International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 11 ǁ November 2022 ǁ PP. 287-295

Hybrid Association Mining Of In-Frequent Itemsets For Business Process Optimization.

A.M Shams Raza[*], B. Md. Tanwir Uddin Haider**, C.Anil Kumar Singh [***]

Sr. Life Member CSI
Dept of Physics &
Dept. of Computer Science

Sr. member IEEE
Dept of Comp. Sc & Engg
N.I.T. Patna

Assitant Professor
Dept. of Physics
V.K.S.University, Ara

ABSTRACT
Association mining is mainly used for identifying the association between frequent item sets in business transactions. It also reveals a combinational aspect of products and establishes a rule for obtaining more and more frequent item sets based on established association rules to predict product combos for boosting sales and enhancing the profitability of a business. The prediction of product association depends on the accuracy of the rule established, and for establishing an accurate rule maximum data set is required. Generally, for the generation of the data set, the Apriori algorithm is used which delivers a maximum collection of frequent item sets along with very few infrequent item sets in a single scan of the database. Generally, in-frequent item sets are eliminated for making association rule of mining, but for the business and organization, each item is important. So we have taken-up in-frequent item sets as a target for our research and tried to generate a sufficient collection of in-frequent item sets, which are simply ignored by the Apriori algorithm method, by implementing the Hybrid Algorithm which is an integration of the Apriori algorithm and Template Matching Algorithm. Therefore it is observed that this Hybrid Algorithm has given a provision to estimate the confidence of in-frequent items along with frequent items, while the Apriori algorithm alone is allowing to estimate confidence for frequent items only, so this is the achievement of this Hybrid Algorithm, it is justified using 2 x 2 Confusion Metrics, given in result section.

Keywords: *Association mining, Apriori algorithm, Machine Learning System, Template Matching Algorithm, Hybrid Algorithm.*

Date of Submission: 14-11-2022 Date of acceptance: 28-11-2022

I. Introduction

In the present era, data mining and knowledge discovery have emerged as the latest, fundamental research areas with several applications to science, engineering, health care, business, and education. Gagandeep and Shruti et al. [1] expressed that data mining tries to model, analyze and implement knowledge discovery processes that help the extraction of meaningful information and knowledge from unstructured data. Agarwal and Shrikant et al.[2] says that in scientific and commercial applications database size is huge where the number of tuples in a dataset may be in the range of some thousands to thousands of millions. Most business data mining objective is product base analysis and under data mining rules the association rule of data mining is mostly recommended rule for market basket analysis. The Association rule of mining is a process in which algorithms are deployed to discover frequent item sets with some kind of associative feature in a list of transactions.

The Association rule of mining is based on support and confidence parameter. Several algorithms are available for association rule of mining such as ECLAT, FP-Growth, and OPUS Search in which we have taken Apriori algorithms of data mining and from the machine learning technique, we have adopted Template Matching Algorithm for building proposed Hybrid Algorithm. Pratiksha and Tina et al.[3] summarized that Apriori algorithms help to find out frequent item sets from database transactions and prepare a final frequent item sets list to build the best rule of mining based on minimum support and minimum confidence target values. Also, this algorithm generates some infrequent item set along with frequent item set.

The Apriori algorithm has some limitations. Chen and Han et al.[4] says it does not generate multiple items set in large volumes having both frequent and infrequent, as it restricts users to choose only one type of item set having a maximum frequency. Liu, Hsu, and Ma et al.[5] have jointly narrated that the Apriori algorithm deploys an iterative approach or level-wise search where k-frequent item sets are used to find k+1 item sets, so runtime increases exponentially in case of different items occur in candidate generation. Both limitations are resolved by using the Hybrid Algorithm as it generates a sufficient number of (earlier declared in-frequent items) as frequent item sets and also runtime increase linearly depending on items and transactions.

Han, Pei, and Yin et al. [6] have jointly concluded that Template Matching Algorithm is a process of machine learning system to match sample-based patterns into the database and mark such data into the database for collection as a classified group of data, based on the provided sample. Under this process, labeled training data is applied to train Template Matching Algorithm. The sample label data set is attached to the input value which ultimately produced a pattern-based output data set.

In a data source there exist lots of such tuples which are simply ignored by Apriori as in-frequent itemset. This happens due to the nearest tuple distance, the items which are existing with a closer tuple distance are taken up as frequent itemset by algorithms like Apriori, but those tuples which are scattered at a far distance are ignored. From the business point of view, every item is important. So in this paper, we have set our point of research to find out a sufficient number of tuples of in-frequent items (which are ignored by Apriori) from the data source so that association mining rules can be formed for such items also. This we have achieved using the proposed Hybrid Algorithm, and helps the business to save from losses being caused due to ignorance of such infrequent items ultimately opening the opportunity for profit. Our proposed Hybrid Algorithm is divided into two parts. In the first part, we applied the Apriori algorithm and obtained sets of in-frequent items which are prepared as a sample model for the next part, and this sample model is applied to Template Matching Algorithm along with the input data source. Finally, we have got a separate large data set of in-frequent items making it feasible to association rule specification for such items which are simply ignored by Apriori as infeasible itemsets for rule specification.

The remaining part of this paper consists as follows. In section 2 Literature review of both Algorithms, Apriori and Template Matching Algorithm is presented. In section 3 proposed integrated algorithm is elaborated and implementation and experimental results are presented, and finally in section 4 conclusion & future scope are presented.

II. Literatures Survey

In this paper, to fulfill the problem definition we have exclusively investigated six papers in which we have summarized three papers for Apriori Algorithm and three papers related to Template Matching Algorithm in detail. This section has two subsections. Subsection 2.1 presents research papers related to Apriori Algorithm whereas subsection 2.2 presents research papers related to Template Matching Algorithm respectively.

2.1 Research Papers related to Apriori Algorithm

In the case of business transaction analysis mostly Apriori algorithms are used especially with the concept of market basket analysis. In the majority of cases of research on the Apriori algorithm, it is found that it has some limitations and it needs improvement. Haoyu et al. [7] have accepted that this is a very popular algorithm but at the same time, it has some disadvantages or limitations which are required to be addressed with some innovative improvements for its optimizations. Out of multiple algorithms for association rule-based data mining, such as ECLAT, FP-Growth, and OPUS Search, the Apriori is one of the most frequently used algorithms. Al-Maolegi and Arkok et al. [8] found that it is based on a process of scanning the data source repeatedly which ultimately counted as a waste of computing resources including CPU time, and memory wastages, and consumes too much effort in sample data modeling. Therefore several researchers have suggested many improvements for the Apriori algorithm, especially for time-saving and output optimization with minimum resource waste. Another aspect of the Apriori algorithms studied by Cheng and Xiong et al. [9] is its elimination of records as the scanning process grows in the data source. So it is suggested that in place of classical Apriori there should be a modified version as incremental Apriori which will manage larger data sets without elimination of growing tuples,

2.2 Research Papers related to Template Matching Algorithm

Julie and Dhome et al.[10] summaries that Template Matching Algorithm's functioning is based on sample models used for training the procedures which in turn act in real-time and provide all matching objects as output for the required event. Mostly it is used for image-based matching but in some cases, text patterns are also delivered based on text models. Fouda et al. [11] say that one important area in machine learning is pattern recognition. There are several algorithms are available for pattern recognition. in which one of the simplest algorithms is Template Matching Algorithm. This algorithm can be used for searching objects with sample matching or pattern matching. This may be applied for 1-D, 2-D, or 3-D image pattern matching as well as it may be applied for text-based tuples matching from databases. This ensures the accuracy of output with zero error. Batta et al. [12] explain Machine Learning Systems as the application of devices and software to train the system with some sample model of data to make them able to identify each data item without further manual interventions. It means if we provide a set of cat images into an image-based software system that may accept each image by graphical input devices and store each image by disintegrating its attributes, then remake the image and compare it with the source and deliver a message that this is a cat or not. Similar to getting trained with the model data the software-based system is applied in real life, such as in a zoo to count the number of cats available, then using a drone camera this system can give an accurate figure of cats by identifying each cat in the zoo, eliminating other animals comes into the camera during scanning. This software may be developed using any one machine learning algorithm from the various available algorithms, such as the Template Matching Algorithm,

After reviewing the above papers on the association rule of mining we have observed that in almost all cases infrequent itemset is simply ignored. Most people have tried to improve the performance of the Apriori algorithm, but they all remain focused on frequent item sets only, in this paper we have focused on in-frequent item sets. And after reviewing papers on machine learning algorithms it is found that one of the algorithms Template Matching has great potential for extracting exactly matching data objects from the data sources and we have taken this opportunity of this algorithm for the creation of our Hybrid Algorithm, which has finally given us success to obtain a sufficient number of tuples of in-frequent item sets from the same data source.

III. Proposed Hybrid Algorithm

We have prepared this Hybrid Algorithm into two phases, in the first phase, the classical Apriori algorithm is used for obtaining the initial data set which includes both frequent item set and infrequent item set. From these initial data sets, we have selected an in-frequent item set for our experimentation in the next phase. Then in the next phase, we applied Template Matching Algorithms with the data model prepared from the initial data set of in-frequent items obtained in the first phase, and finally analyzed the collected result of the second phase.

Phase I

In this phase, we have applied the Apriori algorithm to the transaction database and generated data sets with limited tuples including frequent and infrequent item lists, and stored them in Table 1. Using the available data set in Table 1 received from the Apriori algorithm execution, we have prepared sample data to set A of three infrequent items as { Cookies, Cake & Biscuits}, which are presented in the implementation section.

After verifying the concept and testing algorithms we have applied algorithms using codes in python and a business transactional database from Kaggle on a high-end system on the windows platform.

Data source: Supermart Grocery Sales -
Retail Analytics Dataset (Kaggle.com)
File type: Excel
Columns: 11
Tuples (Records): 9994 rows
https://www.kaggle.com/datasets/mohamedharris/supermart-grocery-sales-retail-analytics-dataset?select=Superma

System configuration and Platform:
Dell PC with 4 GB RAM, i5 7 generation
processor, Windows 10
IDE: Python (Anaconda3)

Snapshot of Data Source:

Order ID	Customer Name	Category	Sub Category	City
OD1	Harish	Oil & Masala	Masalas	Vellore
OD2	Sudha	Beverages	Health Drinks	Krishnagiri
OD3	Hussain	Food Grains	Atta & Flour	Perambalur
OD4	Jackson	Fruits & Veggies	Fresh Vegetables	Dharmapuri
OD5	Ridhesh	Food Grains	Organic Staples	Ooty
OD6	Adavan	Food Grains	Organic Staples	Dharmapuri
OD7	Jonas	Fruits & Veggies	Fresh Vegetables	Trichy

OD8	Hafiz	Fruits & Veggies	Fresh Fruits	Ramanadhapuram
OD9	Hafiz	Bakery	Biscuits	Tirunelveli
OD10	Krithika	Bakery	Cakes	Chennai
OD11	Ganesh	Snacks	Chocolates	Karur
OD12	Yadav	Eggs, Meat & Fish	Eggs	Namakkal
OD13	Sharon	Snacks	Cookies	Dindigul
OD14	Peer	Fruits & Veggies	Fresh Vegetables	Kanyakumari
OD15	Sundar	Eggs, Meat & Fish	Chicken	Kanyakumari
OD16	Ramesh	Oil & Masala	Edible Oil & Ghee	Krishnagiri
OD17	Alan	Bakery	Cakes	Dharmapuri
OD18	Arutra	Beverages	Health Drinks	Bodi
OD19	Haseena	Eggs, Meat & Fish	Mutton	Tenkasi
OD20	Verma	Beverages	Soft Drinks	Kanyakumari
OD21	Hafiz	Beverages	Health Drinks	Vellore
OD22	Alan	Food Grains	Dals & Pulses	Karur
OD23	Haseena	Beverages	Soft Drinks	Krishnagiri
OD24	Alan	Fruits & Veggies	Organic Vegetables	Tenkasi

Figure-1. Data source view of the structure with data values

Implementation of Phase-I of the proposed Hybrid Algorithm

In the first Phase I, our target is to generate Table-1, using the Apriori algorithm, with item sets having frequent items and infrequent items, for building the model data sets A based on the in-frequent item set to explore the same data source using the Template Matching Algorithm in Phase II

Steps in Apriori Algorithm;

This algorithm is processed in sequential steps and searches frequent item sets in the database. This algorithm uses the join and elimination steps iteratively till the approx frequent item set is obtained. The minimum support threshold is specified in the proposal or assumed by the user.
Step 1: In the first cycle of the algorithm, every item is taken as a 1-itemsets candidate. And it will count the occurrences of every item.

Step 2: Consider there is minimum support is 2. Set of 1 – item sets whose occurrence is matching the minimum support are identified. And the data set whose count is found more than or equal to minimum support is selected for the next cycle others are eliminated.

Step 3: After Step 2, 2-itemset frequent items with minimum support are searched. To do this the join process is involved in this step which generates a 2-itemset by making a group of 2-itemset by combining items with itself.

Step 4: Here 2-itemset data sets are further eliminated using minimum support value which results to form a table with a 2-itemset based on minimum support value.

Step 3 to Step 4 may be repeated to generate data sets having 3-itemset, 4-itemset, etc.

Data Extraction and organization
Step A
In the first step of data extraction, the item list is prepared from the data source, by transposing row data of transactions in a column for the purchases made by the same customer and stored in Table 1 as an item list with unique transaction IDs.

Table-1: Generated from the data source by transposing tuples into columns for the purchases made by the same customers

Tran ID	Item List
T1	Chocolates, Cookies, Noodles
T2	Chocolates, Cookies, Cake, Noodles
T3	Chocolates, Noodles, Bread & Buns
T4	Chocolates, Noodles, Biscuits
T5	Chocolates, Noodles, Bread & Buns
T6	Chocolates, Noodles, Bread & Buns
T7	Chocolates, Cookies, Noodles
T8	Chocolates, Cookies, Noodles
T9	Chocolates, Noodles,Cake,Bread & Buns
T10	Chocolates, Noodles, Biscuits
T11	Chocolates, Cookies, Noodles
T12	Chocolates, Noodles, Bread & Buns
T13	Chocolates, Noodles, Bread & Buns
T14	Chocolates, Noodles, Cake
T15	Chocolates, Cookies, Noodles
T16	Bread & Buns, Cake, Noodles
T17	Cake,Biscuits,Chocolates,Noodles
T18	Bread & Buns, Cake, Noodles
T19	Bread & Buns, Biscuits, Chocolates
T20	Bread & Buns, Cake, Biscuits
T21	Bread & Buns, Cake, Noodles
T22	Cake,Biscuits,Chocolates
T23	Bread & Buns, Cake, Biscuits
T24	Breads & Buns,Biscuits,Noodles
T25	Bread & Buns, Cake, Biscuits
T26	Bread & Buns, Biscuits, Chocolates
T27	Bread & Buns, Cake, Noodles
T28	Bread & Buns, Biscuits
T29	Bread & Buns, Cake, Biscuits
T30	Bread & Buns, Biscuits

We have enforced Threshold support of 50% and Confidence of 60% for the Implementation of the Apriori algorithm.

So the estimated minimum
support >= 50% for above data set
is 0.50 * 30 = 15

Step B

In this step, we have counted the frequency of each item from Table 1 and stored it in Table 2

Table-2: Frequency of each item generated in Table 1

Sl.no.	Item	Frequency / support
1	Chocolates	18
2	Cookies	6
3	Noodles	20
4	Cake	12
5	Biscuits	12
6	Bread & Buns	20

Step C

In this step of pruning the items having a minimum support value, >=15 from Table-2 are stored in Table-3

Table-3: Items kept having minimum support >=50% from Table-2

Sl.no.	Item	Frequency /support
1	Chocolates	18
3	Noodles	20
6	Bread & Buns	20

Therefore as per Apriori, Item sl. no. 1,3,6 is frequent itemsets in Table-2 {Chocolates,Noodles, Bread & Buns} as these are having minimum support value >=50%. But Item on sl.no. 2,4,5 do not have minimum support value in Table-2, so these items (Cookies, Cake & Biscuits) are eliminated as in-frequent items in Table-3. Now association rules can be made using the above frequent itemsets taking the following subsets, based on support & confidence;
Non empty subsets = {Chocolates, Noodles}, {Chocolates, Bread& Buns}, {Noodles, Bread & Buns},{Chocolates},{Noodles},{Bread & Buns}

Here our task is not to generate a rule for the above frequent itemsets, but we have to prove that rules can be generated for in-frequent items also which we have done in the next phase. The job of Phase-I of the Hybrid Algorithm is Completed as we have got in-frequent itemsets (Cookies, Cake & Biscuits) and these are used for making model data sets A in the next phase.

Phase II

Implementation of Phase II of the Proposed Hybrid Algorithm

In this second phase, we have implemented the Template Matching Algorithm using model data set A (Cookies, Cake & Biscuits) received in phase I. After execution of this algorithm, we have generated a different data list in Table 4, which contains multiple tuples of in-frequent item sets obtained as a result.

The steps of the Template Matching Algorithm are listed below;

Step A

Data Set A = { Cookies, Cake, Biscuits }

Steps re-define in Template Matching Algorithm for Structured data

Step 1 Select the source database and create an output table with a similar structure to Table-4.

```
        Step 2   Accept the sample data set as
                 a model template

        Step 3  Start scanning the source
                 database by reading tuple

Step4  Matching the itemset in the
          tuple using the model template
         data set

Step 5  If matched fully or partially
          with the model template data
          set then write the tuple in
          Table-4.

Step 6 Skip to the next tuple

Step 7 If not EOF() then goto Step 4
          Else Stop
```

Executing the above Template Matching Algorithm, on the data source, using sample data set A, we have generated a data list of the in-frequent item as Table 4, having tuples of pair of 3-item sets. These itemsets are now presented as frequent itemsets, which are obtained as in-frequent itemsets in Phase I.

Table 4: Generated from the data source by transposing tuples into columns for the purchases made by the same customers

Tran ID	Item List
T1	Cookies, Cakes, Biscuits
T2	Cookies, Cakes, Biscuits
T3	Cookies, Cakes, Biscuits
T4	Cookies, Cakes, Biscuits
T5	Cookies, Cakes, Biscuits
T6	Cookies, Cakes, Biscuits
T7	Cookies, Cakes, Biscuits
T8	Cookies, Cakes, Biscuits
T9	Cookies, Cakes, Biscuits
T10	Cookies, Cakes, Biscuits
T11	Cookies, Cakes, Biscuits
T12	Cookies, Cakes, Biscuits
T13	Cookies, Cakes, Biscuits
T14	Cookies, Cakes, Biscuits
T15	Cookies, Cakes, Biscuits
T16	Cookies, Cakes, Biscuits
T17	Cookies, Cakes, Biscuits
T18	Cookies, Cakes, Biscuits
T19	Cookies, Cakes, Biscuits
T20	Cookies, Cakes, Biscuits

The above Table-4 is prepared from 60 tuples by merging items into 20 tuples. This table is generated using the Template Matching Algorithm based on a sample data set A{ Cookies, Cake, Biscuits }, and it derived all matching items from the huge data source, so it contains only frequent items.

From the above table, Table-4 frequent itemsets can derive as { Cookies, Cakes, Biscuits} and nonempty binary subsets for association rulemaking may be as under;

{ Cookies,Cakes},{ Cookies,Biscuits},{ Cakes,Biscuits}

Table-5: Comparing the result using 2 x 2 Confusion metrics

Actual Data Sets	**Delivered frequency by Apriori in Phase I**	**Delivered frequency by Hybrid Algorithm in Phase II**
Cookies	6	20
Biscuits	12	20
Cakes	12	20

In Table 5 the metric is clearly showing that the frequency generated by Apriori is very less while the frequency generated by Template Matching is higher for the same itemsets.

Therefore it is justified that the itemsets which are simply eliminated by Apriori as in-frequent itemsets, can be also used for associative mining rule generation if we scan the whole data source using Machine Learning, Template Matching Algorithm. It is clear that now using the above item sets association rules can be specified as these itemsets become fully frequent item sets. If we estimate support of these itemsets will be 100% for each item accordingly it will satisfy a higher percentage of confidence.

IV. Conclusion and future scope

Normally people decide confidence from the sample tuples delivered by Apriori for frequent itemsets only and in-frequent itemsets are simply eliminated due to which business could not focus on infrequent items ultimately losing profit on such items. That is why we have designed this hybrid algorithm and generated all possible tuples in different data lists shown in Table 4, from the entire data source and finally made available data set for estimation of the confidence for in-frequent items which is turned into frequent item sets with 100% support for each item.

As in the result section above Hybrid Algorithm, Implementation had shown that by using this algorithm sufficient number of a 3-pairs itemset is generated for infrequent itemsets of Table 2. For the business domain in the present era, every aspect of resource-saving and profit enhancement is important, so even with the minute and negligible infrequent itemset they can make important business planning for the product sale enhancement. Therefore as a result of our research, we may claim that the business people will be making product promos for frequent items as per rules designed by the Apriori algorithm, at the same time with this Hybrid Algorithm they may prepare product promos for infrequent items equally which was undiscovered earlier. Through this process, they will convert the losses of in-frequent items into profit.

We have worked on this Hybrid algorithm with business data and compared it with the Apriori algorithm only, this may be further explored with some other domains such as medical systems, GIS, Climate Control, and Flight Navigation & Control systems. Also, other Association rule-based algorithms may use such ECLAT and FP-GROWTH algorithms.

References

[1]. Gagandeep Kaur, Shruti Aggarwal(2013), Performance Analysis of Association Rule Mining Algorithms, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, August 2013, ISSN: 2277 128X

[2]. Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules. In Proc.of the Int. Conf. on Very Large Databases, pages 487–499, SanTiago, Chile.

[3]. Pratiksha Shendge, Tina Gupta(2013), Comparative Study of Apriori & FPGrowth Algorithms, Indian journal of research, Volume 2, Issue 3, March 2013

[4]. Ming-Syan Chen, Jiawei Han, P.S.Yu, Data mining: an overview from a database perspective, IEEE Transactions on Knowledge and Data Engineering, Volume:8, Issue: 6 ISSN: 1041-4347, 866 - 883

[5]. Liu, B., Hsu,W., and Ma, Y. (1998). Integrating classification and association rule mining. Knowledge Discovery and Data Mining, pages 80–86.

[6]. Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. In Proc. of the Int. Conf. on Management of Data, SIGMOD, pages 1–12. ACM.

[7]. Haoyu Xie(2021), Research and Case Analysis of Apriori Algorithm Based on Mining Frequent Item-Sets. January 2021,Open Journal of Social Sciences 09(04):458-468

[8]. Mohammed Al-Maolegi, Bassam Arkok(2014), An Improved Apriori Algorithm For Association Rules, March 2014, International Journal on Natural Language Computing 3(1)

[9]. Yu Cheng, Ying Xiong(2010) : Research and Improvement of Apriori Algorithm for Association Rules, 2010 2nd International Workshop on Intelligent Systems and Applications IEEE Xplore: Accession Number: 11357493,
DOI: 10.1109/IWISA.2010.5473473

[10]. Frédéric Jurie, Michel Dhome.(2001) A simple and efficient template matching algorithm. International Conference on Computer Vision (ICCV '01), Jul 2001, Vancouver, Canada. pp.544–549, ff10.1109/ICCV.2001.937673ff. ffinria-00548281f

[11]. Fouda, Y.M. (2015) A Robust Template Matching Algorithm Based on Reducing Dimensions. Journal of Signal and Information Processing, 6, 109-122. http://dx.doi.org/10.4236/jsip.2015.62011

[12]. Batta Mahesh (2020), Machine Learning Algorithms - A Review, International Journal of Science and Research (IJSR), Volume 9 Issue 1, January 2020 ISSN: 2319-7064 ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426