ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

Mining Big Sources using Efficient Data Mining Algorithms

Shoban Babu Sriramoju

Associate Professor, Department of Computer Science and Engineering, Varadha Reddy Engineering College, Warangal, India [1]

Abstract: Data mining algorithms are widely used in the real world application in order to discover knowledge from large data sources. These algorithms work on historical data to analyze data in order to bring about trends or patterns. Association rule mining or frequent item set mining is very useful in applications like inductive databases, query expansion and others. A frequent itemset is the itemset when a set of records are repeated for specified number of times in a given dataset. When such frequent itemset is no present in other frequent itemset, it is named as maximal itemset. When it is not as part of other itemset, them it is called closed itemset. These itemsets are used to extract patterns or trends in the real world applications that support in decision making. Recently Uno et al. proposed data mining algorithms to discover maximal itemsets, closed itemsets and frequent itemsets. In this paper we practically explore those algorithms. We implement them in a prototype application and the empirical results reveal that they are very useful for many data mining solutions.

Keywords: Data mining, big databases, association rule mining, frequent itemsets

I. INTRODUCTION

For many years data mining is the domain which provided techniques or algorithms for processing huge amount of data in order to obtain meaningful information that helped in making well informed decisions. With these techniques expert systems have been built that are used to make high quality decisions. The data mining techniques that have been used in the real world include K-means, C4.5, Apriori, Expectation Maximization, Page Rank, kNN, AdaBoost, Support Vector Machine (SVM), CART (Classification and Regression Trees) and Naïve Bayes algorithm. These are in the top 10 algorithm that exists in the world for knowledge discovery. Apriori, for instance, is used to obtain frequent patterns that can be used in decision making. C4.5 is used for making clusters of given objects while K-means is also used for doing the same and it is one of the famous clustering algorithms.

Association rule mining and frequent itemset mining is very useful technique in data mining domain. It discovers the frequently repeated itemsets in the given data source as per the given support and confidence. Generally domain experts provide the required support and confidence. The applications of such technique include query expansion, association rule mining and inductive databases. Implementations of frequent itemsets [1], [2] can be used to mine huge amount of business data in order to make policies that can help companies in the long run. The business intelligence thus obtained can help making profits so as to excel in business. Therefore mining actionable knowledge has very important utility in the real world.

Recently Uno et al. [3] proposed algorithms for discovering itemsets. These algorithms can obtain closed itemsets, frequent itemsets and closed itemsets [4], [5] and [6]. These algorithms are very useful as they are efficient in obtaining essential knowledge. In this paper these algorithms are practically implemented using Java as development platform. The prototype application we built to demonstrate the efficiency of algorithms [7] is user friendly. The experimental results revealed that the application is very useful in the real world for efficient decision making. The remainder of this paper is structured as follows. Section II provides preliminaries pertaining to itemset mining. Section III provides details of proposed algorithms. Section IV presents prototype information. Section V presents experimental results while section VI concludes the paper.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

II. LITERATURE SURVEY

Itemset is nothing but a set of items that can be denoted as I = {1, 2,3, …, n}. Consider T represents a transaction datasets which has many records that are denoted as T= {t1, t2, t3, …, tn). Consider an itemset P for illustration of an example. When any transaction contains P which is known as the occurrence of P which is denoted as T(P) which represents all transactions in the given dataset. The frequency of P can be denoted as |T(P)| which can also be represented as frq(P). The frequency is considered when the given confidence and support are satisfied [8]. When frequent itemset is not as part of any other frequent itemset, such one is known as maximal itemset [9]. When any itemset is not a part of any other itemset, it is known as closed itemset.

III. ALGORITHMS

The data mining algorithms presented recently by Uno et al. [10] are explored here. These algorithms are used to obtain maximal itemsets, closed itemsets and frequent itemsets. Figure 1 shows an algorithm that is used to mine the frequent itemsets.

```
ALGORITHM BackTracking (P:current solution)
1. Output P
2. For each e ∈ I, e > tail(P) do
3.     If P ∪ {e} is frequent then
           call BackTracking (P ∪ {e})
```

Fig. 1 – Backtracking algorithm for frequent itemsets

As shown in Fig. 1, it is evident that the algorithm executes in recursive fashion. It is a recursive mechanism as it invokes itself every time. It takes dataset as input and generated frequent itemsets that are represented by P. Figure 2 presents an algorithm that is used to extract closed itemsets.

```
Algorithm LCM()
1. X := I(T(∅))   /* The root ⊥ */
2. For i := 1 to |E|
3.   If X[i] satisfies (cond2) and (cond3) then
       Call LCM_Iter( X[i], T(X[i]), i ) or
       Call LCMd_Iter2( X[i], T(X[i]), i, DT )
         based on the decision criteria
4. End for

LCM_Iter( X, T(X), i(X) )   /* occurrence deliver */
1. output X
2. For each T ∈ T(X)
     For each j ∈ T, j > i(X), insert t to J[j]
4. For each j, J[j] ≠ ∅ in the decreasing order
5.   If |J[j]| ≥ α and (cond2) holds then
       LCM_Iter( T(J[j]), J[j], j )
6.   Delete J[j]
7. End for

LCM_Iter2( X, T(X), i(X), DT )   /* diffset */
1. output X
2. For each i, X[i] is frequent
3.   If X[i] satisfies (cond2) then
4.     For each j, X[i] ∪ {j} is frequent,
         DT'[j] := DT[j] \ DT[i]
5.     LCM_Iter2( T(J[j]), J[j], j, DT' )
6.   End if
7. End for
```

Fig. 2 – Linear time Closed itemset Mining Algorithm [11]

As shown in figure 2, closed itemsets are extracted by this algorithm. The algorithms work on given dataset in order to generate closed itemsets as output that satisfies given support and confidence. Figure 3 presents an algorithm that can generate maximal itemsets.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

```
ALGORITHM LCMmax (P:itemset, H:items to
                  be added)
1. H' := the set of items e in H s.t. P ∪ {e} is frequent
2. If H' = ∅ then
3.     If P ∪ {e} is infrequent for any e then
         output P ; return
4.     End if
5. End if
6. Choose an item e* ∈ H' ; H' := H' \ {e*}
7. LCMmax (P ∪ {e}, H')
8. P' := frequent itemset of the maximum size
       found in the recursive call in 7
9. For each item e ∈ H \ P' do
10.     H' := H' \ {e}
11.     LCMmax (P ∪ {e}, H')
12. End for
```

Fig. 3 –Algorithm for Discovering Maximal Itemsets

As shown in figure 3, the algorithm takes dataset as input and works on it for computing maximal datasets. The results generated by the algorithm are as per the given support and confidence given by the domain expert.

IV. BUILDING PROTOTYPE

We built a prototype application that is used to test the efficiency of the implemented algorithms. The application has been built in Java platform which provides user-friendly interface. The environment used to build the application includes a PC with 4GB RAM, Core 2 dual processor running Windows 7 operating system. The prototype is meant for performing various mining operations such as mining closed itemsets, frequent itemsets, and maximal itemsets. The results of algorithm can be viewed in figure 4.

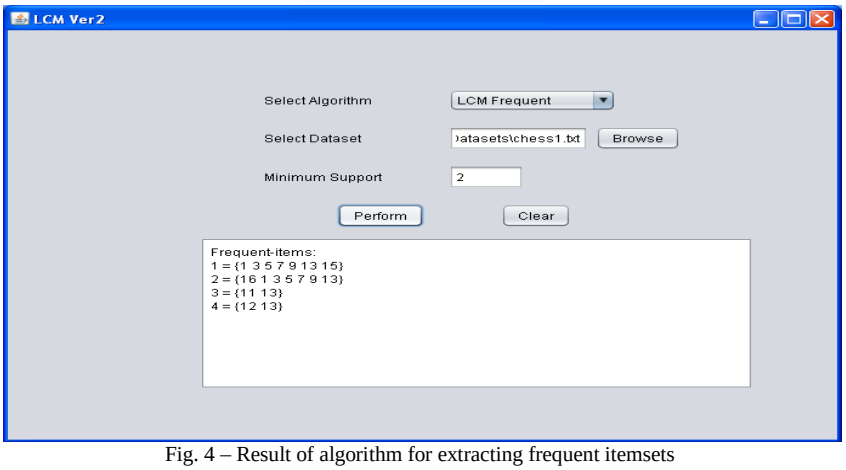

Fig. 4 – Result of algorithm for extracting frequent itemsets

As seen in figure 4, user is given interface for choosing any algorithm and dataset. When user selects dataset and chooses frequent itemsets as algorithm, the extracted itemset is presented in text area provided. Minimum support is the basis for extracting the result. Figure 5 presents the results of another algorithm that returns closed itemsets.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

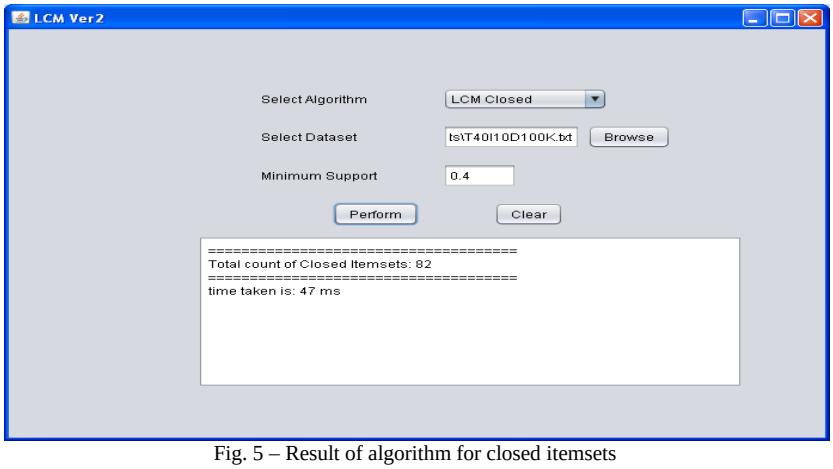

Fig. 5 – Result of algorithm for closed itemsets

As seen in figure 5, data set and algorithm are selected by user. Closed itemsets is the result of the algorithm. Figure 6 presents the result of algorithm that produces maximal itemsets. The results also reveal the time taken to discover maximal itemsets.

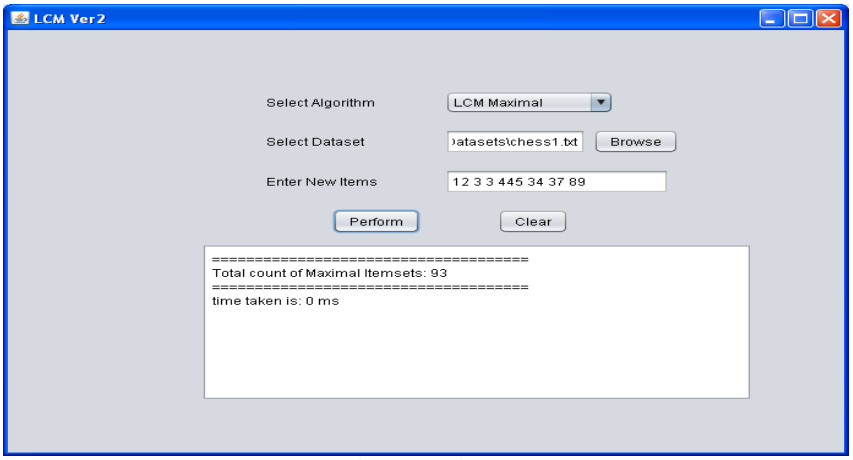

Fig. 6 – Result of algorithm for maximal itemsets

As shown in figure 6, user is given provision to choose algorithm and dataset. The data mining algorithm for extracting maximal itemsets works on the given data and produces results in text area. The algorithm also takes some additional data items as input from end user. The closed itemsets are presented.

V. EXPERIMENTAL RESULT

Various datasets are used with our prototype application in order to make certain experiments. The results thus obtained are also compared with other algorithms of this paper and the ones came prior to this paper. In all experiments the application captures minimum support from a domain expert. The results are presented as follows.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

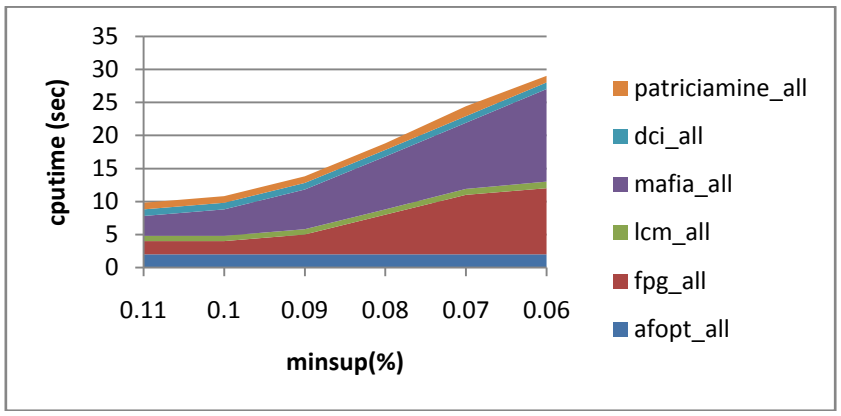

Fig 7. BMS-WebView-2-all

As shown in fig 7. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

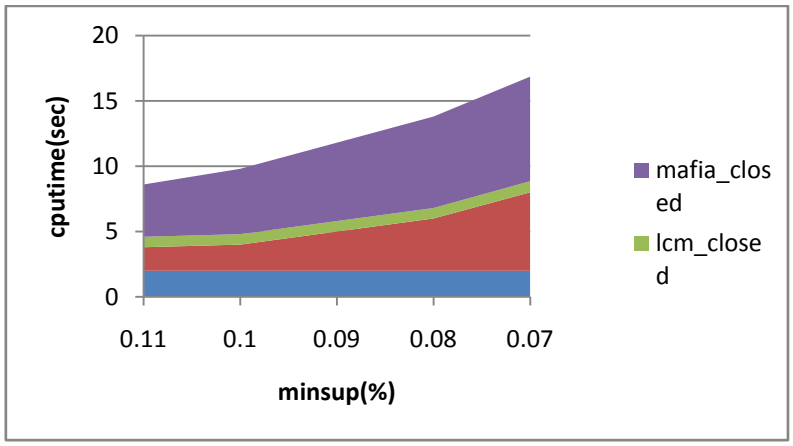

Fig 8. BMS-WebView-2-Closed

As shown in fig. 8. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

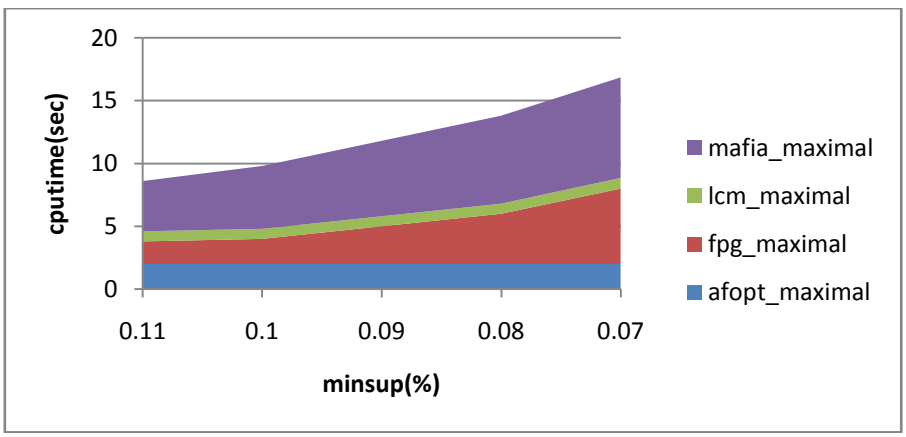

Fig 9-BMS-WebView-2-Maximal

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

As shown in fig. 9. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

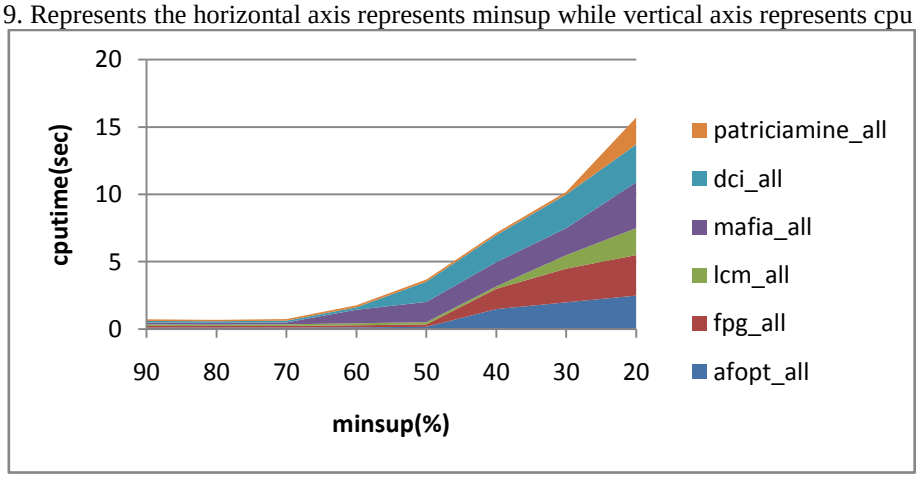

Fig 10 Chess All

As shown in fig. 10. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

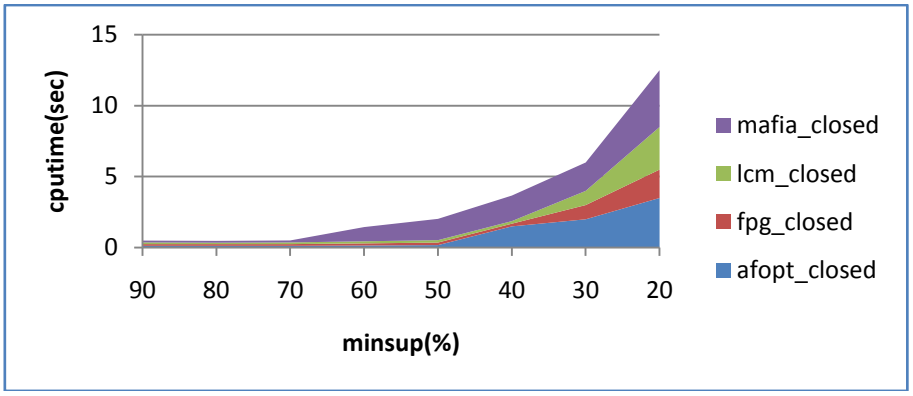

Fig 11- Chess Closed

As shown in fig.11. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

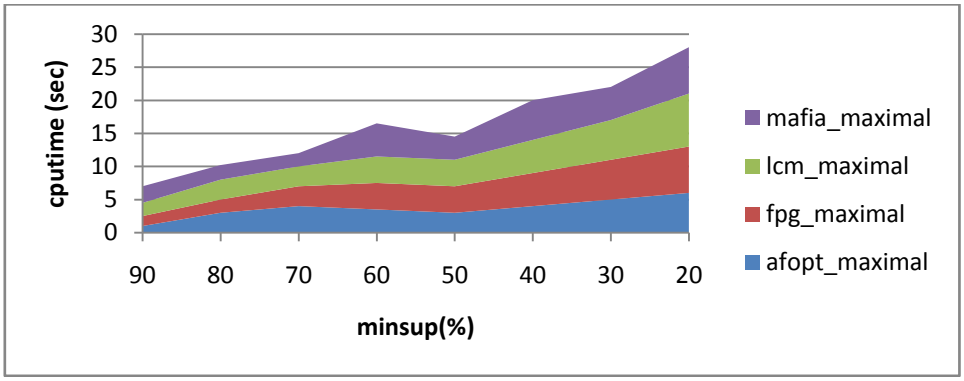

Fig 12-Chess Maximal

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

As shown in fig. 12. Represents the horizontal axis represents minsup while vertical axis represents cpu time.

VI. CONCLUSION

In this paper we studied many data mining algorithms for extracting business intelligence. We implemented algorithms that are recently proposed by Uno et al. [3] for extracting frequent itemsets, maximal itemsets and closed itemsets. These itemsets are used to make expert decisions in real time applications. The algorithms produce actionable knowledge that can be used to make effective decisions. We built a prototype application that demonstrates the effectiveness of the algorithms. The empirical results are encouraging.

REFERENCES

[1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," In Proceedings of VLDB '94, pp. 487{499, 1994.
[2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonenvand A. I. Verkamo, "Fast Discovery of Association Rules," In Advances in Knowledge Discovery and Data Mining, MIT Press, pp. 307{328,1996.
[3] Takeaki Uno, Masashi Kiyomi and Hiroki Arimura, "LCM ver. 2: E_cient Mining Algorithms for Frequent/Closed/Maximal Itemsets".
[4] E. Boros, V. Gurvich, L. Khachiyan, and K. Makino, "On the Complexity of Generating Maximal Frequent and Minimal Infrequent Sets," STACS 2002, pp. 133-141, 2002.
[5] D. Burdick, M. Calimlim, J. Gehrke, "MAFIA: A Maximal Frequent Itemset Algorithm for Transactional Databases," In Proc. ICDE 2001, pp. 443-452, 2001.
[6] D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, and T. Yiu, "MAFIA: A Performance Study of Mining Maximal Frequent Itemsets," In Proc. IEEE ICDM'03 Workshop FIMI'03, 2003. (Available as CEUR Workshop Proc. series, Vol. 90, http://ceur-ws.org/vol-90)
[7] G. Grahne and J. Zhu, "Effciently Using Pre_x-trees in Mining Frequent Itemsets," In Proc. IEEE ICDM'03 Workshop FIMI'03, 2003. (Available as CEUR Workshop Proc. series, Vol. 90, http://ceur-ws.org/vol-90)
[8] J. Han, J. Pei, Y. Yin, "Mining Frequent Patterns without Candidate Generation," SIGMOD Conference 2000, pp. 1-12, 2000
[9] R. Kohavi, C. E. Brodley, B. Frasca, L. Mason and Z. Zheng, "KDD-Cup 2000 Organizers' Report: Peeling the Onion," SIGKDD Explorations, 2(2), pp. 86-98, 2000.
[10] R. J. Bayardo Jr., "Efficiently Mining Long Patterns from Databases", In Proc. SIGMOD'98, pp. 85{93, 1998.
[11] Takeaki Uno, Tatsuya Asai, Yuzo Uchida and Hiroki Arimura, "LCM: An Efficient Algorithm for Enumerating Frequent Closed Item Sets".

BIOGRAPHY

S. Shoban Babu is working as an Associate Professor in Department of Computer Science and Engineering at Varadha Reddy Engineering College, Warangal. He had completed his Ph.D in Computer Science and Engineering. He had more than 16 years of teaching experience at various Engineering colleges in India and also at abroad.
His areas of interest are Data Mining, Web Technologies, and Big Data etc.