Düzce University
Journal of Science & Technology

Research Article

Performance Comparison of Association Rule Algorithms with SPMF on Automotive Industry Data

Melih NAIR[a,*], Fatih KAYAALP[b]

[a] *Department of Computer Engineering, Institute of Natural Sciences, Düzce University, Düzce, TURKEY*
[b] *Department of Computer Engineering, Faculty of Engineering, Düzce University, Düzce, TURKEY*
* *Corresponding author's e-mail address: melihnair@gmail.com*
DOI : 10.29130/dubited.581931

ABSTRACT

By the recent developments about the information technologies, companies can store their data faster and easier with lower costs. All transactions (sales, current card, invoicing, etc.) performed in companies during the day combine at the end of the day to form big datasets. It is possible to extract valuable information through these datasets with data mining. And this has become more important for companies in terms of today's conditions where the competition in the market is high. In this study, a dataset of a company selling car maintenance and repair products in Turkey is used. Association Rules are applied on this dataset for determining the items which are bought together by the customers. These rules, which are calculated specifically for the company, can be used to redefine the sales and marketing strategies, to revise the storage areas efficiently, and to create sales campaigns suitable for the customers and regions. These algorithms are also called Frequent Itemset Mining Algorithms. The most recent 11 algorithms from these are applied to this dataset in order to compare the performances according to metrics like memory usage and execution times against varying support values and varying record numbers by using SPMF platform. Three different datasets are created by using the whole dataset like 6-months, 12-months and 22-months. According to the experiments, it can be said that executon times generally increases inversely with the support values as nearly all algorithms have higher execution time values for the lowest support value of 0.1. dEclat_bitset algorithm has the most efficient performance for 6-months and 12-months dataset. But Eclat algorithm can be said to be the most efficient algorithm for 0.7 and 0.3 support values; on the other hand dEclat_bitset is the most efficient algorithm for 0.3 and 0.1 support values on 22-months dataset.

Keywords: *Association Rules, Market Basket Analysis, SPMF.*

Otomotiv Endüstrisi Verileri Üzerinde Birliktelik Kuralları Algoritmalarının SPMF ile Performans Karşılaştırması

ÖZET

Bilgi teknolojilerindeki son gelişmeler sayesinde, şirketler verilerini daha düşük maliyetlerle daha hızlı ve daha kolay saklayabilirler. Gün içinde şirketlerde gerçekleştirilen tüm işlemler (satışlar, cari kartlar, faturalama vb.), günün sonunda birleştirilir ve büyük veri setleri oluştururlar. Bu veri setlerinden veri madenciliği aracılıyla değerli

Received: 25/06/2019, Revised: 04/07/2019, Accepted: 06/07/2019

bilgiler elde edilmesi mümkündür. Pazardaki rekabetin yüksek olduğu günümüz şartları açısından bu durum şirketler için çok daha önemli hale gelmiştir. Bu çalışmada Türkiye'de araç bakım ve servis ürünleri satan bir şirketin veriseti kullanılmıştır. Bu verisetine, müşteriler tarafından birlikte satın alınmış olan ürünlerin tespiti için Birliktelik Kuralları uygulanmıştır. Şirketlere özgü olarak çıkarımı yapılan bu kurallar şirketlerin satış ve pazarlama stratejilerinin belirlenmesinde, depoların verimli bir şekilde kullanımlarında ve müşteriler ya da bölgelere göre uygun satış kampanyaları oluşturulmasında kullanılabilir. Birliktelik kuralları aynı zamanda Sık Satılan Ürün Algoritmaları olarak da isimlendirilebilmektedir. Bu algoritmalardan en güncel 11 tanesi SPMF yazılımı kullanılarak bu veri setine uygulanmış ve bu algoritmaların değişken destek değerleri ve değişken kayıt sayılarına bağlı olarak performansları, bellek kullanım miktarları ve işlem süreleri açısından karşılaştırılmıştır. Başlangıçtaki veri seti, 6 aylık, 12 aylık ve 22 aylık kayıt içerecek şekilde 3 ayrı veri seti haline getirilmiştir. Deney sonuçlarına bakıldığında, işlem zamanlarının genellikle destek değerleriyle ters orantılı olarak arttığı söylenebilir. Çünkü neredeyse tüm algoritmaların en düşük destek değeri olan 0,1 için daha yüksek işlem zamanı değerlerine sahip oldukları görülmüştür. 6 aylık ve 12 aylık veri setleri için dEclat_bitset algoritması en verimli performansı göstermiştir. Fakat 22 aylık veri setinde, 0,7 ve 0,3 destek değerleri için Eclat algoritması en verimli olarak görünürken; 0,3 ve 0,1 destek değerleri için dEclat_bitset algoritması en verimli olarak görünmektedir.

Anahtar Kelimeler: *Birliktelik Kuralları, Market Sepeti Analizi, SPMF.*

I. INTRODUCTION

As information technologies are commonly used in all domains from telecommunications to marketing today, all data related to sales and customers can be easily stored and this creates huge datasets. However, stored data sets should be analysed in order to contribute to customer and vendor relationships, company management and sales strategies. Therefore, it has become so important for companies.

Nowadays, with the development of information technologies, gathering and storing large amounts of data is both less costly and faster than compared to the past. On the other hand, analyzing this large size data is a very difficult task. As processing this large datasets is so difficult, efficient use of many computer technologies are needed. Identifying the models, similarities and abnormalities in the datasets and simplification of these determinations are one of the most important issues of the information age[1]. At this stage, data mining is involved.

Essentially, data mining can be defined as the process of extracting useful information through the datasets by using various algorithms and software platforms. Data Mining has emerged in the 90s. Especially after 2000, the number of studies in the field of data mining has increased a lot. Over the past three years, thousands of studies on data mining have been presented. Day by day, data mining and information discovery in databases attract the attention of industry. These industries often serve in the research, marketing and media sectors.

Especially marketing companies aims to increase their sales by analyzing product sales data or Customer Relations Management (CRM) data with different data mining algorithms. One of these algorithms is Association Rules. Association Rules algorithms try to find hidden relationships between different product sales data for identifying the products which are sold together. Once these products are discovered, the companies try to use this information for creating more efficient sales strategies.

This article presents an overview of this field and it also explains the relationships between related fields such as machine learning and statistics[2].

II. Related Works

Some of the studies related to association rules in literature are given below.

Erpolat presents a study to evaluate the performances of Apriori and FP-Growth, which are the two of popular association rules algorithms, for identifying the shopping habits of customers to increase the sales of the company by using sales data of an automotive service [3].

Bala and friends present a study evaluating Apriori and Fp-Growth algorithms on the Super Market and voter datasets with a total of 4627 records with 217 attributes, different support values (20%, 50%, 60%) and different confidence values (30%, 40%, 50%, 60%, 70%, 80%) by using WEKA. Then, the run times of both algorithms were compared [4].

Erduran presents a study in 2017, examining online customer complaints related to banking domain with data mining methods. 100,000 customer complaints on the dataset are divided into groups according to the words. Afterwards, inferences were made according to the common words, similar usage and use of the words mentioned in the complaint texts [5].

The aim of the study presented by Aguwa and friends in 2017 is to support the decision-making processes of enterprises. Fuzzy logic was used in the study. In addition, text mining and association rule algorithms were applied to the voice data of the customers. Customer satisfaction index was created with the results[6].

The study presented by Griva and friends in 2018 used the sales data of a store. Data mining techniques like clustering and association rules were applied on sales data. Customers are categorized according to their previous visits to the store. In contrast to similar studies, the study takes into account the products that a customer receives during multiple visits, not the products he has received during a specific visit. [7].

Organizing marketing campaigns of mobile telephone networks was aimed in Boix and Moreno's study in 2018. They proposed a data mining model to select important variables related to customer usage details and perform customer churn analysis, which is necessary to create the appropriate campaigns[8]. Sales data of the customers of an insurance company are analyzed with the K-means algorithm in [9]. With the help of the results obtained from this analysis, identification of the characteristics of similar customers to develop new appropriate marketing strategies are aimed.

The aim of the study presented by Bardak and friends in 2018 is determining the factors affecting the choice of traditional stores or virtual stores in furniture purchase. Predictive Apriori Algorithm was used in the study. The data set used in the study was collected by a questionnaire with 217 participants. As a result of the study, it is reported that people prefer traditional shopping instead of virtual stores. Besides, the customers who are married and have children prefer virtual stores more than other customers[10]. A novel technique that has the ability to mine interesting negative association rules between items in the transactions dataset, by automatically extracting knowledge from that dataset based on the purchased quantities is presented in [11].

In 2019, Bakariya and Thakur have proposed an algorithm called the Mining for Weblog (IIMW), which was based on the Apriori-Rare and Apriori-Inverse algorithms. Afterwards they applied three algorithms on dataset obtained from web traffic archives and compared the performance values[12].

It is seen that using association rules to identify the relationships between products on different domains is so popular as that gives companies the ability to increase their sales and income, efficiently organizing their storage areas by locating the products sold together close to each other.

III. MATERIALS AND METHOD

A. DATA MINING

Nowadays, information systems are indispensable for many organizations because they provide abilities in decision making processes by storing, processing and analyzing large amounts of data[13]. Data mining, with its simplest definition, can be defined as obtaining meaningful information from existing data. Data science is multidisciplinary and consists of many components. Figure 1 shows the intersections of data mining with different disciplines.

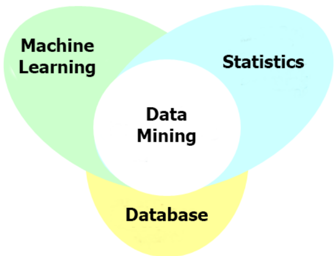

Figure 1. *Intersection of data mining with different disciplines*

Data mining combines tools such as statistics, database management, artificial intelligence, data visualization, and reporting to analyze data sets.

B. ASSOCIATION RULES

Unsupervied learning is a data mining method that does not require a labelled dataset and does not require a domain expert. Association Rules or Association Rule Mining (ARM) is one of the unsupervied learning methods used in data mining. It is a good way to explore relationships between data, especially in large databases. The goal is to determine the strong rules discovered in databases using different inference criterions[20].

Mining rules is an important tool for data mining. Nowadays, mining rules play an important role, especially in the field of artificial intelligence, information retrieval, data science, mold creation in biological databases, clinical databases, web personalization, text mining, statistics, telecommunications, marketing, risk management and inventory control.
The determination of the association rules is based on the confidence and support values. Rules that provide both minimum support and minimum confidence are called strong rules. Generally, support and confidence values are used to identify products that users are interested in. Thus, rules that do not help users to recognize the rules are separated from the useful rules[21].

B. ASSOCIATION RULES MINING ALGORITHMS

These algorithms are used to determine the association rules on the database. In the literature, there are many algorithms developed in order to determine the association rules. In this section, we mention the frequently used association rule mining algorithms.

1. Apriori

Apriori was proposed by Agrawal [7]. The Apriori algorithm successfully finds common elements in the database. However, as the number of items increases, so does the size of the database:

- The space required for the search and I/O cost will increase.
- As the number of database scans increases, the cost of candidate cluster calculation will also increase[24].

2. Fp-Growth

The FP-Growth method was proposed by Han[25] and called the FP-tree (Frequent Pattern tree). The FP-tree is a compact representation of all relevant frequency information in a database[26].

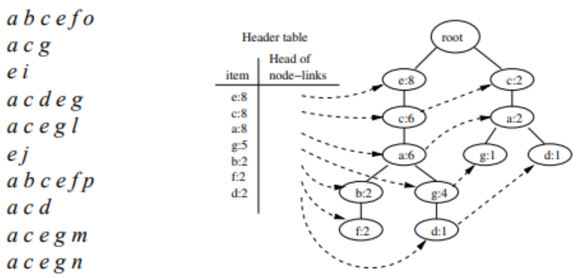

Figure 2. *An Example FP-tree (minsup=20%)*

3. Eclat

Eclat uses a vertical database layout. In the Eclat algorithm, an intersectional approach is used to calculate the support value of an element set. The Eclat algorithm is given in Fig 3.

Input: $\mathcal{D}, \sigma, I \subseteq \mathcal{I}$
Output: $\mathcal{F}[I](\mathcal{D}, \sigma)$
$\mathcal{F}[I] := \{\}$
for all $i \in \mathcal{I}$ occurring in $\mathcal{D}$ **do**
  $\mathcal{F}[I] := \mathcal{F}[I] \cup \{I \cup \{i\}\}$
  // Create $\mathcal{D}^i$
  $\mathcal{D}^i := \{\}$
  for all $j \in \mathcal{I}$ occurring in $\mathcal{D}$ such that $j > i$ **do**
    $C := cover(\{i\}) \cap cover(\{j\})$
    if $|C| \geq \sigma$ **then**
      $\mathcal{D}^i := \mathcal{D}^i \cup \{(j, C)\}$
    end if
  end for
  // Depth-first recursion
  Compute $\mathcal{F}[I \cup \{i\}](\mathcal{D}^i, \sigma)$
  $\mathcal{F}[I] := \mathcal{F}[I] \cup \mathcal{F}[I \cup \{i\}]$
end for

Figure 3. *Pseudo-code for Eclat Algorithm*

As a comparison, Eclat essentially generates candidate itemsets using only the join step from Apriori, since the itemsets necessary for the prune step are not available. [27].

4. dEclat

Figure 4 shows the pseudo-code for dEclat, which performs a pure bottom-up search of the subset tree. As such it is not suitable for very long pattern mining, but our experiments show that diffsets allow it to mine on much lower supports than other bottom up methods like Apriori and the base Eclat method. [28].

DiffEclat($[P]$)**:**
for all $X_i \in [P]$ **do**
  for all $X_j \in [P]$, with $j > i$ **do**
    $R = X_i \cup X_j$;
    $d(R) = d(X_j) - d(X_i)$;
    if $\sigma(R) \geq min\_sup$ **then**
      $T_i = T_i \cup \{R\}$; //T_i initially empty
for all $T_i \neq \emptyset$ **do** DiffEclat(T_i);

Figure 4. *Pseudo-code for dEclat*

5. dCharm

Figure 5 shows the pseudo-code for dCharm, which performs a novel search for closed sets using subset properties of diffsets. As in dEclat all differences for pairs of elements are computed [28].

DiffCharm (P):
  for all $X_i \in [P]$
    $\mathbf{X} = X_i$
    for all $X_j \in [P]$ with $j > i$
      $R = \mathbf{X} \cup X_j$ and $d(R) = d(X_j) - d(X_i)$
      if $d(X_i) = d(X_j)$ **then** Remove X_j from Nodes; Replace all X_i with R
      if $d(X_i) \supset d(X_j)$ **then** Replace all X_i with R
      if $d(X_i) \subset d(X_j)$ **then** Remove X_j from Nodes; Add R to $NewN$
      if $d(X_i) \neq d(X_j)$ **then** Add R to $NewN$
    if NewN $\neq \emptyset$ **then** DiffCharm (NewN)

Figure 5. *Pseudo-code for dCharm*

6. Apriori-TID

M. Adda et al proposed a framework in [29], a generic framework was presented to mine patterns based on the Apriori approach. [30].

On this basis, a new algorithm called Apriori-TID was developed by Agrawal. The main difference from Apriori is that it does not use the database for counting support after the first cycle. Instead, the candidate uses a encoding of the product sets used in the previous cycle shown in k[31].

7. A-Close

The first algorithm proposed for mining closed itemsets was A-Close[32]. A-Close first browses level-wise the frequent itemsets lattice by means of an Apriori-like strategy, and mines all the minimal elements of each equivalence class [33].

```
Procedure FPclose(T, C)
Input: T, an FP-tree
       C, the CFI-tree for T.base
Output: Updated C
Method:
1. if T only contains a single branch B
2.   generate all CFI's from B;
3. for each CFI X generated
4.   if not closed_checking(X, C)
5.     insert X into C;
6. else for each i in T.header do begin
7.   set Y = T.base ∪ {i};
8.   if not closed_checking(Y, C)
9.     if T.FP-array is defined
10.      let tail be the set of frequent items for i in T.FP-array
11.    else
12.      let tail be the set of frequent items in i's conditional pattern base;
13.    sort tail in decreasing order of items' counts;
14.    construct the FP-tree T_Y and possibly its FP-array A_Y;
15.    initialize Y's conditional CFI-tree C_Y;
16.    call FPclose(T_Y, C_Y);
17.    merge C_Y with C;
18. end
```

Figure 6. *Pseudo-code for dCharm*

Fig. 6 gives algorithm FPclose. Before calling FPclose with some (T,C), they already know from line 8 that there is no existing CFI X such that 1) T.base □ X and 2) T.base and X have the same count. If there is only one single branch in T, the nodes and their counts in this single branch can be easily used to list the T.base-local closed frequent itemsets [34].

8. Bitset Table

In this approach there is only one data structure: bitset table. The bitset table stores all transactions with frequent items in bitset representation.

Following are the steps to build bitset table.

I) Scan the dataset once to get frequency of each item. Remove all the items whose frequency is lower than min-support. Generate a list of frequent items by sorting remaining items in frequency ascending order.

2) Scan the dataset again and transform each transaction to vertical bitset representation [35].

D. DATA SET

In this study, a data set of a company selling car maintenance and repair products was used. Fruqently used 11 association rule mining algorithms were applied to this data set. Three different data sets are created including records of 6 months (167,334 rows), 12 months (203,753 rows) and 22 months (543,316 rows) respectively. Each dataset has 33 features.

E. PRE-PROCESSING

At this stage of the study, data cleanup, data integration and data transformation preprocessing steps were used.

- Data cleanup: errors and inconsistencies in the data set have been fixed. The quality of the data is improved.
- Data integration: data were obtained from a single source.
- Data transformation: conversion of data into appropriate format for data mining.

In this study, SPMF program was preferred. SPMF is an open-source data mining library written in Java, specialized in pattern mining (the discovery of patterns in data) (http://www.philippe-fournier-viger.com/spmf/index.php).

The source code of each algorithm can be easily integrated in other Java software.

Moreover, SPMF can be used as a standalone program with a simple user interface or from the command line. SPMF is fast and lightweight (no dependencies to other libraries).

The original data sets (6 months, 12 months and 22 months) extension were xlsx. As SMPF does not accept xlsx, the data set was converted to arff extension (Fig 7.).

```
@Relation YASAR_PETROL
@attribute 'MOTOR YAG'  { t}
@attribute 'FILTRE'  { t}
@attribute 'AKU'  { t}
@attribute 'FREN HIDROLIK'  { t}
@attribute 'OTO BAKIM'  { t}
@attribute 'FREN BALATASI'  { t}
@attribute 'ANTIFRIZ'  { t}
@attribute 'CAM SUYU '  { t}
@attribute 'POMPA'  { t}
@attribute 'ANAHTAR'  { t}
@attribute 'BOR YAG'  { t}
@attribute 'EKIPMAN'  { t}
@attribute 'ELEKTRKMT'  { t}
@attribute 'GAZ MOTOR'  { t}
@attribute 'GEMI MOTOR'  { t}
@attribute 'GRES YAG'  { t}
@attribute 'ISI TRANS'  { t}
@attribute 'KIZAK YAG'  { t}
@attribute 'KALIP YAG'  { t}
@attribute 'KOMPRESSOR'  { t}
@attribute 'LOKMA'  { t}
@attribute 'MAKAS'  { t}
@attribute 'MIKSER'  { t}
@attribute 'PENSE'  { t}
@attribute 'SAF SU'  { t}
@attribute 'SILECEK'  { t}
@attribute 'SIRKILASYON'  { t}
@attribute 'STAND'  { t}
@attribute 'TAKIM DOLAP'  { t}
@attribute 'TORNAVIDA'  { t}
@attribute 'TURBIN'  { t}
@attribute 'USTUBU'  { t}
@attribute 'YANSANAYI'  { t}
@DATA
t,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
t,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
?,?,?,?,t,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
t,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
t,?,?,?,t,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,
```

Figure 7. *Arff file with thirty-three features*

F. METRIC OF ASSOCIATION RULES

Generally two criteria are talked in measuring Association rules which are Support and Confidence.

1) Support: The rule X⇒Y holds with support s if s% of transactions in D contain X∪Y and calculated as shown in Table 1. Rules having a s greater than a user-specified support is said to have minimum support.

Table 1: *Support Example*

ID	Items Sold	Support=Occurrence/Total Support
1	XYZ	Total Support=5
2	XYZK	Support of {XY}=2/5=40%
3	YZ	Support of {YZ}=4/5=80%
4	XZ	
5	MYZK	

2) Confidence: The rule X⇒Y holds with confidence c if c% of the transactions in D that contain X also contain Y and calculated as shown in Table 2. Rules having a c greater than a user-specified confidence is said to have minimum confidence.

Table 2: *Confidence Example*

ID	Items Sold	Confidence=Occurrence(Y)/Occurrence (X)
1	XYZ	Confidence {X⇒Y}=3/4=75%
2	XYZK	Confidence {Y⇒Z}=4/5=80%
3	YZ	
4	XZ	
5	MYZK	

IV. EXPERIMENTAL RESULTS

At this stage of the study, performance values of frequent mining algorithms were compared. For comparison, four different data sets and three different support values were used. The execution times of the algorithms and the amount of memory usage are graphically compared. Now let's take a look at these graphics:

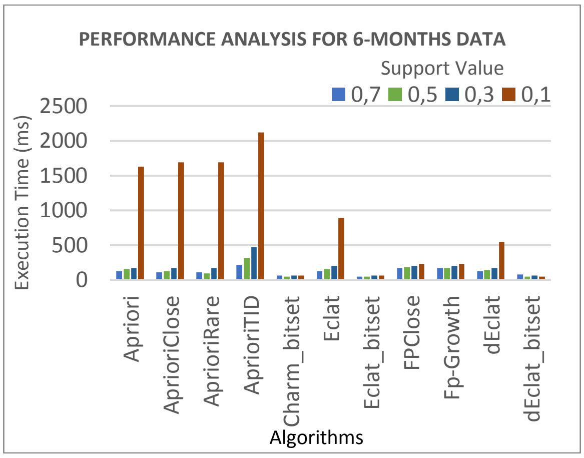

Figure 8. *Run time graph for four different support values of eleven algorithms for 6-mount data.*

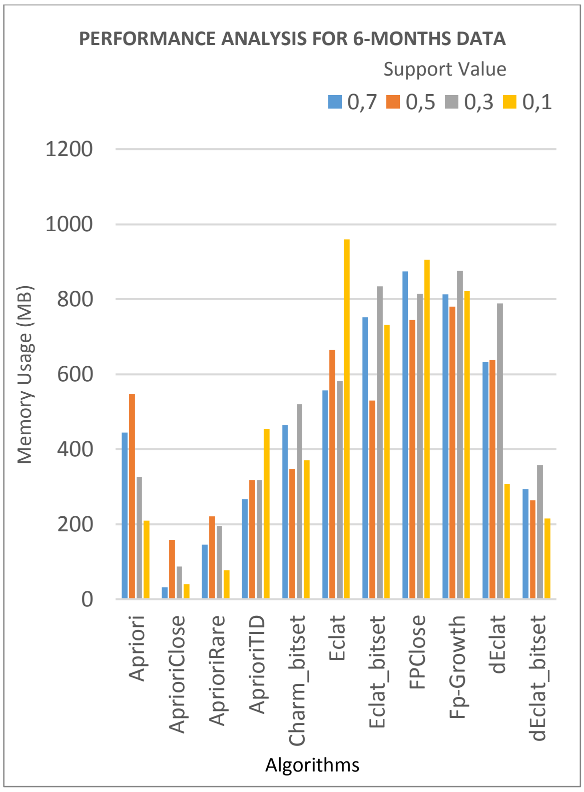

Figure 9. *Memory usage graph for four different support values of eleven algorithms for 6-months data.*

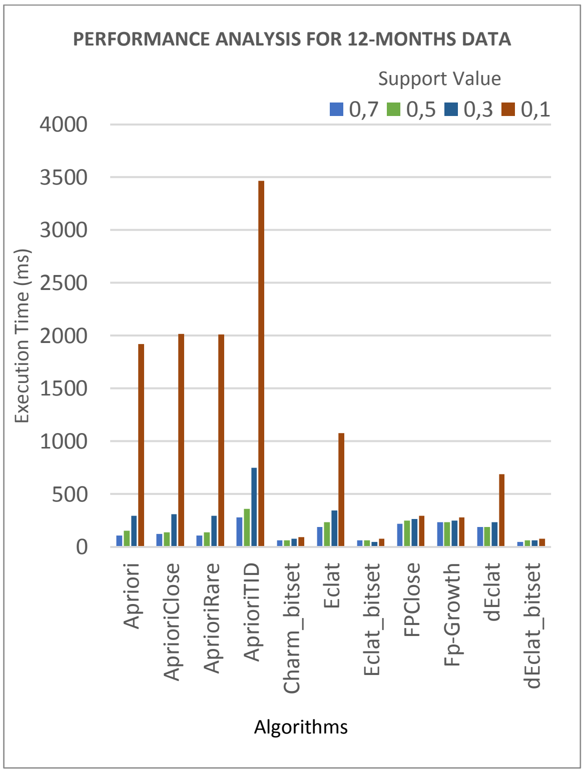

Figure 10. *Run time graph for four different support values of eleven algorithms for 12-months data.*

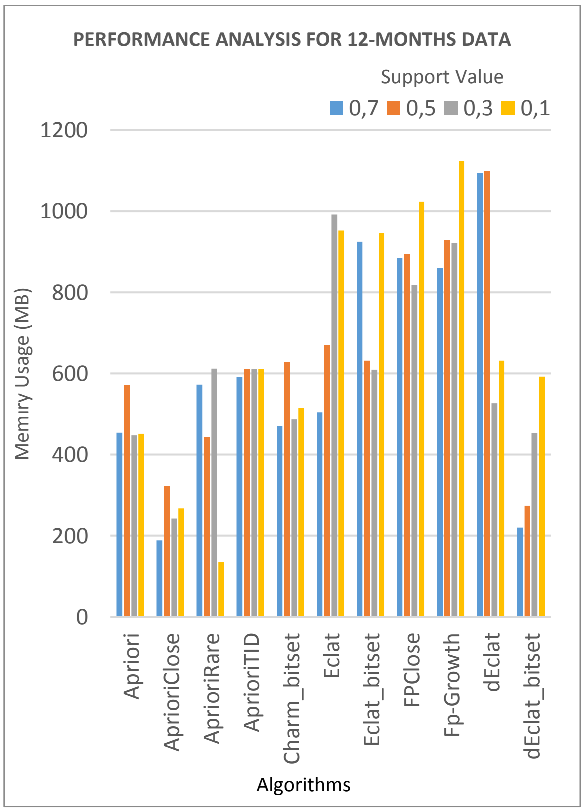

Figure 11. *Memory usage graph for four different support values of eleven algorithms for 12-months data.*

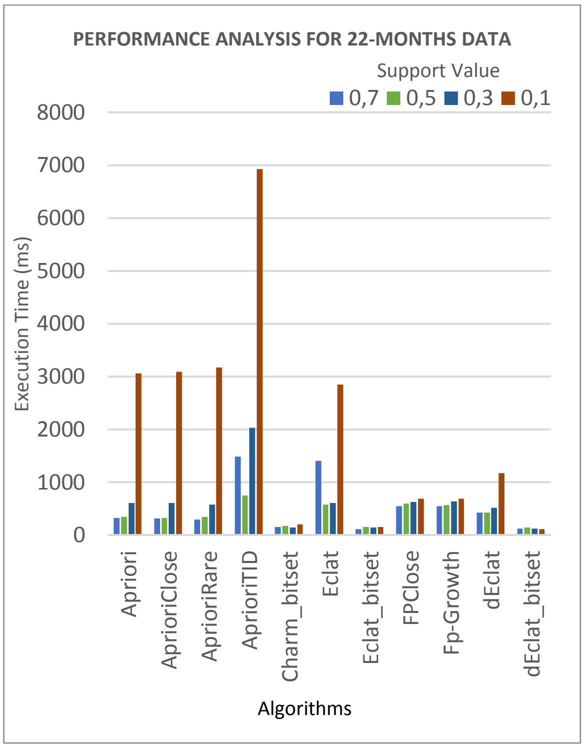

Figure 12. *Run time graph for four different support values of eleven algorithms for 22-months data.*

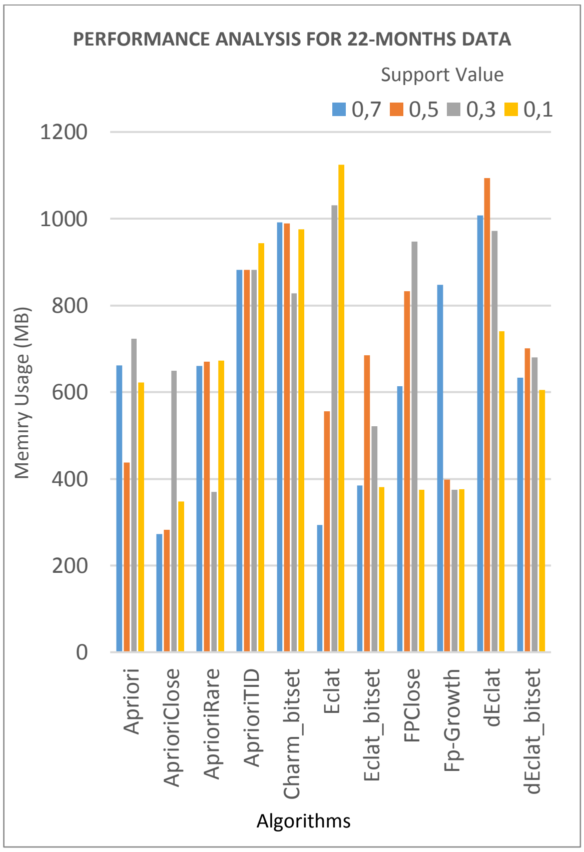

Figure 13. *Memory usage graph for four different support values of eleven algorithms for 22-months data.*

V. DISCUSSION

According to Figure 8, Charm_bitset, Eclat_bitset and dEclat_bitset have approximately the same execution times values and faster than others. But according to Figure 9, the memory usages of these three algorithms are not similar and dEclat_bitset has lower memory usage values than Charm_bitset and Eclat_bitset. When Figure 8 and Figure 9 are examined together, dEclat_bitset algorithm can be said to be the most efficient algorithm for all support values.

According to Figure 10, Charm_bitset, Eclat_bitset and dEclat_bitset have approximately same execution times values and faster than others. But according to Figure 11, the memory usages of these three algorithms are not similar and dEclat_bitset has lower memory usage values than Charm_bitset and Eclat_bitset. When Figure 10 and Figure 11 are examined together, dEclat_bitset algorithm can be said to be the most efficient algorithm for all support values.

According to Figure 12, Charm_bitset, Eclat_bitset and dEclat_bitset have approximately same execution times values and faster than others. But according to Figure 13, the memory usages of these three algorithms are not similar and Eclat has lower memory usage values than Charm_bitset and dEclat_bitset for support values of 0.7 and 0.5. But dEclat_bitset has lower memory usages for 0.3 and 0.1 support values. When Figure 12 and Figure 13 are examined together, Eclat algorithm can be said to be the most efficient algorithm for 0.7 and 0.3 support values; on the other hand dEclat_bitset is the most efficient algorithm for 0.3 and 0.1 support values.

When the execution time values in Figure 8, Figure 10 and Figure 12 are examined together; it can be said that executon times generally increases inversely with the support values as nearly all algorithms have higher execution time values for the lowest support value of 0.1. This may be the result of creating too many rules generated from the dataset.

When memory usage values in Figure 9, Figure 11 and Figure 13 are examined together, a linear relation between memory usage and support values couldn’t be seen as some algorithms have inversely proportional and some have directly proportional values.

As given before, the dataset used in the experiments was divided into 3 parts called 6-months (including 167,334 rows), 12-months (including 203,753 rows) and 22-months (including 543,316 rows) in order to observe how the number or rows effects execution times and memory usage values. 12-months dataset’s row number is 1.217 times of 6-months dataset row number and 22-months dataset’s row number is 2.66 times of 12-months dataset. When Figure 8, Figure 10 and Figure 12 are examined in the light of row numbers, Apriori, AprioriClose, AprioriRare, AprioriTID and Eclat algorithms have taken more times to complete the experiments only for support value of 0.1. Except this support value, a significant difference is not seen on the result graphics both for specified 5 algorithms and the others. When Figure 9, Figure 11 and Figure 13 are examined together in the light of row numbers, a significant difference on the memory usage values both for algorithms types and support values couldn’t be seen.

VI. CONCLUSION

Association Rule algorithms are generally used to to discover the items that are bought together by the customers on a specific dataset. Algorithms used for this aim are called Frequent Itemset Mining algorithms. The most recent 11 algorithms from these are applied to a dataset of a company which is selling car maintenance and repair products in Turkey in order to evaluate the performances by using execution times and memory usage metrics against varying datasize and support values.

As a result, dEclat_bitset algorithm has the most efficient performance for 6-months and 12-months dataset. But Eclat algorithm can be said to be the most efficient algorithm for 0.7 and 0.3 support values; on the other hand dEclat_bitset is the most efficient algorithm for 0.3 and 0.1 support values on 22-months dataset.

A linear relation between memory usage and support values couldn't be seen on graphics. It can be said that executon times generally increases inversely with the support values as nearly all algorithms have higher execution time values for the lowest support value of 0.1. In the light of row numbers of datasets, a significant difference on the memory usage values both for algorithms types and support values couldn't be seen.

VII. REFERENCES

[1] V.J.E.U.R. Gancheva, "Market basket analysis of beauty products". 2013.

[2] U.Fayyad, G. Piatetsky Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases". *Ai Magazine*, vol. 17, no. 3, .pp. 37-54, 1996.

[3] S. Erpolat, "Otomobil Yetkili Servislerinde Birliktelik Kurallarının Belirlenmesinde Apriori ve FP-Growth Algoritmalarının Karşılaştırılması," 2012.

[4] A. Bala et al., "Performance Analysis of Apriori and FP-Growth Algorithms (Association Rule Mining)," vol. 7, no. 2, pp. 279-293, 2016.

[5] G. Yıldız Erduran, "Online müşteri şikayetlerinin veri madenciliği ile incelenmesi," 2017.

[6] C. Aguwa, M. H. Olya, and L. Monplaisir, "Modeling of fuzzy-based voice of customer for business decision analytics," *Knowledge-Based Systems*, 2017, no. 125, pp. 136-145.

[7] A. Griva et al., Retail business analytics: "Customer visit segmentation using market basket data," *Expert Systems with Applications*, 2018, no. 100, pp. 1-16.

[8] M. Postigo-Boix, and J. L. Melus-Moreno, "A social model based on customers' profiles for analyzing the churning process in the mobile market of data plans," *Physica a-Statistical Mechanics and Its Applications*, no. 496, pp. 571-592, 2018.

[9] B. Doğan et al., “Sigortacılık Sektöründe Müşteri İlişki Yönetimi İçin Kümeleme Analizi,” 2018, ss. 11-18.

[10] T. Bardak et al., “Mobilya Alımında Geleneksel Mağaza ile Sanal Mağaza Tercihinin Veri Madenciliğine Dayalı Analizi,” 2018, 6, ss. 25-27.

[11] T. Dökeroğlu, , Z.M.M. Malik, ve A.-S.J.D.Ü.B.v.T.D. Shadi, “Gözetimsiz Makine Öğrenme Teknikleri ile Miktara Dayalı Negatif Birliktelik Kural Madenciliği,” ss. 1119-1138.

[12] B. Bakariya, G. Thakur, and K. J. I. A .J. O. I. T. Chaturvedi, “An Efficient Algorithm for Extracting Infrequent Itemsets from Weblog,” vol. 16, no. 2, pp. 275-280, 2019.

[13] A. Morais et al., “Predicting the need of Neonatal Resuscitation using Data Mining,” *8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (Euspn 2017) / 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (Icth-2017) / Affiliated Workshops*, no. 113, pp. 571-576, 2017.

[14] B. Anandan, and C. Clifton, “Differentially Private Feature Selection for Data Mining,” *Iwspa '18: Proceedings of the Fourth Acm International Workshop on Security and Privacy Analytics*, 2018, pp. 43-53.

[15] A. Stokes et al., “Long Term Lead Survival in Adult Congenital Heart Disease Patients: A Retrospective Analysis Using Clinical Correspondence Data Mining,” *Heart*, 2018, no. 104, pp. A22-A23.

[16] A. Idri et al., “A systematic map of medical data preprocessing in knowledge discovery,” *Computer Methods and Programs in Biomedicine*, no. 162, pp. 69-85,2018.

[17] I. Ionita, and L. Ionita, “Classification Algorithms of Data Mining Applied for Demographic Processes,” *Brain-Broad Research in Artificial Intelligence and Neuroscience*, vol. 9, no. 1, pp. 94-100, 2018.

[18] R. Tibshirani, and J. Friedman, “The elements of statistical learning: data mining, inference, and prediction,” *Springer Heidelberg*, 2001.

[19] A. Berson, S. Smith, and K. Thearling, “Building Data Mining Applications for CRM McGraw-Hill,” 1999.

[20] J. Arora et al., “A review on association rule mining algorithms,” vol. 1, no.5, 2013.

[21] F. H. AL-Zawaidah, et al., “An improved algorithm for mining association rules in large databases,” vol. 1, no. 7, pp. 311-316, 2011.

[22] R. Agrawal, T. Imielinski, and A. Swami. “Mining associations between sets of items in large databases,” *In Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1993.

[23] A. Geyer-Schulz, and M. Hahsler. "Evaluation of recommender algorithms for an internet information broker based on simple association rules and on the repeat-buying theory," *In proceedings WEBKDD*, 2002.

[24] Kaur, C.J.I.J.o.A.R.i.C.E. and Technology, "Association rule mining using apriori algorithm: a survey," 2013, no. 2(6).

[25] J. Han, J. Pei, and Y. Yin. "Mining frequent patterns without candidate generation," *In ACM sigmod record*, 2000.

[26] Grahne, G. and J. Zhu. "Efficiently using prefix-trees in mining frequent itemsets," *In FIMI*, 2003.

[27] Goethals, B.J.U.o.H., "Survey on frequent pattern mining," no. 19, pp. 840-852,2003.

[28] M. J. Zaki, and K. Gouda. "Fast vertical mining using diffsets," *In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003.

[29] M. Adda, L. Wu, and Y. Feng. "Rare itemset mining in Sixth International Conference on Machine Learning and Applications (ICMLA 2007)," *IEEE*, 2007.

[30] J. Pillai, and O.J.I.J.o.C.A. Vyas, "Overview of itemset utility mining and its applications," vol. 5, no.11, pp. 9-13. .

[31] R. Agrawalet al., "Fast discovery of association rules," vol. 12, no. 1, pp. 307-328,1996.

[32] N. Pasquier et al. "Discovering frequent closed itemsets for association rules in International Conference on Database Theory," *Springer*, 1999.

[33] C. Lucchese, S. Orlando, and R. Perego. "DCI Closed: A Fast and Memory Efficient Algorithm to Mine Frequent Closed Itemsets," *FIMI*, 2004.

[34] G. Grahne, J.J.I.t.o.k. Zhu, and D. engineering, "Fast algorithms for frequent itemset mining using fp-trees,", vol. 17, no.10, pp. 1347-1362, 2005.

[35] N. Dwivedi, and S. R. Satti. "Set and array based hybrid data structure solution for Frequent Pattern Mining," *in 2015 Tenth International Conference on Digital Information Management (ICDIM)*, 2015.