Received 7 January 2024, accepted 23 January 2024, date of publication 2 February 2024, date of current version 14 February 2024.
Digital Object Identifier 10.1109/ACCESS.2024.3361667

METHODS

SDFP-Growth Algorithm as a Novelty of Association Rule Mining Optimization

BOBY SISWANTO [1], HARYONO SOEPARNO [1], NESTI FRONIKA SIANIPAR [2,3], AND WIDODO BUDIHARTO [4]
[1]Computer Science Department, BINUS Graduate Program-Doctor of Computer Science, Bina Nusantara University, South Jakarta 11480, Indonesia
[2]Biotechnology Department, Faculty of Engineering, Bina Nusantara University, South Jakarta 11480, Indonesia
[3]Food Biotechnology Research Center, Bina Nusantara University, South Jakarta 11480, Indonesia
[4]Computer Science Department, School of Computer Science, Bina Nusantara University, South Jakarta 11480, Indonesia
Corresponding author: Boby Siswanto (boby.siswanto@binus.ac.id)

ABSTRACT An essential element of association rules is the strong confidence values that depend on the support value threshold, which determines the optimum number of datasets. The existing method for determining the support value threshold is carried out manually by trial and error; the user determines a support value such as 10%, 30%, or 60% according to their instincts. If the support value threshold is inappropriate, it produces useless frequent patterns, overburdens computer resources, and wastes time. The formula for predicting the maximum count of frequent patterns was $2^n - 1$, where n is the number of distinct items in the dataset. This paper proposes a new SDFP-growth algorithm that does not require manual determination of the support threshold value. The SDFP-growth algorithm will perform dimensionality reduction on the original dataset that will generate level 1 and level 2 smaller datasets, thus automatically producing a dataset with an optimum amount of data with a minimum support value threshold. The proposed formula for predicting the maximum number of frequent patterns will become $2^{|A|} - 1$, which is $|A|$ will always be smaller than n. Experiments were performed on five various datasets, which reduced the number of data dimensions by more than 3% on the Level 1 dataset and more than 69% on the Level 2 dataset by maintaining the confidence value of the strong rules. In the execution time evaluated, we found an optimization of more than 2% on the level 1 dataset and more than 94% on the level 2 dataset.

INDEX TERMS Association rule mining, SDFP-growth algorithm, dimensionality reduction, optimization, FP-tree pruning.

I. INTRODUCTION

Association rule mining (ARM) is a method that provides recommendations for strategic decision makers in an institution [1]. Association rule mining produces a set of association rules in the form of interrelationships between variables in a dataset domain. Association rule mining is concerned with the lower limit value of the support value and confidence value, or what is known as the minimum support threshold [2]. The output results from the process of association rule mining in the form of rules resulting from data processing, used by managerial roles to make decisions; the considered rules are those rules with high support and confidence values that exceed the threshold value [3].

Formerly, the user manually determined the minimum support value threshold through trial and error [4]. This is not easy; if the determination of the minimum support value is too low, it will result in a large number of rules being generated even though the items involved are not too important to be considered. Conversely, if the determination of the minimum support value is too high, many items are not considered, even though it is possible that these items are important [5].

Association rule mining depends on the dataset characteristics. The dataset can be analogous to a set of data in a specific domain. In mathematical theory, several techniques exist for associating a set with other sets, including the theory of slices, unions, and differences [6]. Set theory provides the

The associate editor coordinating the review of this manuscript and approving it for publication was Abdullah Iliyasu.

possibility of a technical association rule mining process for datasets where the dataset used does not need to be fully or partially processed. The dataset used in part still represents the entire existing dataset where the results from association rule mining are nearly identical [7].

The *Dataset (D)* consists of *records (r),* where each record consists of several *items (I)* [8]. The dataset can be modeled as $D_i = \{r_i \mid r_i$ *a collection of transaction records*$\}$ or $D = \{r_1, r_2, r_3, \ldots, r_n\}$. The transaction records consist of variables that can be written as $r_i = \{v_{ik} \mid v_{ik}$ *is the value of the variable)* or $r = \{v_{11}, v_{12}, v_{13}, \ldots vn$. The dataset's items may contain a variety of data types, such as strings, numbers, or Booleans. Variables and datasets are transitively dependent on one another; this can be expressed as $v_{ik} \subseteq r_i$ and $r_i \subseteq D_i$ such that $r_i \subseteq D_i$. The dataset was expressed as $\{r_{vx} \,.\, r_{vy}\}$ or a spreadsheet with two dimensions, x and y [9].

Optimization is an activity that obtains the best results based on a predetermined condition on a research object that can be implemented in mathematical theory closely related to computer science. In computer science, optimization is related to improving the performance of an algorithm, where the results obtained are as minimal as possible according to certain conditions or requirements. Optimization can be divided into two types: constrained and unconstrained. Constrained optimization is an optimization technique that considers the limit value, which is the reference target [10].

The existing optimization applied to association rule mining creates a new algorithm variant whose output results are the same as those of existing techniques or algorithms. For example, the FP-Growth algorithm [11] optimizes the existing previous algorithm, namely the Apriori algorithm, which produces the same output with a more efficient computing process. The type of optimization applied to association rule mining is constrained optimization because the desired target must be within the support and confidence value limits [12].

Another existing optimization of association rule mining is the TKIFI miner algorithm proposed by Rehman et al. in 2022 [13]. This algorithm resulted from advanced research on the top-K most frequent pattern mining algorithm, which generates large itemset candidates. The TKIFI mining algorithm implements the concept of depth-first search on the top-K identical frequent pattern mining. It has been proven that TKIFI's miner algorithm can produce optimal rules on datasets with slight attribute variations. The weakness of this method is the high computational resources required for obtaining dense data sets.

Ahmad et al. proposed a measure of attractiveness (measure g) for 2021 [5]. This research is motivated by the lack of definite standard to determine the optimal minimum support, which can potentially eliminate important rules from the ARM. The proposed method yields better results than classification techniques and can produce optimal rules. The weakness of this study is that it does not calculate and consider the important lift ratio values in association rule mining.

Iqbal et al. proposed Top-k frequent itemsets mining (TKFIM) in 2021 [14]. This method is motivated by the need to determine the optimal threshold value for the existing algorithm. Determining the threshold value is very important in producing an optimal frequent itemset; however, it is not easy for those who do not know the characteristics of the dataset. Top-k frequent itemsets (TKFIM) mining is a proposed algorithm that uses class equivalence combined with set theory concepts. Based on this study, TKFIM has advantages in terms of execution and performance. A weakness of this method is that it requires a large amount of memory during the first scan.

Hikmawati et al. proposed an adaptive support model for 2021 [15]. The background of this research is that sometimes the user incorrectly determines the support threshold value such that the rules generated by the ARM are not optimal. The value of the current support is determined randomly or by trial and error, which results in enormous memory consumption and considerable time. An adaptive support model is introduced to automatically determine the minimum support threshold value by calculating the average summary comparison with the number of transactions. In the calculation process, the utility is determined by multiplying the support value of each item with the specified criteria. This research is functionally good but has a weakness: it processes the entire dataset by brute force or exhausting searches [16], [17].

Based on a literature review of ARM that ignores support threshold optimization, two general structures are found in many ARM variants: candidate generation and FP-Tree. Both of these structures have their respective advantages, but both still have weaknesses, namely problems with data characteristics [18], [19], or one that implements a genetic algorithm [20]. Table 1 lists the common issues found in existing ARM.

Based on the data in Table 1, these problems are related to the computing process, data characteristics, and memory consumption. One possible solution to this problem is to increase the compactness of the data structure. A more concise data dimension will provide various advantages, including speeding up the computation process and reducing the use of computer resources, provided that the results obtained remain valid. One technique that can be used is dimensional reduction [21], [22], which occurs at the feature selection stage [23].

The motivation of this research is to implement a new dimension reduction method to solve the problems stated in Table 1 and eliminate subjectivity in determining the support value thresholds in the association rule mining domain. The main contribution of the research results is the proposal of an algorithm that can produce a Pruned Tree based on the FP-Tree dataset structure.

The composition of the following chapters is Section II, which discusses the association rule mining concept; Section III discusses the set theory; Section IV discusses

TABLE 1. Problem finding on association rule mining without support threshold method Based On Systematic Literature Review.

ARM's Structure Types	Problems
Candidate Generation	High computation resource for dense datasets.
	Repeat full dataset scans.
	Takes a lot of time due to re-evaluating each rule.
	High memory Consumption.
	Only focus on trivial rules.
FP-Tree	Consuming a lot of memory at the time of the first scan.
	Less optimal on dense datasets.
	Higher memory usage.

the research and methodology used; Section V discusses the research and discussion; and Section VI concludes the study.

II. ASSOCIATION RULE MINING

The basic principle of association rule mining is to determine strong rules based on support and confidence values. The support value is the number that indicates how frequently an item is found in the dataset against the number of transactions, as denoted by equations 1 and 2. Equation 1 shows how frequently one item is against the number of transactions, and Equation 2 shows how frequently a combination of two items is found together with the number of transactions [24].

$$Sup\ (X) \ = (\Sigma X)/\Sigma T. \tag{1}$$

$$Sup\ (X, Y) = \Sigma(X, Y)/\Sigma T. \tag{2}$$

Association rule mining (ARM) is a technique used to find relationships between an item and other items in a dataset [25], [26]. Initially, association rule mining was used in market basket analysis (MBA) techniques, which functioned to analyze consumer buying patterns in a supermarket [27]; for example, if a consumer buys bread, he usually buys milk. Currently, MBA is used in other sectors, such as the health sector [28], [29] and socio-economics [30].

The result of association rules mining is a set of rules that determine the confidence value of an item against other items that appear together. A high confidence value indicates that the rule is strong and considered to be used as a managerial decision. Equation 3 presents the formula used to determine confidence value. A rule will have a pattern $X \Rightarrow Y$ where X is called the antecedent or Left-Hand Side (LHS), and Y is called the consequent or Right-Hand Side (RHS). The antecedent and consequent will consist of an item or several combinations of items where there is no intersection between the consequent and antecedent ($X, Y \in I$ and $X \cap Y = \emptyset$) [31].

$$Conf\,(X \Rightarrow Y) = (Sup(X, Y))/(Sup(X)). \tag{3}$$

The lift ratio is another metric considered in association rule mining. The lift ratio was used to validate the confidence values. The rule with a high confidence value still needs to be investigated using the lift ratio values. A rule with high confidence and a lift ratio equal to or greater than one is considered valid, but a rule with a lift ratio less than one, even with a high confidence value, cannot be considered valid. Equation 4 shows the formula for calculating the lift ratio [32].

$$Lift\ Ratio\ (X \Rightarrow\ Y) = Conf\ (X \Rightarrow\ Y)/(Sup\ (Y)). \tag{4}$$

Before the rules are formed, association rule mining will changes the dataset to a frequent pattern. A frequent pattern is a collection of items often found in a dataset's transaction records. The formation of frequent patterns was obtained through a frequent pattern generation process using an itemset generation algorithm. Examples of itemset generation algorithms include the Apriori and FP-Growth algorithms [33].

The FP-Growth algorithm is an implementation algorithm for association rule mining in addition to the Apriori algorithm for finding association rules [11], [34]. In contrast to the Apriori algorithm [3], the FP-Growth algorithm does not need to produce candidate itemsets. Another difference between the FP-Growth and Apriori algorithm is that the FP-Growth algorithm forms an FP-Tree, whereas the Apriori algorithm forms candidate itemsets. An FP-Tree is a logical tree structure that describes the relationship between items in a dataset. The FP tree is formed in the computer's main memory, therefore, there is no need to scan the dataset repeatedly as in the Apriori algorithm. The Apriori algorithm performs repeated readings on the dataset, resulting in a very large set of candidate items, and requires considerable computational processing. FP-Tree produces frequent patterns that are then formed into association rules.

Forming an FP-Tree on a large dataset will be very burdensome for computer performance because it requires a large main memory allocation. One technique to overcome the problem of allocating large memory is to utilize a Database Management System (DBMS), which uses tables as a container to form an FP-Tree; this is known as the EFP (Expand Frequent Pattern) algorithm [35]. The FP-Tree table created in the database uses an adjacency table structure where there is a connection between parent data and child data [36].

The FP-Growth algorithm generates association rules based on frequent patterns from a dataset. The formation of frequent patterns goes through several stages: (1) sorting items in descending order of frequency, (2) forming FP-Tree, (3) forming a Conditional Pattern Base, and (4) forming frequent patterns [37].

III. SET THEORY

Set theory is a part of mathematics that models a collection of items into certain groups [6]. Set theory groups form universal or universal sets (U). Groups of items can be modeled using set theory as intersections, combinations, differences, or subsets [38], [39].

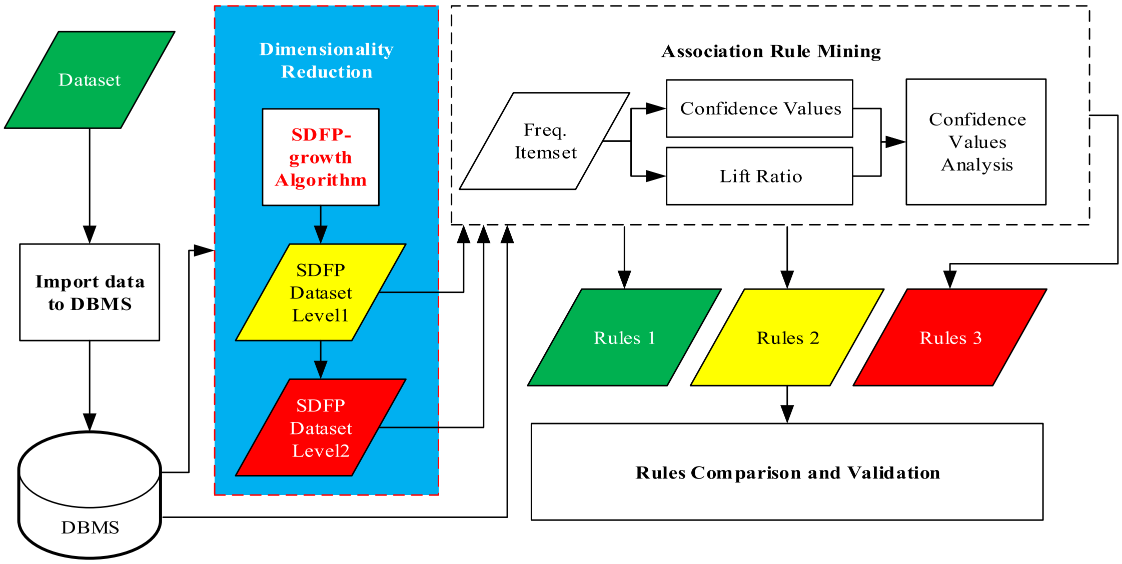

FIGURE 1. The dimensional reduction model using the SDFP-growth algorithm as a proposed model is shown in a blue box, consisting of proposed SDFP-growth algorithms that will result in SDFP Dataset level 1 and SDFP Dataset level 2. Three datasets will be processed on the confidence values and lift ratio with the output of Rules 1, Rules 2, and Rules 3. The three obtained rules will be compared and evaluated.

The combination of two sets, for example, sets A and B, can be written as $A \cup B$ where $A \subseteq U$ or $B \subseteq U$, in set theory, is known as a Union. The intersection of two related sets can be written as $A \cap B$ where $A \subseteq U$ and $B \subseteq U$ are known as Intersections. The Difference is a reduction in the members of a set based on another set; it can be written as $A - B$ where $A \subseteq U$ and $B \subseteq U$ are known as Set Differences. A subset is a set as a whole, which is a member of another set; it can be written as $A \subseteq B$ where $A \subseteq U$, $B \subseteq U$ and $\forall x\ [x \in A \rightarrow x \in B]$; in set theory, it is called Subset.

IV. PROPOSED METHOD (SDFP-GROWTH ALGORITHM)

This study proposed a new algorithm called the Set Difference FP-Growth (SDFP-growth) algorithm. It shows the implementation of a custom set difference theory in a database environment represented as an adjacency table. This study proposes SDFP-growth level 1 and SDFP-growth level 2, where level 1 reduces the dimensions of the original data or raw data, whereas level 2 reduces the dimension of data to 55% [7], [40] raw data size.

Algorithm 1 presents the SDFP-growth level 1 algorithm pseudocode. The input is obtained from the raw data. *FreqTable* variable that sorts the appearance of items from raw data, which is then sorted in descending order. The *AdjacencyTable* variable represents the structure of the FP-Tree dataset, consisting of a collection of connected parents and children based on their appearance in the raw data records. The SDFP_Table_Level1 variable was formed by implementing the raw data set difference to the adjacency table.

The Algorithm 2 shows the SDFP-growth level 2 algorithm pseudocode. Similar to SDFP-growth level 1, a frequent table containing the number of occurrences of items and an adjacency table containing the FP-Tree structure were formed. SDFP-growth level 2 performs an initial reduction process on raw data sorted in descending order of 55% [7]. The SDFP-growth Level 2 table was formed from implementing the 55% difference dataset against the adjacency table.

An example of a dummy dataset from [36] consists of six rows of data with six items. In the association rule mining process, all items are sorted based on the highest number of occurrences, as shown in Table 2 . Table 2 (a) shows the original dataset that was not sorted. Table 2 (b) shows the frequency of occurrence of each item in the dataset; most items were stored at the top. Table 2 (c) shows the arrangement of

Algorithm 1 Proposed SDFP-Growth Level 1 Algorithm

```
Procedure createSDFP_Level1 (Dataset)
    FreqTable = EmptyTable
    AdjacencyTable = {id, parent, child}
    SDFP_Table_Level1 = EmptyTable
Begin
    FreqTable ← SortDescending(ItemOccurrenceList(Dataset))
    where count > 1
    For j in FreqTable do
        If Dataset.item = j.item then
            AdjacencyTable(Dataset)
        End if
    End for
    For i in AdjacencyTable do
        SDFP_Table_Level1.append(setDifference(Dataset.
        item, i.child))
    End for
    return SDFP_Table_Level1
End
```

Algorithm 2 Proposed SDFP-Growth Level 2 Algorithm

```
Procedure createSDFP_Level2 (Dataset)
    Dataset55 = EmptyTable
    FreqTable = EmptyTable
    AdjacencyTable = {id, parent, child}
    SDFP_Table_Level2 = EmptyTable
Begin
    FreqTable←SortDescending(ItemOccurrenceList(Dataset))
    where count > 1
    For k in dataset do
        If k.item in FreqTable And sum(FreqTable)>55%
         then
            Add k.item to Dataset55
        End if
    End for
    For j in FreqTable do
        If Dataset55.item = j.item then
            generate AdjacencyTable(Dataset)
        End if
    End for
    For i in AdjacencyTable do
        SDFP_Table_Level2.append(setDifference(Dataset5
        5.item, i.child))
    End for
    return SDFP_Table_Level2
End
```

TABLE 2. Description of the attributes in the dummy dataset: (a) original dataset, (b) frequency on each item, (c) sorted original dataset.

TID	Items
T1	I2, I3, I5
T2	I6, I2
T3	I3, I1, I4
T4	I4, I2, I3, I1, I5
T5	I3, I5, I4
T6	I5, I6

(a)

Item	Count
I3	4
I5	4
I2	3
I4	3
I1	2
I6	2

(b)

TID	Items
T1	I3, I5, I2
T2	I2, I6
T3	I3, I4, I1
T4	I3, I5, I2, I4, I1
T5	I3, I5, I4
T6	I5, I6

(c)

the dataset sorted by occurrence; more items are mentioned first.

The research methodology used in this study is illustrated in Figure 1. There are four main processes in this study, namely the process of importing data from flat files to the DBMS, the process of dimensionality reduction, the process of association rule mining, and finally, the process of rules comparison and validation.

The first step is to import the data into the DBMS environment for the original dataset. The use of a DBMS environment is proposed to use the ability to process high-dimensionality raw data and implement the FP-growth algorithm inside the DBMS a novelty technique. A small dummy dataset was used in this study. The data import process can be performed in two ways: manually creating a table or carrying out the extract-transform-loading (ETL) process using the features available in the Oracle SQL Developer [41].

The next stage was to perform dimensionality reduction using the proposed method. At this stage, two datasets were formed, which acquired the theory of reshaped and reduced datasets [7]. The results obtained were SDFP dataset level 1 and SDFP dataset level 2. The SDFP dataset level 2 should have a smaller dimension size than the SDFP dataset level 1.

In the association rule mining stage in Figure 1, the frequent pattern formation process is carried out from three dataset sources: the original Dataset, SDFP-growth level 1 dataset, and SDFP-growth level 2 dataset. The three frequent patterns that are formed are each processed by calculating the confidence value and lift ratio. The analysis was carried out on the three datasets by comparing the confidence value to the lift ratio; theoretically, only rules with a lift ratio greater than one are considered valid.

In the rule comparison and validation stage shown in Figure 1, a comparison process is carried out between the produced rules. The goal is to obtain rules from SDFP-growth level 1 and SDFP-growth level 2 whose confidence values are identical to the strong rules. Predictably, even though the number of rules from SDFP-growth level 2 is fewer than that of other datasets, it will still produce relatively the same confidence rule values. The environment used in this study is the Oracle Database [42] along with Oracle SQL and Oracle PL/SQL [43], which are managed using the Oracle SQL Developer [41].

The prediction of frequent itemsets that will be formed can be predicted using equation 6 [44]. The number of Frequent Patterns (NFP) is the predicted number of frequent itemsets that will be obtained, and n is the number of distinct items found in the dataset. For example, based on a dataset from [36] comprising 6 items, the prediction of the rules obtained was 63.

$$NFP = 2^n - 1. \tag{5}$$

The proposed optimization method predicts the number of frequent patterns obtained after dimensionality reduction on the original dataset. The basic idea is to obtain a smaller dataset with fewer cardinalities of distinct items. Figure 2 shows the proposed optimization process for predicting frequent patterns. The dataset was transformed into FP-Tree structures as an adjacency table inside the database. Dimensionality reduction is implemented on FP-Tree, which forms a smaller dataset along with the pruned FP-Tree structure.

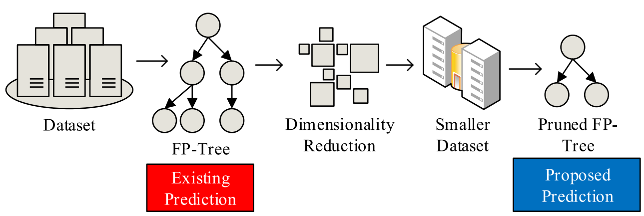

FIGURE 2. The proposed optimization methods on the Number of Frequent Patterns Prediction based on dimensionality reduction using the SDFP-growth algorithm. The proposed prediction should be smaller than the existing prediction.

The proposed optimization is evaluated on two important measurements: the number of dataset reductions and

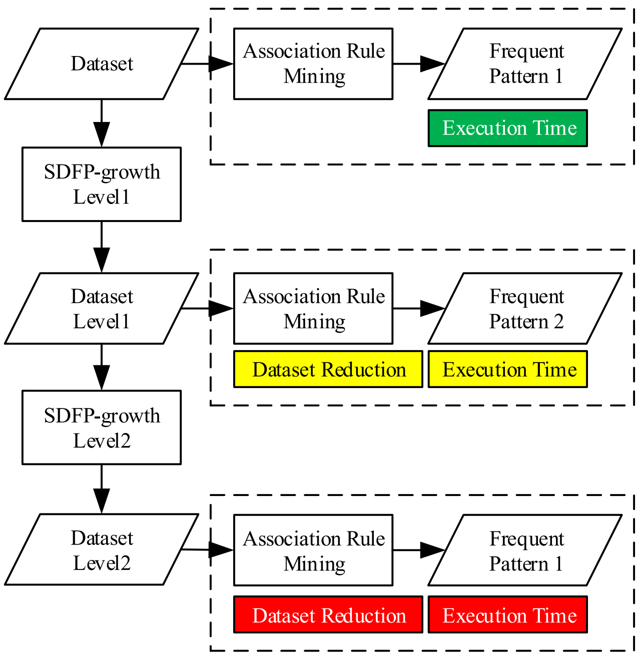

FIGURE 3. The evaluation scenario on proposed optimization methods for measuring the number of dataset reductions and the execution times. Three datasets will be evaluated.

TABLE 3. The dummy dataset consists of 6 records and six items: (a) sorted original dataset, (b) SDFP-growth level 1 dataset, and (c) SDFP-growth level 2 dataset.

TID	Items
T1	I3, I5, I2
T2	I2, I6
T3	I3, I4, I1
T4	I3, I5, I2, I4, I1
T5	I3, I5, I4
T6	I5, I6

(a)

TID	Items
T1	I3, I5, I2
T2	I2
T3	I3, I4
T4	I3, I5, I2, I4
T5	I3, I5, I4
T6	I5

(b)

TID	Items
T1	I3, I5
T2	-
T3	I3
T4	I3, I5
T5	I3, I5
T6	I5

(c)

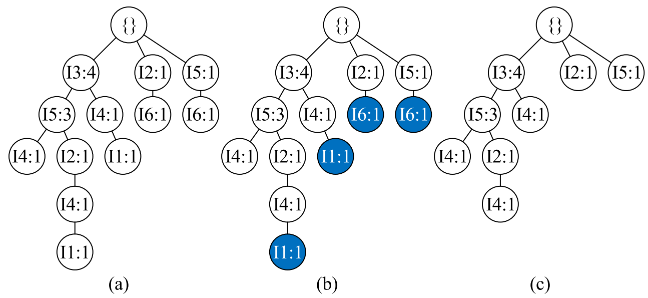

FIGURE 4. The illustration of FP-tree transformation for obtaining SDFP-growth level 1 dataset: (a) FP-Tree structure on original dummy dataset, (b) The pruned item on the original dataset, (c) The obtained FP-Tree on SDFP-growth level 1 dataset.

execution times, as shown in Figure 3. The measurements of the number of dataset reductions will be implemented on the original dataset, which will obtain two reduced datasets: SDFP-growth level 1 and SDFP-growth level 2. Both reduced datasets are observed based on the number of reductions. The measurements of the execution times were implemented on three datasets: the original dataset, SDFP-growth level 1 dataset, and SDFP-growth level 2 datasets. The observation of execution time is done by comparing the efficiency of association rule formations against the three datasets; the smaller dataset should obtain shorter times. The evaluation of the number of dataset reductions and the execution times is implemented on five datasets, as shown in Table 5 .

V. RESULTS AND DISCUSSION

The experiment was done on an Intel Core i5-4590 CPU @ 3.30 GHz 3.30GHz, 8 GB of Installed memory (RAM). Table 3 (a) shows the initial dataset before reduction, Table 3 (b) shows the SDFP-growth level 1 dataset after reduction with the proposed algorithm, and Table 3 (c) shows the SDFP-growth level 2 dataset. There is an item reduction process in several rows of data, where the SDFP-growth level 1 dataset has smaller data dimensions than the original dataset, and the SDFP-growth level 2 dataset has smaller dimensions than the SDFP-growth level 2 dataset.

Figure 4 illustrates the formation of the SDFP-growth level 1 dataset from the original dataset in the FP-Tree structure. Figure 4(a) shows the FP-Tree structure of the original Dataset, Figure 4(b) shows the items that were eliminated by the proposed algorithm process, and Figure 4(c) shows the result of the SDFP-growth level 2 dataset in the form of an FP-Tree. There is shrinkage of the tree shape in the illustration of the image.

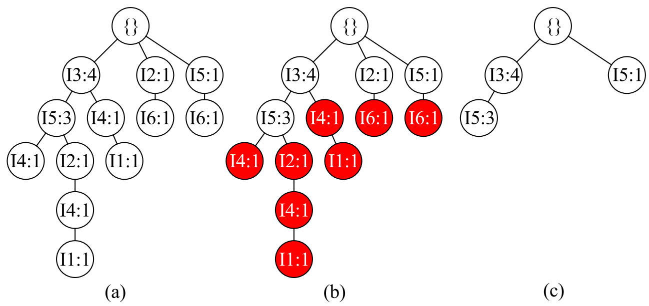

FIGURE 5. The illustration of FP-tree transformation for obtaining SDFP-growth level 2 dataset: (a) FP-Tree structure on original dummy dataset, (b) Illustration of the pruned item on the original dataset, (c) Obtained FP-Tree on SDFP-growth level 2 dataset.

Figure 5 illustrates the formation of the SDFP-growth Level 2 dataset as an FP tree. Figure 5(a) show the FP-Tree structure of the original Dataset, Figure 5(b) shows the process of reducing items because of the implementation of the proposed algorithm, and Figure 5(c) shows the FP-Tree structure of the SDFP-growth level 2. SDFP tree structure level 2 was smaller than SDFP tree structure level 1. Even though the resulting FP-Tree structure is smaller, the top items with a high frequency of occurrence are maintained. This retains the association rules with high confidence values, which are strong rules, as shown in Figure 6.

TABLE 4. Association rules, support values, confidence value, and lift ratio obtained of SDFP-growth Algorithm on the original dataset, SDFP-growth Level 1 dataset, and SDFP-growth Level 2 dataset.

Rule	Original			SDFP-growth L1			SDFP-growth L2		
	Sup	Conf	Lift	Sup	Conf	Lift	Sup	Conf	Lift
I4 -> I3	0.5	1	1.5	0.5	1	1.5	-	-	-
I3 -> I4	0.5	0.8	1.5	0.5	0.8	1.5	-	-	-
I3 -> I5	0.5	0.8	1.1	0.5	0.8	1.1	0.6	0.8	0.9
I5 -> I3	0.5	0.8	1.1	0.5	0.8	1.1	0.6	0.8	0.9
I2 -> I3	0.3	0.7	1	0.3	0.7	1	-	-	-

Table 4 shows the top five results of the SDFP-growth algorithm implementation on the original dataset, SDFP-growth level 1 dataset, and SDFP-growth level 2 dataset. The table shows the association rules obtained, the support values, the confidence values, and the lift ratio. The original dataset obtained 24 rules, the SDFP-growth level 2 dataset obtained 12 rules, and the SFP-growth level 2 dataset obtained two rules. We found identical results for the original dataset and the SDFP-growth level 1 dataset. The SDFP-growth level 2 dataset results show equal confidence values on the 3^{rd} and 4^{th} rules with a slight decrease in the lift ratio, which means that they still have similar results on strong rules against the original dataset.

Based on these findings, equation 7 shows the proposed formula for predicting the number of frequent itemset using the SDFP-growth algorithm. The notation $|A|$ is the cardinality of sets consisting of deducted items, $|A| < n$, n is the original cardinality of sets based on equation 6. Based on the basic computational principal theory, a reduced dataset results in a smaller cardinality.

$$NFP = 2^{|A|} - 1; \ |A| < n. \tag{6}$$

TABLE 5. Dataset characteristics on five data sources used consist of the number of records and number of distinct items on every data source with the source link.

Dataset	Number of Records	Number of Distinct Items	Data Source
Zoo	101	28	https://archive.ics.uci.edu/dataset/111/zoo
Cardio	33988	29	https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset
MBA	522061	3934	https://www.kaggle.com/datasets/aslanahmedov/market-basket-analysis
Food Nutrition	1296	23	https://www.kaggle.com/datasets/rakkesharv/fast-food-joint-nutrition-values-dataset
Minimarket	50000	3567	Transaction data from minimarket in Bandung, West Java

Table 5 lists the data used as experimental data sources. It contains five different datasets from various domains: zoo, cardiovascular, market basket, food nutrition, and real-time transactions in the minimarket dataset. The zoo dataset has the smallest number of records, 101 records with 28 distinct items; meanwhile, the market basket analysis dataset has the highest number of records, 522061 records with 3934 distinct items. Four datasets are from a public dataset, and one, the minimarket dataset, is from real-time store transactions in Bandung, West Java.

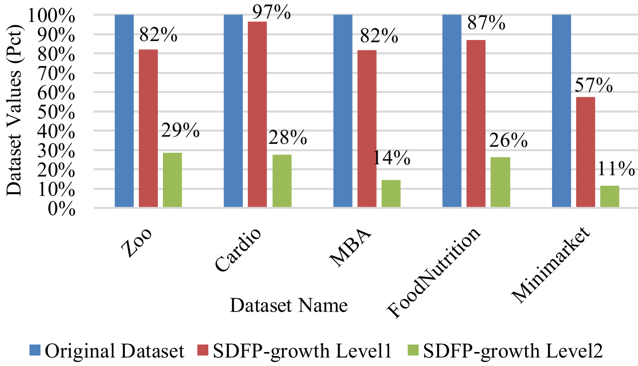

FIGURE 6. Optimization results on dataset reduction of Zoo dataset, Cardiovascular Dataset, Market Basket Analysis dataset, Food Nutrition dataset, and minimarket transactional dataset. The percentage value indicates the remaining size of the optimized dataset.

Figure 6 shows the optimization results for the number of dataset reductions obtained from the five datasets in Table 4. All datasets were successfully reduced by implementing the SDFP-growth algorithms for both levels 1 and 2. The reduction in level 2 was higher than that at level 1. The zoo dataset was reduced by 18% for Level 1 and 71% for Level 2. The cardiovascular dataset was reduced by 3% for Level 1 and 72% for Level 2. The market basket analysis datasets were reduced by 18% for Level 1 and 86% for Level 2. The food nutrition dataset was reduced by 13% for Level 1 and 74% for Level 2. The Minimarket dataset was reduced by 43% for Level 1 and 89% for Level 2 datasets.

Figure 7 shows the optimization execution times for the five datasets. It compares the execution times of the original datasets against those of the SDFP-growth level 1 and SDFP-growth level 2 datasets. The execution times on the Zoo dataset were 7% optimized on the SDFP-growth level 1 dataset and 93% optimized on the SDFP-growth level 2 dataset. The execution times on the Cardio dataset were 2% optimized on the SDFP-growth level 1 dataset and 93% optimized on the SDFP-growth level 2 dataset. The execution times on the MBA dataset were 3% optimized on the SDFP-growth level 1 dataset and 38% optimized on the SDFP-growth level 2 dataset. The execution times on the FoodNutrition dataset were optimized by 11% on the SDFP-growth level 1 dataset and 94% on the SDFP-growth level 2 dataset. The execution times on the Minimarket dataset were 9% optimized on the SDFP-growth level 1 dataset and 42% optimized on the SDFP-growth level 2 dataset.

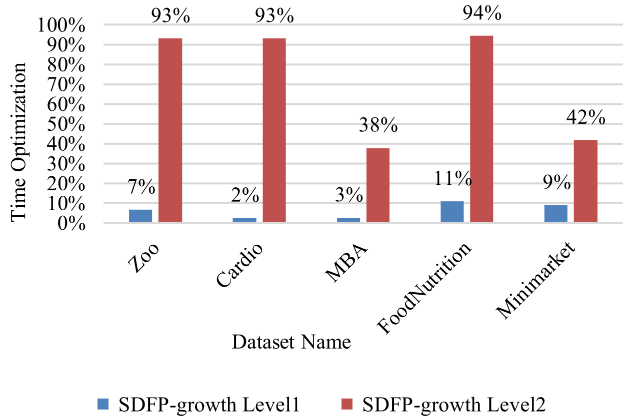

FIGURE 7. Optimization results on execution times of Zoo dataset, Cardiovascular Dataset, Market Basket Analysis dataset, Food Nutrition dataset, and minimarket transactional dataset for obtaining frequent patterns. A higher percentage value indicates a faster result.

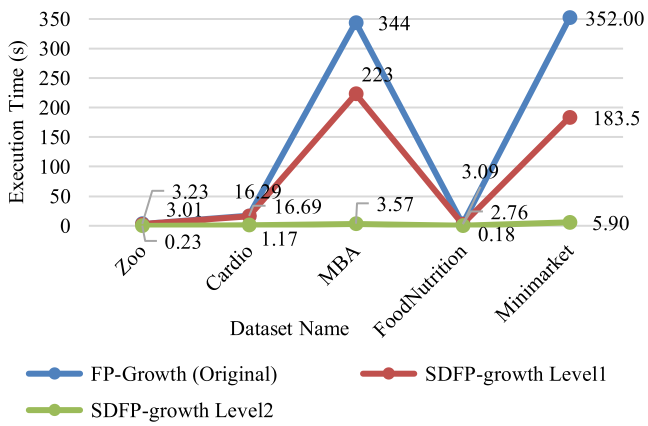

FIGURE 8. Optimization results on execution times of Zoo dataset, Cardiovascular Dataset, Market Basket Analysis dataset, Food Nutrition dataset, and minimarket transactional dataset when performing frequent pattern. The dot indicates the execution times in seconds.

Figure 8 shows the gap execution times between the original Dataset, SDFP-growth level 1 dataset, and SDFP-growth level 2 dataset on the five public datasets. A small gap was found between the execution times on the original dataset and the SDFP-level 1 dataset, but a large gap was found between the original dataset and the SDFP-level 2 dataset. The execution time gap between the Zoo original dataset and the SDFP-growth dataset level 1 is 0.2 seconds or 7% faster; on the SDFP-growth dataset level 2 is 3 seconds or 93% faster. The execution time gap between the Cardio original dataset and SDFP-growth dataset level 1 is 0.4 seconds or 2% faster; on the SDFP-growth level 2 dataset, it is 15.5 seconds or 93% faster. The execution time gap between the MBA original dataset and the SDFP-growth dataset level 1 was 121 or 35% faster; on the SDFP-growth level 2 dataset, it was 340 or 99% faster. The execution time gap between the FoodNutrition original dataset and the SDFP-growth dataset level 1 is 0.3 seconds or 11% faster; on the SDFP-growth level 2 dataset, it is 2.9 seconds or 94% faster. The execution time gap between the Minimarket original dataset and the SDFP-growth dataset level 1 was 168.5 seconds or 48% faster; on the SDFP-growth level 2 dataset, it was 346 or 98% faster.

TABLE 6. Confidence values comparison on top five association rules of Zoo's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Confidence		
	Original	SDFP-L1	SDFP-L2
'backbone', 'tail'	89.16	89.16	89.16
'backbone', 'breathes'	83.13	83.13	83.13
'backbone', 'toothed'	73.50	73.50	73.50
'breathes', 'tail'	76.25	76.25	76.25
'tail', 'toothed'	69.34	69.34	69.34

TABLE 7. Lift ratio comparison on top five association rules of Zoo's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Lift Ratio		
	Original	SDFP-L1	SDFP-L2
'backbone', 'tail'	1.20	1.20	1.20
'backbone', 'breathes'	1.05	1.05	1.05
'backbone', 'toothed'	1.22	1.22	1.22
'breathes', 'tail'	1.03	1.03	1.03
'tail', 'toothed'	1.15	1.15	1.15

The top five association rules results for Zoo's dataset are shown in Tables 6 and 7. Table 6 shows the confidence value comparison on the original SDF-growth level 1 and SDFP-growth level 2 datasets. We found identical results for the three confidence values, which were greater than 73.5%. Table 7 presents a comparison of lift ratios. There was no difference between the original, SDFP-growth level 1 and SDFP-growth level 2 lift ratio results, which were greater than one.

TABLE 8. Confidence values comparison on top five association rules of Cardio's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Confidence		
	Original	SDFP-L1	SDFP-L2
'Alco_No', 'Smoke_No'	93.67	93.67	93.68
'Alco_No', 'Gluc_Normal'	81.97	81.97	81.98
'Gluc_Normal', 'Smoke_No'	91.71	91.71	91.71
'Active_Yes', 'Alco_No'	94.57	94.57	94.57
'Active_Yes', 'Smoke_No'	91.46	91.46	91.46

Tables 8 and 9 show the top five association rules results of Cardio's dataset. Table 8 shows the confidence value comparison of the original SDF-growth level 1 and SDFP-growth level 2 datasets. We found identical results for the three confidence values of more than 81.98%. Table 9 presents a comparison of lift ratios. There was no difference between the

TABLE 9. Lift ratio comparison on top five association rules of Cardio's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Lift Ratio		
	Original	SDFP-L1	SDFP-L2
'Alco_No', 'Smoke_No'	1.02	1.02	1.02
'Alco_No', 'Gluc_Normal'	1.00	1.00	1.00
'Gluc_Normal', 'Smoke_No'	1.00	1.00	1.00
'Active_Yes', 'Alco_No'	1.00	1.00	1.00
'Active_Yes', 'Smoke_No'	1.00	1.00	1.00

original, SDFP-growth level 1 and SDFP-growth level 2 lift ratio results, which were greater than or equal to one.

TABLE 10. Confidence values comparison on top five association rules of market basket Analysis's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Confidence		
	Original	SDFP-L1	SDFP-L2
'other vegetables', 'whole milk'	38.66	38.68	38.68
'rolls/buns', 'whole milk'	30.55	21.89	30.58
'whole milk', 'yogurt'	21.88	40.14	21.90
'root vegetables', 'whole milk'	44.86	44.84	44.83
'other vegetables', 'root vegetables'	24.50	24.49	24.49

TABLE 11. Lift ratio comparison on top five association rules of market basket Analysis's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Lift Ratio		
	Original	SDFP-L1	SDFP-L2
'other vegetables', 'whole milk'	1.51	2.76	1.32
'rolls/buns', 'whole milk'	1.20	1.56	1.05
'whole milk', 'yogurt'	1.57	2.87	1.37
'root vegetables', 'whole milk'	1.76	3.20	1.53
'other vegetables', 'root vegetables'	2.25	2.24	1.96

Tables 10 and 11 show the top five association rules resulting from the Market Basket Analysis dataset. Table 10 shows the confidence value comparison of the original SDF-growth level 1 and SDFP-growth level 2 datasets. We found identical results for the three confidence values, which were more than 81.98%. Table 11 shows a comparison of the lift ratios. There was no difference between the original, SDFP-growth level 1 and SDFP-growth level 2 lift ratio results, which were greater than or equal to one.

The top five association rules results of the Nutrition dataset are shown in Tables 12 and 13. Table 12 shows the confidence value comparison of the original SDF-growth level 1 and SDFP-growth level 2 datasets obtained. We found identical results for the three confidence values, which were greater than 84.03%. Table 13 shows a comparison of the lift ratios. There was no difference between the original,

TABLE 12. Confidence values comparison on top five association rules of Nutrition's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Confidence		
	Original	SDFP-L1	SDFP-L2
'Chol1', 'Sodium1'	100.00	100.00	100.00
'Chol1', 'Sugar1'	93.06	93.06	93.06
'Sodium1', 'Sugar1'	93.06	93.06	93.06
'Chol1', 'Energy2'	84.03	84.03	84.03
'Energy2', 'Sodium1'	100.00	100.00	100.00

TABLE 13. Lift ratio comparison on top five association rules of Nutrition's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Lift Ratio		
	Original	SDFP-L1	SDFP-L2
'Chol1', 'Sodium1'	1.00	1.00	1.00
'Chol1', 'Sugar1'	1.00	1.00	1.00
'Sodium1', 'Sugar1'	1.00	1.00	1.00
'Chol1', 'Energy2'	1.00	1.00	1.00
'Energy2', 'Sodium1'	1.00	1.00	1.00

SDFP-growth level 1 and SDFP-growth level 2 lift ratio results, which were greater than or equal to one.

TABLE 14. Confidence values comparison on top five association rules of Minimarket's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Confidence		
	Ori ginal	SDFP -L1	SDFP-L2
'Ind, Ayam Bawang', 'Telur, Aym Ngr Crh'	25.37	25.42	25.43
'Ind, Grg Spc Saus', 'Telur, Aym Ngr Crh'	23.28	23.33	23.20
'Fortune, Pouch 2lt', 'Telur, Aym Ngr Crh'	21.88	21.79	21.78
'Telur, Aym Ngr Crh', 'Yg, Gula Lokal 1kg'	4.76	4.79	4.80
'Ind, Ayam Bawang', 'Ind, Grg Spc Saus'	10.07	10.02	10.03

TABLE 15. Lift ratio comparison on top five association rules of market basket Analysis's Original, SDFP-growth level1, and SDFP-growth level2 dataset.

Frequent Pattern	Lift Ratio		
	Ori ginal	SDFP-L1	SDFP -L2
'Ind, Ayam Bawang', 'Telur, Aym Ngr Crh'	1.29	1.28	1.20
'Ind, Grg Spc Saus', 'Telur, Aym Ngr Crh'	1.18	1.18	1.09
'Fortune, Pouch 2lt', 'Telur, Aym Ngr Crh'	1.11	1.10	1.03
'Telur, Aym Ngr Crh', 'Yg, Gula Lokal 1kg'	1.12	1.12	1.05
'Ind, Ayam Bawang', 'Ind, Grg Spc Saus'	1.74	1.72	1.61

Tables 14 and 15 list the top five association rules resulting from the minimarket's primary dataset. Table 14 shows the confidence value comparison of the original SDF-growth level 1 and SDFP-growth level 2 datasets obtained. We found identical results for the three confidence values, which were greater than 81.98%. Table 15 shows the lift ratio comparison. There was no difference between the original, SDFP-growth level 1, and SDFP-growth level 2 lift ratio results, which were greater than or equal to one.

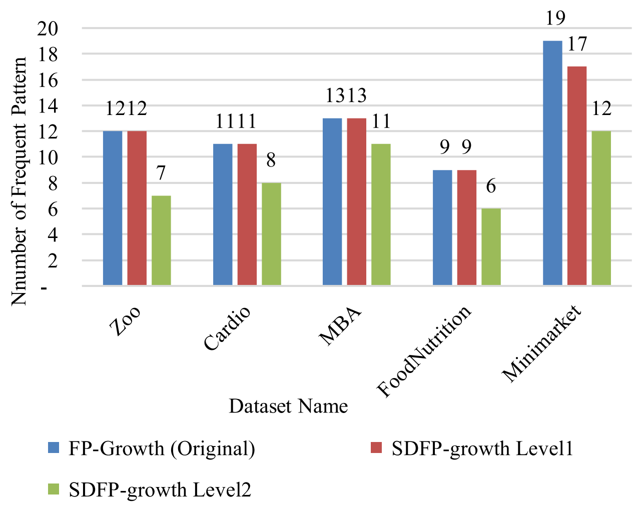

FIGURE 9. Number of K-items obtained of Zoo dataset, Cardiovascular Dataset, Market Basket Analysis dataset, Food Nutrition dataset, and minimarket transactional dataset.

Figure 9 shows the number of k-items obtained from the five datasets used. The zoo's original and SDFP-growth level 1 dataset found the same number of rules obtained, which was 12, and decreased the SDF-growth level 2 dataset to 7. The cardio's original and SDF-growth level 1 dataset found the same quantity of the number of rules obtained, which is 11 rules, and decreased on the SDF-growth level 2 dataset to eight rules. In the market basket analysis, the original and SDF-growth level 1 datasets found the same number of rules obtained, which were 13 rules, and it decreased on the SDF-growth level 2 dataset to 11 rules. The foodNutrition's original and SDF_growth level 1 dataset found the same quantity of the number of rules obtained, which is nine rules, and it decreased on the SDF-growth level 2 dataset to six rules. The minimarket dataset found 19 rules on the original dataset, 17 rules on the SDFP-growth level 1 dataset, and 12 rules on the SDF-growth level 2 dataset. The SDFP-growth level 2 dataset results in smaller quantities of k-items compared with the original and SDFP-growth level 1 dataset.

The proposed SDFP-growth method was compared with an adaptive support method [15] using two datasets, the Chess dataset and the Mushroom dataset [45]. SPFM tools were used for comparison [46]. In the comparison of the characteristics of the dataset shown in Table 16, the Chess dataset was reduced by 6.7% for the number of distinct items in the SDFP growth-level 1 dataset and by 69.3% for the SDFP-growth level 2 dataset. The Mushroom dataset is reduced by 15.1% in the number of distinct items in the SDFP growth-level 1 dataset and reduced by 82.5% for the SDFP-growth level 2 dataset.

TABLE 16. Number of items reduction result on Chess dataset and Mushroom dataset by using SDFP-growth Level 1 and SDFP-growth Level 2 proposed method.

Dataset	Number of Records	Number of Items		
		Original	SDFP-L1	SDFP-L2
Chess	3,196	75	70	23
Mushroom	8,124	119	101	20

TABLE 17. Number of rules obtained comparison between adaptive support method and SDFP-growth proposed method on Chess dataset.

Chess Dataset (Apriori-Number of Rules)			
MinSup	Adaptive Support	SDFP-L1	SDFP-L2
90%	10,742	10,742	6,076
80%	552,564	552,564	481,846
70%	8,111,370	8,111,370	7,530,268
60%	83,735,890	83,864,464	75,190,748
50%	879,828,936	880,936,478	625,170,214

The number of rules obtained was then compared. Table 17 shows at comparison of the number of rules obtained using adaptive support methods compared to the SDFP-growth level 1 and SDFP-growth level 2 proposed method using the Chess dataset. The SDFP-growth Level 2 dataset reduces the obtained association rules by more than 7%.

TABLE 18. Number of rules obtained comparison between adaptive support method and SDFP-growth proposed method on Mushroom dataset.

Mushroom Dataset (Apriori-Number of Rules)			
MinSup	Adaptive Support	SDFP-L1	SDFP-L2
90%	22	14	14
80%	52	88	88
70%	180	180	180
60%	266	266	266
50%	1,248	1,248	1,248
40%	5,904	5,020	4,890
30%	78,888	74,894	51,550
20%	19,174,370	19,171,655	1,683,930

Table 18 shows a comparison of the number of rules obtained between the adaptive support method and the SDFP-growth proposed method on the mushroom dataset. When the minimum support is below 50%, the number of rules obtained by the proposed SDFP-growth level 1 method is smaller than the number of rules obtained by the adaptive support method. The SDFP-growth level 2 proposed method obtained smaller numbers than the SDFP-growth level 2 proposed method by more than 17%.

VI. CONCLUSION

The SDFP-growth algorithm can improve the execution times of frequent pattern generation. The improvement in execution times is due to the reduction in the dimensions of the datasets. Even if dimensionality reduction occurred, the strong rules remained identical in the original Dataset, SDFP-growth level 1 dataset, and SDFP-growth level 2 dataset. Dimensional reduction changes the formula for predicting the maximum frequent patterns optimized from $2^n - 1$ to $2^{|A|}$ - 1; $|A| < n$. The finding based on the experiments is that the optimization reduces the number of data dimensions by more than 3% on the Level 1 dataset and more than 69% on the Level 2 dataset, while the frequent pattern generation time improved by more than 2% on the Level 1 dataset, and more than 94% on the Level 2 dataset.

Future research will expand the implementation of the resulting dataset using techniques other than association rule mining. Level 1 and level 2 output datasets will be implemented for use in other machine learning techniques, such as classification and clustering, to optimize the computing processes.

REFERENCES

[1] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Manage. data*, Jun. 1993, pp. 207–216.

[2] N. Lakshmi and M. Krishnamurthy, "Frequent itemset generation using association rule mining based on hybrid neural network based billiard inspired optimization," *J. Circuits, Syst. Comput.*, vol. 31, no. 8, May 2022, doi: 10.1142/s0218126622501389.

[3] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*, 1994, pp. 487–499.

[4] S. Neelima, N. Satyanarayana, and P. K. Murthy, "A survey on approaches for mining frequent itemsets," *IOSR J. Comput. Eng.*, vol. 16, no. 4, pp. 31–34, 2014.

[5] H. I. Ahmad, A. T. H. Sim, R. Ibrahim, M. Abrar, and A. Gul, "Mining predicate rules without minimum support threshold," *Kuwait J. Sci.*, vol. 48, no. 4, pp. 1–9, Aug. 2021, doi: 10.48129/kjs.v48i4.9782.

[6] G. O. Regan, *Guide To Discrete Mathematics*. Cham, Switzerland: Springer, 2016.

[7] B. Siswanto, E. Tanuar, and R. Rahmania, "Reshaped and reduced dimensionality reduction data technique on association rule mining," in *Proc. 3rd Int. Symp. Mater. Electr. Eng. Conf. (ISMEE)*, Nov. 2021, pp. 87–91.

[8] M. Abdel-Basset, M. Mohamed, F. Smarandache, and V. Chang, "Neutrosophic association rule mining algorithm for big data analysis," *Symmetry*, vol. 10, no. 4, p. 106, Apr. 2018, doi: 10.3390/sym10040106.

[9] D. W. Barowy, E. D. Berger, and B. Zorn, "ExceLint: Automatically finding spreadsheet formula errors," *Proc. ACM Program. Lang.*, vol. 2, pp. 1–26, Oct. 2018.

[10] M. P. Deisenroth, A. A. Faisal, and C. S. Ong, *Mathematics for Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2020.

[11] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, May 2000.

[12] M. Narvekar and S. F. Syed, "An optimized algorithm for association rule mining using FP tree," *Proc. Comput. Sci.*, vol. 45, pp. 101–110, Jan. 2015.

[13] S. U. Rehman, N. Alnazzawi, J. Ashraf, J. Iqbal, and S. Khan, "Efficient top-K identical frequent itemsets mining without support threshold parameter from transactional datasets produced by IoT-based smart shopping carts," *Sensors*, vol. 22, no. 20, p. 8063, Oct. 2022. [Online]. Available: https://www.mdpi.com/1899542

[14] S. Iqbal, A. Shahid, M. Roman, Z. Khan, S. Al-Otaibi, and L. Yu, "TKFIM: Top-K frequent itemset mining technique based on equivalence classes," *PeerJ Comput. Sci.*, vol. 7, p. e385, Mar. 2021.

[15] E. Hikmawati, N. U. Maulidevi, and K. Surendro, "Minimum threshold determination method based on dataset characteristics in association rule mining," *J. Big Data*, vol. 8, no. 1, pp. 1–17, Dec. 2021.

[16] J. Hvorecký, L. Korenova, and T. Barot, "Combining brute force and IT to solve difficult problems," in *Proc. 27th Asian Technol. Conf. Math.*, 2022.

[17] J. Guo, D. Hermelin, and C. Komusiewicz, "Local search for string problems: Brute-force is essentially optimal," *Theor. Comput. Sci.*, vol. 525, pp. 30–41, Mar. 2014, doi: 10.1016/j.tcs.2013.05.006.

[18] B. S. Neysiani, N. Soltani, R. Mofidi, and M. H. Nadimi-Shahraki, "Improve performance of association rule-based collaborative filtering recommendation systems using genetic algorithm," *Int. J. Inf. Technol. Comput. Sci.*, vol. 11, no. 2, pp. 48–55, Feb. 2019. [Online]. Available: https://www.researchgate.net/profile/Behzad-Soleimani-Neysiani/publication/330761376_Improve_Performance_of_Association_Rule-Based_Collaborative_Filtering_Recommendation_Systems_using_Genetic_Algorithm/links/5c6410d045851582c3e5a94a/Improve-Performance-of

[19] A. Salam and M. S. H. Khayal, "Mining top-k frequent patterns without minimum support threshold," *Knowl. Inf. Syst.*, vol. 30, no. 1, pp. 57–86, Jan. 2012, doi: 10.1007/s10115-010-0363-3.

[20] J. Ashraf, A. Habib, and A. Salam, "Top-*K* miner: Top-*K* identical frequent itemsets discovery without user support threshold," *Knowl. Inf. Syst.*, vol. 48, no. 3, pp. 741–762, Sep. 2016, doi: 10.1007/s10115-015-0907-7.

[21] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea, "Toward a quantitative survey of dimension reduction techniques," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 3, pp. 2153–2173, Mar. 2021, doi: 10.1109/TVCG.2019.2944182.

[22] B. Nath, "Dimensionality reduction for association rule mining," *Int. J. Intell. Inf. Process.*, vol. 2, pp. 1–15, Oct. 2015.

[23] S. Velliangiri, S. Alagumuthukrishnan, and S. I. Th. Joseph, "A review of dimensionality reduction techniques for efficient computation," *Proc. Comput. Sci.*, vol. 165, pp. 104–111, Jan. 2019.

[24] I. Aqra, N. Abdul Ghani, C. Maple, J. Machado, and N. Sohrabi Safa, "Incremental algorithm for association rule mining under dynamic threshold," *Appl. Sci.*, vol. 9, no. 24, p. 5398, Dec. 2019, doi: 10.3390/app9245398.

[25] N. H. Miswan, I. M. Sulaiman, C. S. Chan, and C. G. Ng, "Association rules mining for hospital readmission: A case study," *Mathematics*, vol. 9, no. 21, pp. 1–21, 2021.

[26] B. Siswanto and P. Thariqa, "Association rules mining for identifying popular ingredients on Youtube cooking recipes videos," in *Proc. Indonesian Assoc. Pattern Recognit. Int. Conf. (INAPR)*, Sep. 2018, pp. 95–98, doi: 10.1109/INAPR.2018.8627002.

[27] K. Tatiana and M. Mikhail, "Market basket analysis of heterogeneous data sources for recommendation system improvement," *Proc. Comput. Sci.*, vol. 136, pp. 246–254, Jan. 2018.

[28] S. J. Lee and K. B. Cartmell, "An association rule mining analysis of lifestyle behavioral risk factors in cancer survivors with high cardiovascular disease risk," *J. Personalized Med.*, vol. 11, no. 5, p. 366, May 2021, doi: 10.3390/jpm11050366.

[29] H.-P. Liew, "Dietary habits and physical activity: Results from cluster analysis and market basket analysis," *Nutrition Health*, vol. 24, no. 2, pp. 83–92, Jun. 2018.

[30] T. Tiyasha, S. K. Bhagat, F. Fituma, T. M. Tung, S. Shahid, and Z. M. Yaseen, "Dual water choices: The assessment of the influential factors on water sources choices using unsupervised machine learning market basket analysis," *IEEE Access*, vol. 9, pp. 150532–150544, 2021, doi: 10.1109/ACCESS.2021.3124817.

[31] Y. Ali, A. Farooq, T. M. Alam, M. S. Farooq, M. J. Awan, and T. I. Baig, "Detection of schistosomiasis factors using association rule mining," *IEEE Access*, vol. 7, pp. 186108–186114, 2019, doi: 10.1109/ACCESS.2019.2956020.

[32] E. Hikmawati, N. U. Maulidevi, and K. Surendro, "Pruning strategy on adaptive rule model by sorting utility items," *IEEE Access*, vol. 10, pp. 91650–91662, 2022, doi: 10.1109/ACCESS.2022.3202307.

[33] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, S. Shahriar Shafin, and Md. Z. A. Bhuiyan, "A survey on behavioral pattern mining from sensor data in Internet of Things," *IEEE Access*, vol. 8, pp. 33318–33341, 2020, doi: 10.1109/ACCESS.2020.2974035.

[34] M. S. Mythili and A. R. Mohamed Shanavas, "Performance evaluation of apriori and FP-growth algorithms," *Int. J. Comput. Appl.*, vol. 79, no. 10, pp. 34–37, Oct. 2013.
[35] X. Shang, K.-U. Sattler, and I. Geist, "SQL based frequent pattern mining with FP-growth," in *Applications of Declarative Programming and Knowledge Management*. Cham, Switzerland: Springer, 2004, pp. 32–46.
[36] M. Yin, W. Wang, Y. Liu, and D. Jiang, "An improvement of FP-growth association rule mining algorithm based on adjacency table," in *Proc. MATEC Web Conf.*, 2018, p. 10012.
[37] C. Borgelt, "An implementation of the FP-growth algorithm," in *Proc. 1st Int. Workshop Open Source Data Mining: Frequent Pattern Mining Implementations*, Aug. 2005, pp. 1–5.
[38] S. S. Epp, *Discrete Mathematics With Applications*. Boston, MA, USA: Cengage learning, 2010.
[39] C. Jongsma, "Basic set theory and combinatorics," in *Introduction to Discrete Mathematics via Logic and Proof*. Cham, Switzerland: Springer, 2019, pp. 205–253.
[40] M. Hossain, A. H. M. S. Sattar, and M. K. Paul, "Market basket analysis using apriori and FP growth algorithm," in *Proc. 22nd Int. Conf. Comput. Inf. Technol. (ICCIT)*, Dec. 2019, pp. 1–6, doi: 10.1109/ICCIT48885.2019.9038197.
[41] A. Narayanan, *Oracle SQL Developer*. Birmingham, U.K.: Packt Publishing, 2016.
[42] B. Brumm, *Beginning Oracle SQL for Oracle Database 18c*. Cham, Switzerland: Springer, 2019.
[43] B. Siswanto, "Oracle DBMS scheduler package for data integrity test on web-based application," *J. Telecommun. Electron. Comput. Eng.*, vol. 12, no. 4, pp. 1–4, 2020.
[44] P. Fournier-Viger, J. C. Lin, B. Vo, T. T. Chi, J. Zhang, and H. B. Le, "A survey of itemset mining," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 7, no. 4, p. e1207, 2017.
[45] B. Goethals. (2004). *Frequent Itemset Mining Dataset Repository*. Accessed: Jan. 6, 2024. [Online]. Available: http://fimi.uantwerpen.be/data/
[46] P. F. Viger. (2022). *SPMF an Open-Source Data Mining Library*. Accessed: Jan. 6, 2024. [Online]. Available: https://www.philippe-fournier-viger.com/spmf/index.php

BOBY SISWANTO received the bachelor's degree in computer science from STMIK Indonesia Mandiri, Bandung, Indonesia, in 2006, and the master's degree in computer science from Telkom University, Bandung, in 2014. He is currently pursuing the Ph.D. degree in computer science with the Doctor of Computer Science Program, Bina Nusantara University. He is also a Faculty Member of Computer Science with the School of Computer Science Study Program, Bina Nusantara University. His teaching experience and research interests include databases, algorithm and programming, machine learning, the Internet of Things, and natural language processing.

HARYONO SOEPARNO received the bachelor's degree in statistics and computation from IPB University, Bogor, Indonesia, in 1980, the master's degree in computer science from Western Michigan University, MI, USA, in 1987, and the Ph.D. degree in computer science from the School of Engineering and Technology, Asian Institute of Technology, Bangkok, Thailand, in 1995. He has been an Associate Professor in computer science with the School of Computer Science, Bina Nusantara University, South Jakarta, Indonesia, since 1984. He was a member of the National Research Council, Ministry of Research, Technology, and Higher Education, Indonesia, from 2011 to 2019, and a reviewer of research and innovation funded by multiple donors. He is also the Head of the Concentration in Computer Science, Doctor of Computer Science Program, Binus University. He is the coauthor of more than three books on interdisciplinary research in computer science with various application domains. His teaching experience and research interests include databases, software engineering, analysis of algorithms, advanced knowledge systems, machine learning, deep learning, and natural language processing.

NESTI FRONIKA SIANIPAR received the bachelor's degree in agronomy from North Sumatera Islamic University, Medan, Indonesia, in 1995, and the master's degree in food science and the Ph.D. degree in biotechnology from Bogor Agricultural University, Bogor, Indonesia, in 1998 and 2008, respectively.

She is currently an Associate Professor and the Head of the Research Center of Food Biotechnology, Bina Nusantara University. She is heavily involved in multidisciplinary research related to food, health, and technology. She is the author of several books about typhonium flagelliforme for health. Her teaching experience and research interests include computational biology, research methodology, and biotechnology. Her research that has received a patent are typhonium flagelliforme as a cancer drug and genetic engineering on banana plantain.

WIDODO BUDIHARTO received the bachelor's degree in physics from the University of Indonesia, South Jakarta, Indonesia, the master's degree in information technology from STT Benarif, South Jakarta, and the Ph.D. degree in electrical engineering from the Institute of Technology Sepuluh Nopember, Surabaya, Indonesia. He took the Ph.D. Sandwich Program in robotics with Kumamoto University, Japan, and conducted his postdoctoral research in robotics and artificial intelligence with Hosei University, Japan, where he was a Visiting Professor with the Erasmus Mundus French Indonesian Consortium (FICEM), France, and Erasmus Mundus Scholar with EU Universite de Bourgogne, France, in 2007, and in 2017 and 2016, respectively. He is currently a Professor in artificial intelligence with the School of Computer Science, Bina Nusantara University, South Jakarta. His research interests include intelligent systems, data science, robotic vision, and computational intelligence.