,5},{2,3}, {2,5}{3,5}, {1,2,3},{2,3,5}.

V. EXAMPLE SOLVED BY APRIORI ALGORITHM

Considering the same logs as shown in Table1 constructing table of cardinality 1.

Table.2: Support count of each page

Pages	Support Count
{1}	5
{2}	4
{3}	4
{4}	0
{5}	4
{6}	0
{7}	1
{8}	1
{9}	1
{10}	0

Frequent pattern generated from above table are{1},{2},{3}{5}

https://dx.doi.org/10.22161/ijaems.4.2.3 *ISSN: 2454-1311*

Table.3: Support count of cardinality 3

Pages	Support Count
{1,2}	2
{1,3}	3
{1,5}	2
{2,3}	3
{2,5}	3
{3,5}	2

The above table shows all are frequent pattern

Table.4: Support count of cardinality 3

Pages	Support Count
{1,2,3}	2
{1,2,5}	1
{1,3,5}	1
{2,3,5}	2

The above table shows the frequent pattern are {1,2,3},{2,3,5}

As seen from above table the frequent pattern are generated by scanning database thrice and the frequent patterns are

{1},{2},{3},{5},{1,2}{1,3}{1,5},{2,3},{2,5}{3,5}, {1,2,3},{2,3,5}

The various Association rules that can be generated from above algorithm are

Table.5: Association rule and their Confidence

{1}	{2} C=.40, {3} C=.60, {5}C=.40, {2,3}C=.40
{2}	{3}C=.75,{5} C=.75,{1}C=0.5,{1,3}C=0.5,{3,5}=0.5
{3}	{1}C=.75,{2}C=.75,{5}C=0.5,{1,2}C=0.5,{2,5} C=0.r
{5}	{1}C=0.5,{2}C=.75,{3}C=0.5,{2,3}C=0.5
{1,2}	{3}C=1
{1,3}	{2}C=.67
{2,3}	{5}C=.67
{2,3}	{1}C=.67
{3,5}	{2}C=1
{2,5}	{3}C=.67

Considering the threshold for confidence is .50.Delete the rule which has confidence less than .50.

Table 5 specify that the is confidence of 60% that if the user access page 1, page 3 is recommended. If the user access pages {3,5} then there is 100% confidence that page 2 is recommended to the user.

VI. CONCLUSION

In this paper, We have focused on recommending frequent pattern to user. A comparative study was made between Apriori algorithm and BW-mine algorithm. Both the algorithms are explained with the help of an example. Apriori algorithm is the easy and basic to use.. As shown in example it scan the dataset again and again to find frequent item sets. For smaller dataset this algorithm can be used but for a big data it is not efficient. Whereas BW-Mine algorithm scan the dataset only once and create multiple tables as shown in example. This algorithm can generate frequent item set in parallel so it is efficient for problem having big data also. Therefore real life problem can be solved more efficiently using BW-Mine algorithm

REFERENCES

[1] Mohammed Al-Maolegi , Bassam Arkok," An Improved Apriori Algorithm For Association Rules", International Journal on Natural Language Computing (IJNLC) Vol. 3, No.1, February 2014.

[2] R. Aggarwal and R. Srikant "Fast algorithms for mining association rules "In Proc. VLDB 1994, 487-399.

[3] Carson K. Leung , Fan Jiang , Adam G.M. Pazdor ,"Bitwise Parallel Association Rule Mining for Web Page Recommendation " In Proceedings of WI '17, Leipzig, Germany, August 23-26, 2017.

[4] Han, J. Pei, and Y. Yin,"Mining frequent patterns without candidate generation", In Proc ACM SIGMOD 2000, 1-12.

[5] M Suman ,T Anuradha ,K Gowtham, "A Frequent Pattern Mining Algorithm Based On Fp-Tree Structure Andapriori Algorithm " ,International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 ,Vol. 2, Issue 1, Jan-Feb 2012, pp.114-116.

[6] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan,"Automatic subspace clustering of high dimensional data for data mining applications" In SIGMOD'98, pages 94–105.

[7] Jian Pei1, Jiawei Han2, Hongjun Lu3, Shojiro Nishio4, Shiwei Tang5 And Dongqing Yang ,"H-Mine: Fast and space-preserving frequent pattern mining in large databases" Received February 2004 and accepted, IIE Transactions (2007).

[8] Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. (1996)."From Data Mining to Knowledge Discovery in Databases".AI Magazine, 17(3), 37-54.

[9] Han, J. &Kamber, M. (2012). "Data Mining: Concepts and Techniques". 3rd.ed. Boston: Morgan Kaufmann Publishers.

[10] Tipawan Silwattananusarn, Dr. KulthidaTuamsuk," Data Mining and Its Applications for Knowledge Management : A Literature Review from 2007 to 2012", International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.2, No.5, September 2012.