

An Improved Apriori Algorithm for Frequent Item Set Mining Accuracy

M.Ranjanisindu¹, K.Jenifer²

PG Scholar, Department of CSE, Coimbatore Institute of Engineering and Technology, Coimbatore, India¹

Assistant Professor, Department of CSE, Coimbatore Institute of Engineering and Technology, Coimbatore, India²

ABSTRACT: Frequent sets play an important role in many Data Mining tasks to find item patterns from transactional databases. Apriori algorithm is the typical algorithm of frequent item sets. Apriori algorithms accuracy not correct when the transaction time is considered in mining frequent item sets. This paper proposes improved algorithm of apriori by considering the transaction time in the frequent item set generation. Improved algorithm is verified and thus the results are more accurate.

KEYWORDS: Data Mining, Frequent Item set Mining, apriori, Frequent item set, Improved apriori.

I. INTRODUCTION

Frequent sets finds patterns such as association rules, correlations, sequences, episodes, classifiers and clusters. The mining of association rules is one of the most popular problems of all these. The identification of sets of items, products, symptoms and characteristics, which often occur together in the given transactional database, can be seen as one of the most basic tasks in Data Mining. The original motivation for searching frequent sets came from the need to analyse so-called supermarket transaction data, that is, to examine customer behaviour in terms of the purchased products (Agrawal et al., 1993). Frequent sets of products describe how often items are purchased together.

II. THE APRIORI ALGORITHM

The first algorithm to generate all frequent itemsets and confident association rules was the AIS algorithm by Agrawal et al., which was given together with the introduction of this mining problem. Shortly after that, the algorithm was improved and renamed Apriori by Agrawal et al., by exploiting the monotonicity property of the support of itemsets and the confidence of association rules. The same technique was independently proposed by Mannila et al. Both works were cumulated afterwards.

Itemset mining

The itemset mining phase of the Apriori algorithm is given in below Algorithm. We use the notation $X[i]$, to represent the i th item in X . The k -prefix of an itemset X is the k -itemset $\{X[1], \dots, X[k]\}$.

Algorithm Apriori – Item set mining

```
Input: D,  $\sigma$ 
Output: F(D,  $\sigma$ )
1:  $C_1 := \{\{i\} \mid i \in I\}$ 
2:  $k := 1$ 
3: while  $C_k \neq \{\}$  do
4: // Compute the supports of all candidate itemsets
5: for all transactions  $(tid, I) \in D$  do
6: for all candidate itemsets  $X \in C_k$  do
```

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 5, May 2019

```

7:  if  $X \subseteq I$  then
8:    X.support++
9:  end if
10: end for
11: end for
12: // Extract all frequent itemsets
13:  $F_k := \{X \mid X.\text{support} \geq \sigma\}$ 
14: // Generate new candidate itemsets

15: for all  $X, Y \in F_k, X[i] = Y[i]$  for  $1 \leq i \leq k - 1$ , and  $X[k] < Y[k]$  do
16:   $I = X \cup \{Y[k]\}$ 
17:  if  $\forall J \subset I, |J| = k : J \in F_k$  then
18:     $C_{k+1} := C_{k+1} \cup I$ 
19:  end if
20: end for
21: k++
22: end while

```

The algorithm performs a breadth-first search through the search space of all itemsets by iteratively generating candidate itemsets C_{k+1} of size $k + 1$, starting with $k = 0$ (line 1). An itemset is a candidate if all of its subsets are known to be frequent. More specifically, C_1 consists of all items in I , and at a certain level k , all itemsets of size $k + 1$ in $Bd-(F_k)$ are generated. This is done in two steps. First, in the join step, F_k is joined with itself. The union $X \cup Y$ of itemsets $X, Y \in F_k$ is generated if they have the same $k - 1$ -prefix (lines 20–21). In the prune step, $X \cup Y$ is only inserted into C_{k+1} if all of its k -subsets occur in F_k (lines 22–24). To count the supports of all candidate k -itemsets, the database, which retains on secondary storage in the horizontal database layout, is scanned one transaction at a time, and the supports of all candidate itemsets that are included in that transaction are incremented (lines 6–12). All itemsets that turn out to be frequent are inserted into F_k (lines 14–18). Note that in this algorithm, the set of all itemsets that were ever generated as candidate itemsets, but turned out to be infrequent, is exactly $Bd-(F)$.

Problem definition

The frequent item set is not accurate. Consider the items {apple, orange, rice}. Frequent set {apple, orange} support was high for the whole transactional database but the frequent set {apple, rice} support was high on morning compared with frequent set {apple, orange}. In our typical algorithm every time frequent itemset {apple, orange} gets high priority but in our case frequent itemset {apple, rice} will be highest priority on morning. In this paper we solve this problem and provide the accurate frequent item set.

Improved apriori algorithm

In this algorithm typical apriori algorithm is used. In that algorithm we consider the transactional time. Based on the need the transactional time will be considered. Following are the sample use cases
Frequent Item set mining based on the particular range of time period. Example: Every day 07:00AM to 11:00AM
Frequent Item set mining based on the seasons. Example: Frequent item set for the summer season.
Frequent Item set mining based on the date. Example: New year season every year.

```

Input: D,  $\sigma$ 
Output: F(D,  $\sigma$ )

1:   $C_1 := \{\{i\} \mid i \in I\}$ 
2:  k := 1
3:  while  $C_k \neq \{\}$  do
4:    // Compute the supports of all candidate itemsets

```

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 5, May 2019

```

5: for all transactions (tid, I) ∈ D do
6: if transaction time = Frequent set expected time then
7: for all candidate itemsets X ∈ Ck do
8: if X ⊆ I then
9: X.support++
10: end if
11: end for
12: end if
13: end for
14: // Extract all frequent itemsets
15: Fk := {X | X.support ≥ σ}
16: // Generate new candidate itemsets
17: for all X, Y ∈ Fk, X[i] = Y [i] for 1 ≤ i ≤ k - 1, and X[k] < Y
[k] do
18: I = X ∪ {Y [k]}
19: if ∀J ⊂ I, |J| = k : J ∈ Fk then
20: Ck+1 := Ck+1 ∪ I
21: end if
22: end for
23: k++
24: end while

```

Transaction time is considered in the frequent item set mining. If the transaction time is in the expected time range then the transaction entry is considered to perform frequent item set mining or the transaction entry is ignored. In this proposal transactional database to be modified to support the transaction time tracking. Sample transactional database shown below

Transaction Id	TransactionTime	Set of items
100	20 mar 2019 10:8:5	{beer, chips, wine}
200	20 mar 2019 13:8:5	{beer, chips}
300	21 mar 2019 9:8:5	{pizza, wine}
400	21 mar 2019 20:8:5	{chips, pizza}

III. CONCLUSION

In this paper, An improved apriori algorithm is proposed to provide the accurate frequent itemset. From this algorithm frequent itemset mined accurately based on the expected time. Transactional database also changed to store the transaction time, which will be used in the proposed algorithm.

REFERENCES

- [1] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," Knowledge and Information Systems, vol. 14, no. 1, pp. 1–37, Dec. 2007.
- [2] S. Rao, R. Gupta, "Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm", International Journal of Computer Science And Technology, pp. 489–493, Mar. 2012

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 5, May 2019

- [3] H. H. O. Nasereddin, "Stream data mining," International Journal of Web Applications, vol. 1, no. 4, pp. 183–190, 2009.
- [4] F. Crespo and R. Weber, "A methodology for dynamic data mining based on fuzzy clustering," Fuzzy Sets and Systems, vol. 150, no. 2, pp. 267–284, Mar. 2005.
- [5] R. Srikant, "Fast algorithms for mining association rules and sequential patterns," UNIVERSITY OF WISCONSIN, 1996.
- [6] J. Han, M. Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, Book, 2000.
- [7] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," AI magazine, vol. 17, no. 3, p. 37, 1996. Goswami D.N et. al. "An Algorithm for Frequent Pattern Mining Based On Apriori " (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, Pp. 942-947.
- [8] F. H. AL-Zawaidah, Y. H. Jbara, and A. L. Marwan, "An Improved Algorithm for Mining Association Rules in Large Databases," Vol. 1, No. 7, 311-316, 2011
- [9] T. C. Corporation, "Introduction to Data Mining and Knowledge Discovery", Two Crows Corporation, Book, 1999.
- [10] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in ACM SIGMOD Record, vol. 22, pp. 207–216, 1993
- [11] M. Halkidi, "Quality assessment and uncertainty handling in data mining process," in Proc, EDBT Conference, Konstanz, Germany, 2000.
- [12] DENG ZhiHong, WANG ZhongHui and JIANG JiaJian, "A new algorithm for fast mining frequent itemsets using N-lists". Key Laboratory of Machine Perception (Ministry of Education), School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China