International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

Mining Weighted Association Rules Using Probabilistic and Combinational Approach

A I Liton[1], M A Rahman[2], T Rahman[3]

[1, 2, 3]Lecturer, Department of CSE, University of South Asia, Bangladesh

Abstract: ***Association Rule Mining (ARM) is one of the most popular data mining techniques. Weight Association rule mining (WARM) is adapted to handle weighted associated mining problems where each item is allowed to have a weight. The goal is to steer the mining focus to those significant relationships involving items with significant weights rather than being flooded in the combinatorial explosion of insignificant relationships. Predictive models developed by applying Data Mining techniques are used to improve forecasting accuracy in the airline business. In this paper, we apply data mining techniques to real airline frequent flyer data in order to derive customer relationship and recommendations. We are going to introduce a new measure using HIPRO & Apriori algorithm, on the passenger database system of an Airline.***

Keynotes: Data Mining, Weight Association Rules, WARM, Probabilistic, HIPRO

1. Introduction

We are in an age often referred to as the information age. In this information age, because we believe that information leads to power and success, and thanks to sophisticated technologies such as computers, satellites, etc., we have been collecting tremendous amounts of information like business transactions, scientific data, medical data, satellite data, surveillance video & pictures, world wide web repositories to name a few. With the enormous amount of data stored in files, databases, and other repositories, it is increasingly important, if not necessary, to develop powerful means for analysis and perhaps interpretation of such data and for the extraction of interesting knowledge that could help in decision-making.

Data Mining, also popularly known as Knowledge Discovery in Databases (KDD) [1], refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. While data mining and knowledge discovery in databases (or KDD) are frequently treated as synonyms, data mining is actually part of the knowledge discovery process.

The kinds of patterns that can be discovered depend upon the data mining tasks employed. By and large, there are two types of data mining tasks: descriptive data mining [2] tasks that describe the general properties of the existing data, and predictive data mining tasks that attempt to do predictions based on inference on available data.

One of the popular descriptive data mining techniques is Association rule mining (ARM), owing to its extensive use in marketing and retail communities in addition to many other diverse fields. Mining association rules is particularly useful for discovering relationships among items from large databases. Association rule mining deals with market basket database analysis for finding frequent item sets and generate valid and important rules. Various association rules mining algorithms have been proposed [3], [4]. Other algorithms for finding frequent item sets include pincer search [5], FP (frequent pattern) tree [6]. Apriori- generation function follows bottom- up approach. Pincer search algorithm o finds frequent item sets but it follows both bottom2up and top-down approach. Frequent pattern tree also generate frequent item sets without candidate generation.

1.1 Probability based approaching Algorithm

The most important step in mining association is generation frequent item sets. In algorithm apriori the most time is consumed by scanning the database repeatedly. It would reduce the running time of the algorithm by reducing the times it scans the database far and away. In this paper a method of mining frequent item sets by evaluating their probability of supports based on association analyzing were mentioned. First, it gained the probability of every 1-itemset by scanning the database, the 1-itemset with the more larger support than the probability the user sets would be frequent 1-itemsets[7][8]. Second, it evaluates the probability of every 2-itemset, every 3-item sets, and every k- item set from the frequent 1-itemsets[9]. Third, it gains the entire candidate frequent item sets [10]. Fourth, it scans the database for verifying the support of the candidate frequent item sets, last the frequent item sets are mined and association rules also do [11]. In the method it reduces a lot of times of scanning database and shortened the calculate time of the algorithm [12].

The improved algorithm for apriori:
Let P1, P2…Pn are the independent probability of every item A1, A2…An, the probability for any two item Ak, Am (Pk<Pm)both appeared in one transaction is Pkm[13].

If Ak and Am are total non-correlation, from definition 3 it can be concluded that Pkm = Pk*Pm, if Ak and Am are total correlation, then Pkm is the minimum of the Pk and Pm that is Pk , so , $Pk*Pm \leq Pkm \leq Pk$.

Now the problem is:
Given Pk and Pm, also $Pk*Pm \leq Pkm \leq Pk$, Please evaluate Pkm . The problem couldn't be solved with the conditions in mathematics. But in fact, there is a lot of information without accurate mathematic formula which be omitted. In this paper it offered a method by association analysis to confirm the formula.

Let parameter a be the probability which Ak and Am are total correlation, and parameter b for total non-correlation. a+b = 1, 0 < a, b < 1, then Pkm can be defined as the following formula
Pkm=a*Pk+b*Pk*Pm

Determination parameter "a" and "b":

There are a series of criterion about environment. The most ingredients of the pollution can be confirmed based on the source. So we can consider the criterion as a referenced list and the list needed to find the correlation as a comparison list. Then we'll get the correlation coefficient which is the parameter "a" in our formula (2), and b=1-a. The details as below:

Let S={S_1, S_2, …Sm} be the value list of item Am, S_1, S_2, …Sm are sample extracted from the DB and X={ X_1, X_2,…X_m }be the value list for item Ak , X_1, X_2, …X_m are sample extracted from the DB[14].

1.2 Algorithm for Combinational Approach-HIPRO

1. Let "a" is the vector of authority and "h" be the vector of hub
2. Initialize all weights to 1
3. Add to all weights of authorities pointed b an item in database D.
4. In each iteration calculate the authority weight for each item in database D; for every hub P
5. In each iteration calculate the hub weight for each item in database D; for every authority Q
6. After new weight are computed for all nodes, the weights are normalized
7. While Auth(i) and Hub(i) do not converge
8. Count the probability of each attribute item
9. The probability of any two items A_k and A_m appeared synchronously in one record is P_{km}. min(P_k , P_m)≤P_{km} ≤P_k *P_m , if A_k and A_m is total correlation, then the P_{km} is the minimum of the P_k and P_m, [17]; if A_k and A_m is total independent, then the P_{km} is P_k *P_m; So we can estimate :
P_{km} =(a*min(P_k, P_m)+b*P_k*P_m)/(a+b); a+b=1
10. If P_{km} is more than the threshold value which the user set, then A_k, A_m are the frequent item sets.
11. Count the support of the frequent item sets by scanning the DB another time.
12. Output the association rules from the frequent item sets [15] [16].

2. Proposed Model

If we analyze any airline database, it will be seen their takes a lot of transactions and definitely the item set of the transactions have some association among them. For example analyze the following transactions of an airline database system:

The passengers who travel to domestic route usually buy economy class tickets. Again passenger who travel to domestic route usually travel in the weekend and at evening. Similarly all the other itemsets are related to each other with various rules. At first we apply the HITS algorithm with the association of traditional apriori algorithm.

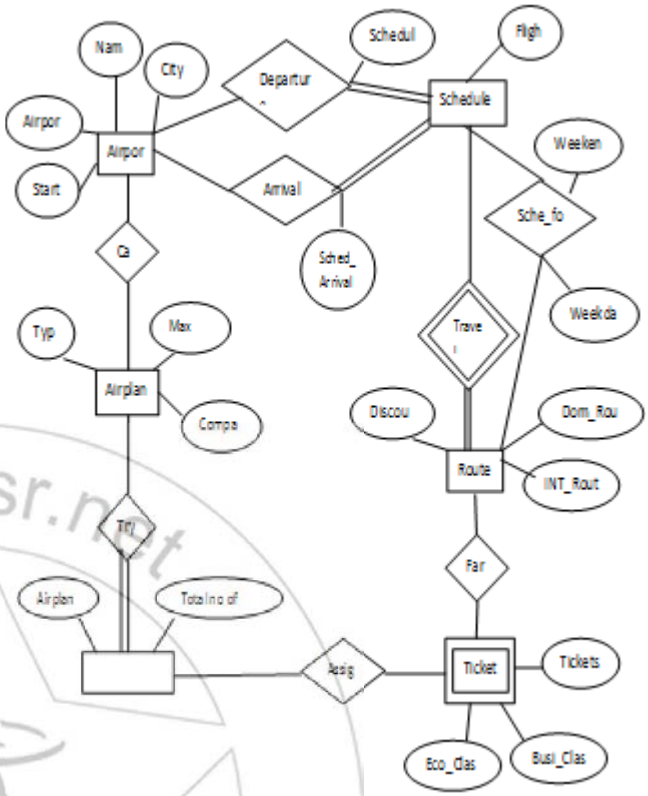

Figure 1: An ER Diagram for an Airline Database

Data Representation for an airline database transaction

Item Transaction	Auth (0) Domestic route	Auth (1) Economy class ticket	Auth (2) Weekend	Auth (3) Evening	Auth (4) Weekday
1 or Hub(0)	1	1	0	0	0
2 or Hub(1)	1	0	1	1	0
3 or Hub(2)	0	1	1	1	0
4 or Hub(3)	1	1	1	1	0
5 or Hub(4)	1	1	0	1	1
6 or Hub(5)	1	0	0	0	1

Representation of Hub weights

Transaction ID	*Transaction*	*Hub Weights*
1	Hub(0)	0.293
2	Hub(1)	0.409
3	Hub(2)	0.391
4	Hub(3)	0.547
5	Hub(4)	0.496
6	Hub(5)	0.212

Total Hub Weight=2.348

W-Support(DR)=(0.293+0.409+0.547+0.496+0.212)/2.348 = 0.833
W-Support(ECT) = (0.293+0.391+0.547+0.496)/2.348 = 0.735
W-Support(Wk) = (0.409+0.391+0.547+0.496)/2.348 = 0.785
W-Support (Ev) = (0.409+0.391+0.547)/2.348 = 0.573
W-Support (Wd) = (0.496+0.212)/2.348 = 0.301

Representation of W-support corresponding 1-Itemset

1-Itemset	W-support
Domestic Route(DR)	0.833
Economy Class Ticket(ECT)	0.735
Weekend(Wk)	0.785
Evening(Ev)	0.573
Weekdays(Wd)	0.301

3. Analysis

So far we have seen the improved apriori algorithm based on probability and then use of HITS algorithm with the association of traditional apriori algorithm with very phenomenal examples. Earlier we have seen the application of the two algorithms individually on individual examples and then both algorithms were applied on an airline database.

After the completion of the simulation there are some significant changes which have seen in the results. Now if we compare both results we can identify the depth of changes.

Comparison of the algorithms:

The comparison of two algorithms is given bellow:

Frequent Item set	*HITS algorithm combine with traditional apriori algorithm*	*HIPRO algorithm combination of HITS and Probability based apriori algorithm*
1-Itemset	{DR},{ECT}, {Wk}, {Ev}	{DR}, {ECT}, {Wk}, {Ev}
2-Itemset	{DR, ECT}, {DR, Wk},{DR, Ev}, {ECT, Wk}	{DR, ECT}, {DR, Wk}, {DR, Ev}, {ECT, Wk}
3-Itemset		{DR, ECT, Wk}, {DR, ETC, Ev}, {DR, Wk, Ev},{ECT, Wk, Ev}

The analysis of the results

If we analyze the results of two algorithms we see that when HITS algorithm is used with the traditional apriori algorithm some frequent item sets are missed which shows that this algorithm is not efficient. On the other hand HIPRO algorithm did not miss the frequent item sets which were missed by the previous algorithm. So definitely we can tell that it increases the efficiency of weighted association rule mining as well as data mining.

Analysis of the time and space complexity:

In the previous algorithm where apriori algorithm is used in association with HITS algorithm, all the candidate item sets with the same length must be stored in the man memory, which results in a waste of space. To generate large item sets the database is passed as many times as the length of the longest larger item sets. The database is scanned and the support of each candidate item set is counted after the new candidate item sets are generated, which results in waste of time for large database.

In contrast, HIPRO algorithm helps us to get rid of these problems by allowing the system not to store all the candidate item sets in the memory and pass over the database only once. It finds out all the high frequency 1-dimensional data item sets and then they are used to identify all the high frequency 2-dimensional data item sets and so on.

Now suppose, in the traditional algorithm |Lk-1| indicates the number of data item sets in Lk-1, P=|Ck| indicates the number of data item sets in Ck. Now as the number of n elements set's subsets is 2n, therefore this algorithm needs a total of 2p*|Lk-1| times operations.

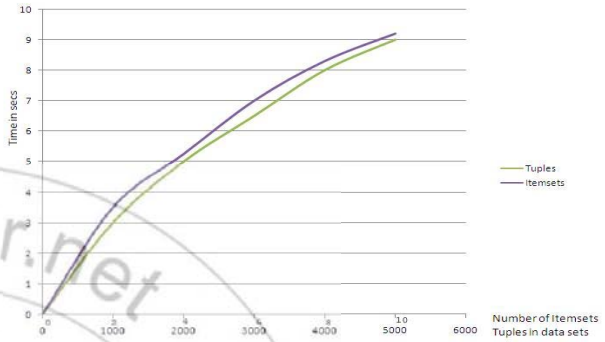

Figure: The graph for execution time of existing algorithm

The above graph shows that as the number of tuples and item sets increase the execution time also increases. In case of HIPRO algorithm let |PFAk-1[n]| indicates the number of data item sets in PFA[n], p=|PUAk-1[m]| indicates the number of data item sets in PUAk.

This algorithm needs a total of p*|PFAk-1[n]| times operations

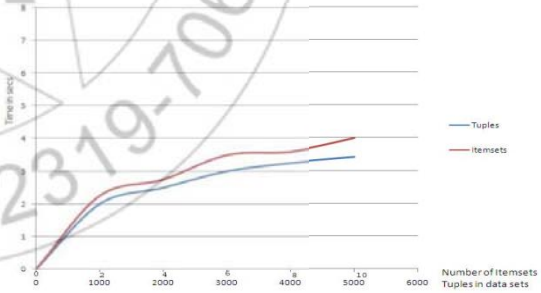

Figure: The graph for execution time of HIPRO algorithm

The above graph shows the effect of algorithm HIPRO as the tuples and number of item sets grows. Now if we plot the both graphs in a single plot area we will be able to see the comparison in the execution time as the number of tuples as well as item sets grows.

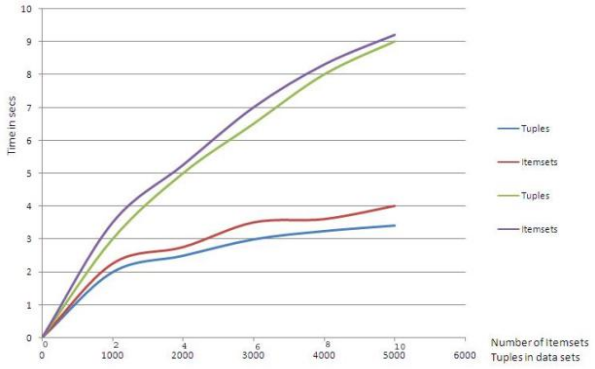

Figure: The comparison of the algorithms execution time

The first data set we use has 1000 tuples, then increase 1000 tuples every time. We test the execution time of the algorithms with respect to number of tuples and itemsets. Figure shows that: With tuples from 0 to 5000, when the number of tuples are small, both algorithms have similar performance. However, as the number of tuples grows, the algorithm HIPRO takes effect. It keeps the runtime low. In contrast, the previous algorithm i does not scale well under large number of tuples. It also shows the execution time of both algorithms with respect to the item sets increased. As the number of item sets goes up , the runtime of both algorithms has increases and the algorithm HIPRO grows slower than the previous algorithm.

4. Conclusion

An efficient way for discovering the frequent itemset can be very useful in various data mining problems, such as discovery of association rules. In this Thesis, new approaches to association rule mining has been explored in depth. The thesis was mainly focused on weighted association rule mining without pre-assigned weights using w-support and using algorithm HIPRO (combination of HITS and probability based apriori algorithm). The comparison of the algorithms, were done by applying them on real life data set. It was found the algorithm which is proposed in this paper is more advantageous over the previous algorithm.

5. Future Scope

For our approach, the related information may not fit in the main memory when the size of the database is very large. This problem should be considered by reducing the memory space requirement. Also, the approach we introduced in this paper should be applied on different applications, such as document retrieval and resource discovery in the World Wide Web environment. Best part of previously known algorithms can be combined with to develop hybrid approaches which perform best for all cases. Number of solutions has been presented; but still a lot of research is possible in this particular area.

And last but not the least; here also we are dealing with the time-space tradeoff problem. As the size of frequent itemset increases, computational time for the initial phases increases exponentially with increase in the requirement in memory space. So, a better way to consider only the relevant transaction or items can be possible field of research. If data cannot fit in the memory than more page faults may occur resulting in the decrease in the performance of the system.

References

[1] R. Agrawal, T. Imielinski, and A. Swami, “Mining Association Rules between Sets of Items in Large Datasets,” Proc. ACM SIGMOD ’93, pp. 207-216, 1993.
[2] Rakesh Aggarwal , Ramakrishanan Srikant, "Fast Algorithm for mining Association Rules", IBM Almaden Research Centre, Proceedings of 20th VLDB Conference, Santiago, Chile, 1994
[3] Lin D., Z. M. Kedem, “Pincer-Search: An Efficient Algorithm for Discovering the Maximum Frequent Set”, IEEE Tran. Know. and Data Engg., Vol. 14, No. 3, pp.553-556, May/June 2002.
[4] J.Han, J.Pei, and Y Yin, “Mining Frequent Patterns Without Candidate Generation”, Proc. ACM SIGMOD 2000.
[5] K.Sun and F.Bai, "Mining Weighted Association Rules Without Preassigned Weights", IEEE Transactions on Knowledge and Data Engineering, Vol. 20, No. 4, pp. 489-495, April 2008.
[6] J.M. Kleinberg, “Authoritative Sources in a Hyperlinked Environment,”J. ACM, vol. 46, no. 5, pp. 604-632, 1999
[7] Li Pingxiang , Chen Jiangping , Bian Fuling ,” A Developed Algorithm Of Apriori Based On Association Analysis”, Geo-Spatial Information Science vol.7,no.2, 108-112, DOI: 10.1007/BF02826646, 2004
[8] Sotiris Kotsiantis, Dimitris Kanellopoulos, “Association Rules Mining: A Recent Overview”, GESTS International conference on computer science and engineering, Vol.32(1) pp- 71-82, 2006
[9] Baralis, E., Psaila, G., “Designing templates for mining association rules”, Journal of Intelligent Information Systems, 9(1):7-32, July 1997.
[10] Cristofor, L., Simovici, D., “Generating an informative cover for association rules”, Proc. of the IEEE International Conference on Data Mining, 2002.
[11] Brin, S., Motwani, R. and Silverstein, C., “Beyond Market Baskets: Generalizing Association Rules to Correlations”, Proc. ACM SIGMOD Conf., pp. 265-276, May 1997.
[12] Ashrafi, M., Taniar, D. Smith, K., “A New Approach of Eliminating Redundant Association Rules”, Volume 3180, pp. 465 – 474, 2004.
[13] Ashrafi, M., Taniar, D., Smith, K., “Redundant Association Rules Reduction Techniques”, Volume 3809, pp. 254 – 263, 2005,
[14] Techapichetvanich, K., Datta, A., “Visual Mining of Market Basket Association Rules”, Volume 3046, pp. 479 – 488, Jan 2004.
[15] Omiecinski, “Alternative Interest Measures for Mining Associations in Data- bases”,IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No. 1, pp. 57-69.
[16] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S., “ Dynamic itemset counting and implication rules for

market basket data", ACM SIGMOD International Conference on Management of Data, pp.255-264. May 13-15, 1997,

[17] Liu, B. Hsu, W., Ma, Y., "Mining Association Rules with Multiple Minimum Supports," Proc. Knowledge Discovery and Data Mining Conf., pp. 337-341, Aug. 1999.

[18] Savasere, A., Omiecinski, E., Navathe, S., "Mining for strong negative associations in a large database of customer transactions", Proc. of ICDE, pp.494–502, 1998.