International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2016, pp. 384-387
e-ISSN:2278-621X

A STUDY ON THE CONTRIBUTION OF APRIORI ALGORITHM IN CUSTOMER BEHAVIOR PREDICTION

Sujith Baby[1], Shamanth Rai[2], Manish Poojary[3] and Manimozhi R[4]

Abstract- Apriori algorithm is one among the most discussed algorithm among association rules. Even though there exists many algorithm for association rule, Apriori stands first because of it's efficiency in mining association rules. In large databases, mining association rule takes lot of time, but Apriori algorithm can make it little more easier with it's reduced number of data base scanning. Since the data mining methods have advantages and disadvantages it is important to find out which is the appropriate techniques to mine data bases. The objective of this paper is to know how suitable is Apriori algorithm for customer behavior prediction.

I. INTRODUCTION

To study how a customer wants to purchase certain product from a shop and how randomly they are buying the items, this is very interesting concept of study. To understand the psychological mind-set and converting it into statistical format so that it can be helpful in analyze buying behavior. These all things help an organization to improve their marketing strategies. The strategies such as how a person select between different choices, how a living environment affects his way of purchasing and how effective is a marketing campaign are analyze through different association rules.

II. ALGORITHM

A. Apriori algorithm –

All frequent itemset can be generated using Apriori Algorithm. The itemset whose support is greater than some user-specified minimum support is considered as frequent itemset. Candidate itemset is potentially frequent itemset. To make multiple passes over the database is the key idea of Apriori algorithm. An iterative approach breadth-first search is used through the search space, where k-itemsets are used to explore (k+1)-itemsets. At First, the set of frequent 1-itemsets is found.The set of this 1-itemsets contains one item, which satisfy the support threshold is denoted

[1] *AIMIT, Mangaluru, Karnataka, India*
[2] *AIMIT, Mangaluru, Karnataka, India*
[3] *AIMIT, Mangaluru, Karnataka, India*
[4] *AIMIT, Mangaluru, Karnataka, India*

by L1. We begin with a seed set of itemsets found to be large in the previous pass in each subsequent pass.This seed set is used for generating new potentially large itemsets. A new potentially large itemsets called candidate itemsets is generated using this seed set, and count the actual support for these candidateitemsets during the pass over the data. At the end of the pass, the candidate itemsets which are actually large (frequent) is determined, and they become the seed for the next pass. Therefore, first L1 is used to find L2, the set of frequent 2-itemsets, then L2 is used to find L3, and so on, until no more frequent k-itemsets can be found.Then, Aprioriproperty is employed to reduce the search space, where the Apriori property is described as —"All nonempty subsets of a large itemset must also be large" or —"If a set is not large, then its superset can't be large either".This property belongs to a special category of properties called antimonotone which means that if a set cannot pass a test, all of its supersets will fail the same test as well.

1. **Pass 1**
1. Generate the candidate itemsets in C1
2. Save the frequent itemsets in L1

2. **Pass k**
(i). Generate the candidate itemsets in Ckfrom the frequent itemsets in Lk-1
Join Lk-1p with Lk-1q, as follows:
insertntoCk
select p.item1, p.item2, . . . , p.itemk-1, q.itemk-1
from Lk-1p, Lk-1q
where, p.item1 = q.item1, . . . p.itemk-2 = q.itemk-2, p.itemk-1<q.itemk-1
• Generate all (k-1)-subsets from the candidate itemsets in Ck
• Prune all candidate itemsets from Ckwhere, some (k-1)-subset of the candidate itemset is not in the frequent itemset Lk-1
(ii). Scan the transaction database to determine the support for each candidate itemset in Ck
(iii). Save the frequent itemsets in Lk.

II. EXPERIMENT AND RESULT

The test set for this evaluation experiment chess data randomly selected from the internet. NetBeans IDE 8.1 software platform is used to perform the experiment. The PC for experiment is equipped with an core i3 2.10GHz CPU with speed and 4GB RAM in personal computer.

From the simulation of the experiment results, we can draw to the conclusion that the algorithm consumes less time and minimum database scans to generate the results.

```
Input configuration: 76 items, 3196 transactions,
minsup = 0.8%
Passing through the data to compute the frequency of 76 itemsets of size 1
[3]   (0.8882978723404256 2839)
[5]   (0.9295994993742178 2971)
[7]   (0.9624530663329162 3076)
[9]   (0.8992490613266583 2874)
[25]   (0.8948685857321652 2860)
[29]   (0.9953066332916145 3181)
[34]   (0.951188986232791 3040)
[36]   (0.9696495619524406 3099)
[40]   (0.9918648310387985 3170)
[42]   (0.8491864831038799 2714)
[44]   (0.8172715894868585 2612)
[48]   (0.9427409261576971 3013)
[52]   (0.996558197747184 3185)
[56]   (0.945244055068836 3021)
[58]   (0.9996871088861077 3195)
[60]   (0.9852941176470589 3149)
[62]   (0.9574468085106383 3060)
[64]   (0.8232165206508135 2631)
[66]   (0.945244055068836 3021)
Found 19 frequent itemsets of size 1 (with support 80.0%)
Creating itemsets of size 2 based on 19 itemsets of size 1
Created 171 unique itemsets of size 2
Passing through the data to compute the frequency of 171 itemsets of size 2
```

```
[7, 29, 34, 36, 40, 48, 58, 60, 66]   (0.8069461827284106 2579)
[29, 34, 36, 40, 48, 52, 58, 60, 66]   (0.8376095118898623 2677)
[7, 29, 36, 40, 48, 52, 58, 60, 66]   (0.8423028785982478 2692)
[36, 40, 48, 52, 56, 58, 60, 62, 66]   (0.8016270337922403 2562)
Created 78 unique itemsets of size 10
[29, 34, 36, 40, 52, 56, 58, 60, 62]   (0.8225907384230288 2629)
[5, 7, 29, 34, 40, 52, 58, 60, 66]   (0.8191489361702128 2618)
[5, 29, 34, 36, 40, 52, 56, 58, 60]   (0.8028785982478097 2566)
[7, 29, 34, 36, 40, 48, 52, 60, 66]   (0.8044430538172715 2571)
[5, 29, 34, 40, 52, 56, 58, 60, 62]   (0.8110137672090113 2592)
Passing through the data to compute the frequency of 78 itemsets of size 10
[29, 34, 36, 40, 48, 52, 58, 60, 62, 66]   (0.8031914893617021 2567)
[7, 29, 36, 40, 48, 52, 58, 60, 62, 66]   (0.8050688360450563 2573)
[7, 29, 36, 40, 48, 52, 56, 58, 60, 62]   (0.8016270337922403 2562)
[7, 29, 34, 36, 40, 48, 52, 58, 60, 66]   (0.8041301627033792 2570)
Found 4 frequent itemsets of size 10 (with support 80.0%)
Creating itemsets of size 11 based on 4 itemsets of size 10
Created 2 unique itemsets of size 11
Passing through the data to compute the frequency of 2 itemsets of size 11
Execution time is: 6.343 seconds.
Found 8227 frequents sets for support 80.0% (absolute 2557)
Done
BUILD SUCCESSFUL (total time: 7 seconds)
```

IV.CONCLUSION

A chess data set has been used, which includes 76 items in it for analysis. A set of association rules have been obtained by applying Apriori algorithm for it. During the analysis we found that Apriori algorithm is efficient and consumed less time to generate the result.All these results are collected from Intel core i3 2.10GHz CPU with speed and 4GB RAM.

REFERENCES

[1] SanjeevRao and Priyanka Gupta and Implementing Improved Algorithm Over APRIORI Data MiningAssociation Rule Algorithm" 2012.

[2] Shilpa; SunitaParashar "Performance Analysis of Apriori Algorithm with Progressive Approach for Mining Data" October 2011.

[3] Bo Wu; Defu Zhang; QihuaLan; JieminZheng "An Efficient Frequent Patterns Mining Algorithm based on Apriori Algorithm and the FP-tree Structure" Xiamen 361005, China 2008.
[4] Agrawal R; SrikantR ; Fast algorithms for mining association rules ; In Proceedings 20th International Conference on Very Large Data Bases (VLDB' 94); pp 487-499; 1994.
[5] AgrawalR ;Srikant R ; Fast algorithms for mining association rules [A]; In; Proceedings of the 20th Internationan Conference on Very Large Databases[C]; Santiago ; Chile ; 1994 487-499.
[6] Herbert A Edelstein; Introduction to Data Mining and Knowledge Discovery; 3rd Edition; pp 22-26; Oct 1999.
[7] Kumar B.S., RukmaniK.V.,"Implementation of Web Usage Mining Using APRIORI and FP Growth Algorithms", in Int. J. of Advanced Networking and Applications Vol. 01, Issue, 06, pp. 400-404, 2010.
[8] Pujari A. K.,"Data mining techniques", Universities Press(India) Private Limited, 2001.