CHAPTER - 5

RULE BASED PREDICTION

5.1 Introduction

Many companies and academics use data mining to retrieve crucial information, find hidden patterns in datasets, and enhance decision-making. The technique of association rule mining (ARM) is a type of data mining that identifies frequent itemsets from relational and transactional databases, revealing hidden relationships among them. ARM aims to discover associations between frequently co-occurring data objects in a transaction set, allowing researchers to predict the occurrence of one itemset based on the presence of another object in the transaction. Several apps have used the ARM approach extensively [172] [173]

In essence, association rule mining involves two primary sub-tasks [174]: firstly, generating frequent itemsets by identifying sets of items that meet a minimum support threshold, and secondly, generating association rules expressed as A→ B, where A and B denote itemsets, by using the resulting frequent itemsets and satisfying a minimum confidence threshold. Since frequent itemset mining typically demands a lot of resources and computation time, most research has concentrated on increasing its effectiveness.

Three basic categories can be used to classify the rule-based categorization approach used in machine learning: tree-based classification models [175], rule-based learning [176], and association-based classification (AC) [177]. AC is a fusion of classification and ARM methods to achieve its objectives. The relationship between elements in a dataset is determined using association rule mining. A learning-labelled dataset is used in classification to anticipate the class label for each event. Assuring exact lowest support and confidence levels, the main focus of Associative Classification (AC) is on

generating Class Association Rules (CARs). CARs are expressed as x → c, where x represents a set of attribute scores, and c is the class label.

To increase classification accuracy, reduce classifier size, and adapt to various types of data sets in many disciplines, most AC algorithms have improved the metrics, algorithms, and methodologies used in rule generation, evaluation, and classification [179]. The support and confidence metrics of static association rules are the focus of most modern AC algorithms. A crucial parameter in Associative Classification is the minimum support threshold, which is used to choose frequent rule items. Frequent rule items are then filtered based on the minimum confidence criterion. However, this can result in a huge number of frequently emerging rule items. Performing an exhaustive search for realizing rules in classifiers can be difficult, especially when dealing with big datasets or small minimum support values. Candidate CAR generation takes a while and uses a lot of memory. Even the simplest candidate generation process is challenging since it depends on training time and memory usage [180].

This chapter explains a proficient forecasting algorithm using generated rules from the data after clustering via AC. The proposed approach contains three main steps. The first step is preprocessing, which converts numerical and categorical attributes using discretization. This step is used only if the dataset is a numerical or mixed-type dataset. The second step is rule generation, which generates the number of rules based on the cluster label. In general, the selection of rules in this chapter relies on support and confidence metrics. Specifically, two confidence values are used to determine the Predominant Rule (PR) and Less Significant Rule (LSR) for forecast. The final step involves making predictions based on the constructed rules to determine the equivalent class. Once the forecasted class is attained, the features in the ruleset are examined for further analysis.

5.2 Association Rule Mining

One of the methods used in data mining the most frequently is ARM. [181]. Researchers started deriving rules of association from the databases of transactions in 1993 [182]. Large transactional databases are queried using association rules to extract pertinent information. A transactional database could, for instance, be a shopping cart database, with the products serving as the items, or a text database. Let t= $\{p_1,p_2,p_3\}$ be a transaction of three items, and any combination of them makes up an itemset, for example, having $\{p_1,p_2,p_3\}$, $\{p_1,p_2\}$, $\{p_2,p_3\}$, $\{p_1,p_3\}$, $\{p_1\}$, $\{p_2\}$, $\{p_3\}$. As a result, an association rule would be written as X→Y, where X is an itemset that stands for the antecedent, and Y is an itemset that stands for the consequent. As a result, it can be said that subsequent items and antecedent items frequently co-occur.

Association rules can be applied to obtain knowledge from transactional databases, data warehouses, or other kinds of storage that are useful to extract information to aid in decision-making processes. Three metrics—support, confidence, and lift—are traditionally used to evaluate the effectiveness of association rules for a particular problem [183]. These metrics are defined as follows:

Support (Item set). Supp(X) represents the fraction of transactions in the dataset (D) which has item X out of all transactions. The following equation determines an itemset's support:

$$supp(X) = \frac{|t \in D: X \subseteq t|}{|D|}$$

Support (Associational Rule). The sum of all transactions involving both X and Y is denoted as supp(X→Y), and the following equation defines it:

$$supp(X \rightarrow Y) = supp(X \cup Y)$$

Confidence (Association rule): The confidence value, conf(X→Y), indicates the proportion of transactions that have both item X and Y. It is defined as:

$$conf(X \rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

Lift.It is a significant metric to evaluate the degree of independence between the constituents of a certain association rule. Lift (X→Y) is expressed by the frequency with which X occurs when Y is present or vice versa. The mathematical definition of lift is as follows:

$$lift(X \rightarrow Y) = \frac{conf(X \rightarrow Y)}{supp(Y)}$$

There are numerous methods for extracting association rules. A brute-force method is one choice, although it is not particularly effective. The method used most frequently is based on two steps and the downward-closure feature. The production of frequently occurring itemsets is the first of these processes.The itemset must be more frequent than the minimal support criterion to be deemed frequent. The second stage uses the minimum confidence threshold to obtain the association rules.

Numerous industries, including banking, education, transportation, and others, have adopted association rule mining to help decision-making with scientific data. For instance, in their work, Yu and Zhang [184] proposed a novel weighted integration model that acts as a predictive tool for estimating the risk of loan default. They achieved this by merging techniques from association rule mining, feature bagging, and credit classification research. Likewise, to present solid, scientific evidence supporting acceptable pavement maintenance, in their study, Cao et al. [185] utilized association rules to explore the traffic influence, environmental conditions, and their effects on roadway circumstances.

In their research, Li [186] delved into the relationship between book information and personalized recommendations for readers, employing a powerful association rule algorithm to handle vast amounts of data on book readings. Ariannezhad and Wu [187] introduced a systematic approach to recognize and investigate the patterns of data faults, using mining associations in data to rectify issues with large-scale loop detectors and address problems related to misplaced or inaccurate information gathered through traffic detectors. Guo et al. [188] proposed a situational route planning recommendation system that considers individual preferences in a scenic area, utilizing association rules to reveal distinct travel preferences among various travel groups.

The association rule can be used in education for student behavior analysis, predicting student academic performance. Wang et al. [42] analyze student behavior using ARM. The objective of improving the current data mining for association rules technique resulted in the creation of a comprehensive 4 layer data association mining framework. This framework incorporates acquisition, recording, computation, and examination of data, enabling streamlined executing of vast datasets. Moreover, the algorithm outlines a mining procedure with three steps, starting with "data preprocessing," followed by "discovering association rules," and concluding with "attaining pertinent knowledge." A new categorization model for forecasting the outcome of a student's academic subject using logical association rules is presented by Czibula et al. in [43]. The traditional association rules are expanded to indicate different relationships between data attributes.

5.3 Rule-based Classification

Rule-based classification is any classification method that uses IF-THEN rules to predict classes. Selecting the attributes and related values to consider for classification is a necessary step in the process of investigating

such rules. Typically, rule-based classification schemes include the following elements:

Rule Induction Method This process entails the removal of pertinent IF-THEN rules from the data, and it can be achieved directly through sequential coverage algorithms or indirectly using other DM techniques such as decision tree construction or ARM.

Metrics for Ranking Rules This refers to a set of numbers used to evaluate how effective a rule is in making reliable predictions. The rule induction method frequently uses ranking metrics to eliminate pointless rules and boost productivity. Also, they rate the rules in the class prediction method, which will be used to forecast the class of new cases.

In the realm of data mining, associative classification emerges as a unique subfield that blends association rule mining with classification. It stands as a distinctive form of association rule discovery, focusing solely on the class attribute on the right-hand side (consequent) of the rules. AC generates multiple rules and employs rule trimming and ranking methodologies to identify a subset of high-quality rules. Then, using the smaller rule set, AC may create a powerful classifier. In actual use, associative categorization typically outperforms the decision tree approach regarding accuracy.

A typical AC algorithm consists of three key steps:

Understanding the rules (Training). The algorithm processes the data to produce the rules.

Pruning and ranking of rules. The potential rules extracted from the data are prioritized based on multiple parameters, such as rule confidence, support, and length, to identify the rules incorporated into the classifier. Duplicate rules are eliminated at this point.

Classifying test results. The classifier's rules are used in this step to predict the class values of test data. In addition, this step uses prediction accuracy or error rate to evaluate the classifier's predictive strength.

The classification-based Association (CBA) algorithm, developed by in their study, Lui et al. [189] incorporated ARM and categorization using the CBA technique, which involves two distinct phases. The Apriori algorithm, a well-known search technique, is used to produce CARs initially. The second step is sorting and pruning the CARs to choose the most effective CARs for the classifier. It has been demonstrated that the CBA algorithm produces fewer errors than the decision tree algorithm. Regrettably, the Apriori inheritance in the CBA algorithm, which detects each feasible rules with high frequency at all levels, leads to numerous candidate generation challenges.

Li et al. [190] introduced the Classification using Multiple Association Rules (CMAR) algorithm, which utilizes a cosine R-tree (CR-tree) and a frequent pattern tree (FP-tree) during the rule creation and classification stages. To find frequent rule items, it separates the subset in the FP-tree and then inserts the frequent rule items into the CR-tree based on their frequencies. The database has to be scanned once. The CMAR algorithm utilizes multiple criteria based on the chi-square approach to forecast unforeseen events.

Abdelhamid [191] proposed the Enhanced Multi-label Classifier-based Associative Classification (eMCAC) method for identifying fraudulent websites. This technique constructs rules having multiple class labels from a solitary dataset devoid of requiring learning through recursion. It adopts a vertically structured dataset to represent datasets and calculates the support and confidence scores for a rule with numerous labels by averaging them across all classes. When the attribute values satisfy the rule's antecedent, the test data point is assigned to the class.

The FACA algorithm was suggested by Hadi et al. [192] for detecting phishing websites. By frequently expanding (k-1)-rule items, it finds k-rule items. The quantity of feature values, confidence, support, and frequency determines the order of the rule items. The FACA technique employs forecasting techniques with precise matches to predict unseen data. It compares all Class Association Rules (CARs) in the classification algorithms with the hidden information and assigns the class label having the uppermost total to the unobserved data.

The Predictability-Based Collective Class Association Rule (PCAR) approach was developed by Song and Lee [126] to enhance rule evaluation. The PCAR algorithm calculates the predictability value of Class Association Rules (CARs) through internal cross-validation among the training and test datasets. The CARs are then ranked based on their predictive values, confidence, support, length of antecedents, and occurrences. Finally, the full-matching approach allocates a class label to the hidden data.

The WCBA technique was presented by Alwidian et al. [127] to improve classifier precision using a weighting method. The pruning process involves the use of a statistical metric, and the algorithm employs a weighted technique to choose meaningful Class Association Rules (CARs). Moreover, the harmonic mean—a metric between support and confidence—categorizes CARs priors.

Rajab [120] introduced the Active Pruning Rule (APR) technique. Initially, the classifier expands to include the initial rule which equals a data point. Subsequently, instances containing the first rule are removed. Lastly, all Class Association Rules (CARs) are reranked, and the support and confidence of the residual rules are recomputed. It has been demonstrated that the APR algorithm can decrease classifier size while maintaining predicted accuracy.

A new AC algorithm (ECARG) is suggested by Thanajiranthorn and Songram [193] to quickly identify a small number of effective classification rules without the need for pruning. Moreover, to evade superfluous rule creation, a perpendicular data depiction method is employed by the algorithm and expedites the mining procedure.

Table 5.1 shows some associative classification algorithm advantages and drawbacks.

Table 5. 1 AC algorithms merits and demerits

Algorithm	Merits	Demerits
CBA	It employed the association rule technique, which is more appropriate than the conventional classification technique, to categorize the data.	The minimal support criterion sensitivity needs to be addressed. When a less minimum support level is specified, a excess of rules is created.
CMAR	The implementation employs a powerful FP-tree, which optimizes memory and storage usage more effectively compared to CBA.	In scenarios with numerous attributes, the FP-tree might exceed the capacity of the main memory.
eMCAC	By utilizing vertical data representation, the system conserves space and efficiently locates a multi-label class.	It employs a method akin to Apriori, which generates numerous frequent itemsets as its outcome.
FACA	Additionally, employing set difference contributes to reduced memory usage and faster mining time.	Since it uses an Apriori-like methodology, the algorithm must look for every frequent itemset among all potential candidate itemsets at every level.
PCAR	It eliminates redundant rules by using predictability value.	The prolonged execution duration can be attributed, in part, to the inner cross-validation process used to calculate the predictability value.

Algorithm	Merits	Demerits
WCBA	By incorporating a weighted approach, the system selects relevant rules to improve the classifier's performance significantly.	Due to expert determinations, weighted elements are liable to vary, which may lead to a various experimental outcome.
APR	It also introduces a novel evaluation technique that utilizes a modest classifier yet achieves remarkable accuracy.	Specifying a low minimum support criterion result in the generation of numerous rules
ECARG	It produces fewer rules with a high accuracy rate, speeds up the calculation, and uses less memory.	The performance on unbalanced datasets is poor. It is unable to categorize the minority class accurately.

5.4 Proposed Methodology

This section explains the ARMpred algorithm based on AC. The approach contains three steps. The first step is preprocessing, which converts numerical and categorical attributes using discretization. This step is used only if the dataset is a numerical or mixed-type dataset. The second step is rule generation, which generates the count of rules based on the cluster label. Typically, support and confidence metrics are employed to choose rules. In this chapter, two support values MIR and LIR, are utilized to forecast. The final step involves making predictions based on the created rules to determine the corresponding class. Following the prediction, attribute analysis is performed on the rule set. For a visual representation of the proposed method's workflow, refer to Figure 5.1.

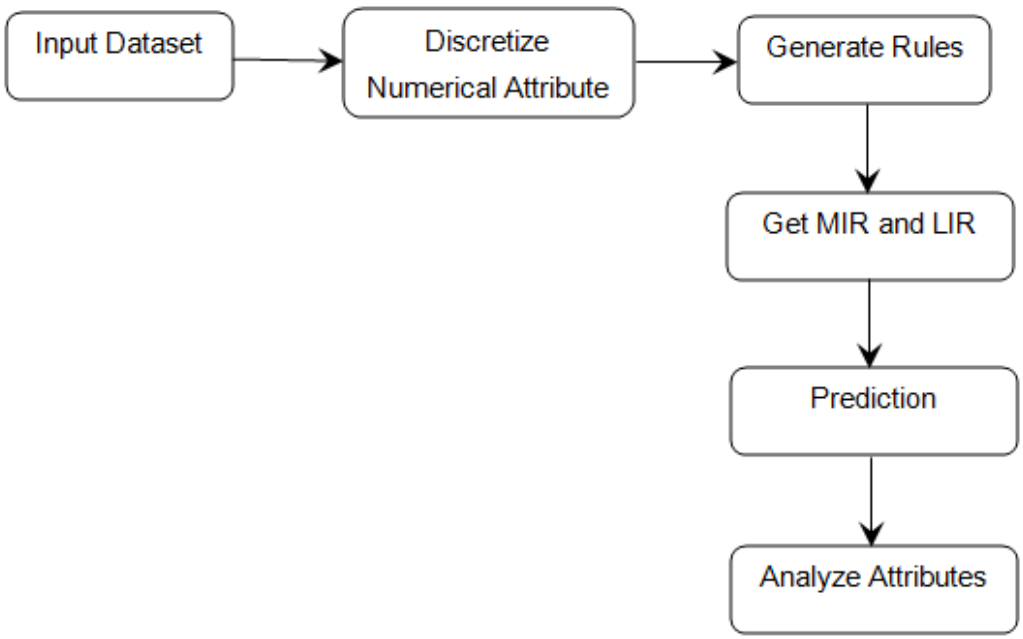


Figure 5. 1 ARMpred Workflow

Let T be the dataset containing n number of instances (t_1, t_2, …, t_n) with m number of attributes A={a_1, a_2. ….,a_m} with k class label C = {c_1, c_2, …, c_k}.

An item is represented as an attribute ai, comprising a nominal value vj denoted as (ai, vj).

An itemset (attribute set) is the set of items (attributes) represented as (a_{i1}, v_{i1}),(a_{i2}, v_{i2}), ...,(a_{ij}, v_{ij}).

A rule takes the form ⟨attributeSet, c_k⟩, signifying the association between an attribute set (itemsets) and a class within a dataset, denoted as attributeSet → c_k. The absolute support of a rule, represented as sup(r), corresponds to the number of transactions containing that specific rule, r. The support of r could be represented by

$$sup(r) = |f(r)|$$

where f(r) indicates the number of instances containing r.

The confidence of the rule ⟨attributeSet, c_k⟩ is determined by the ratio between the number of instances containing the attribute set and belonging to class c_k, and the total number of instances containing the attribute set, as

$$conf(\langle attributeSet, c_k \rangle) = \frac{|f(\langle attributeSet, c_k \rangle)|}{|f(attributeSet)|}$$

The minimum support threshold (minsup) is considered, and a rule is deemed recurrent if its support is equivalent to or exceeds minsup. On the other hand, a recurrent rule with confidence equal to or higher than the minimum confidence threshold (minconf) is referred to as a CAR.

Algorithm-1 explains the proposed ARMpred. First, step1 converts the numerical attribute into a nominal attribute using equal-width discretization. Then, step2 generates rules using algorithm-2.

Algorithm-1 ARMpred

Input: Dataset D, minsup1, minsup2

Output: CAR, Prediction Results

Step1: Preprocessing

1.1 If D contains numerical attributes,then

1.2 apply Discretization

1.3 End If

Step2: Rule Generation

2.1 Generate MIR and LIR using Algorithm-2

2.2 Remove redundant rules

Step3: Prediction

Algorithm-2 Rule Generation

Input: Dataset D, min_support1, min_support2

Output: MIR and LIR

Step 1 R1_set = Generate 1-attribute set rules from D for each class

Step 2 If support(R1_set) ≥ min_support1 then

Step 3 add R1_set into MIR

Step 4 Else If min_support2 ≤ support(R1_set) < min_support1

Step 5 add R1_set into LIR

Step 6 Else

Step 7 Discard R1_set

Step 8 End If

Step 9 While attributeSet > 0 do

Step10 Find Frequent (m+1)-attribute set

Step11 Rm+1_set = Generate (m+1)-attribute set rules from D

Step12 Add Rm+1_set into MIR or LIR based on support value

Step13 Stop If no (m+1)-attribute set frequent rule

Step14 End While

In this algorithm, the frequent -1 attribute set is generated for each class.In addition, two support values are used to generate two kinds of rules MIR and LIR, for efficiently identifying the minority class rules. Table 5.2 and Tabl3 5.3 show student academic data's sample MIR and LIR.

Table 5. 2 MIR - Student Data

EI-High EP- Extroversion SE- High EH-Happy PANA- Positive RSE-High SD-High Performance-Excellent
EI-High EP-Extroversion SE-High EH-Moderately PANA-Positive RSE-High SD-High Performance-Very Good
EI-Low EP-Psychotism SE-Low EH-Moderately PANA-Negative RSE-Average SD-Low Performance-Fail
EI-High EP- Neuroticism SE-High EH-Happy PANA-Positive RSE-High SD-High Performance-Above Average
EI-Low EP-Psychotism SE-Low EH-Unhappy PANA-Negative RSE-Average SD-Low Performance-Fail

Table 5. 3 LIR - Student Data

EI-High EP- Extroversion SE- High EH-Happy PANA- Positive RSE-High SD-High Performance-Fail
EI-High EP-Extroversion SE-High EH-Moderately PANA-Positive RSE-High SD-High Performance-Fail
EI-Low EP-Psychotism SE-Low EH-Moderately PANA-Negative RSE-Average SD-Low Performance-Excellent
EI-High EP- Neuroticism SE-High EH-Happy PANA-Positive RSE-High SD-High Performance-Fail
EI-Low EP-Psychotism SE-Low EH-Unhappy PANA-Negative RSE-Average SD-Low Performance-Above Average

5.5 Experimental Results

The efficiency of the suggested approach is assessed through experimentation in this chapter. The evaluation involves the utilization of seven student questionnaires and three UCI datasets to examine the outcomes. Table 5.4 provides a summary of the student questionnaires dataset utilized in the experiments.

Table 5. 4 Dataset Details

Dataset	No of Attributes	No of Classes	Instances
Emotional Intelligence (EIQ)	13	3	1000
Eysenck Personality (EPQ)	13	3	1000
General Self -Efficacy (GSE)	13	2	1000
Emotional Happiness (EHQ)	13	3	1000
Positive and Negative Attitude (PANA)	13	3	1000
Self Esteem (RSE)	13	3	1000
Self Determination (SDS)	13	2	1000

The evaluation of the outcomes involves the utilization of the following metrics: number of rules, accuracy and execution time. Table 5.5 shows the performance metric comparison for seven student questionnaires dataset

Table 5. 5 Assessment Metrics for the Dataset

Dataset	Rules Count	Accuracy in %	Execution time in ms
EIQ	27	71	256
EPQ	7	68	133
GSE	27	79	121
EHQ	15	71	131
PNA	24	86	110
RSE	23	88.82	87
SDS	20	65.95	102

Table 5.6 shows the data summary of three UCI datasets.

Table 5. 6 UCI Dataset Description

Dataset	FACA	ECARG	ECARG2	ARMpred
Lenses	**5**	7	8	7
Post-Operative	12	**11**	27	31
Tic-Tac-Toe	12	**6**	33	9

Table 5.7 shows the performance metric comparison for three UCI datasets.

Table 5. 7 Assessment Metrics for UCI dataset

Dataset	Rules #	Accuracy (%)	Execution Time (ms)
Lenses	7	87.5	73
Post-Operative	31	91.11	649
Tic-Tac-Toe	9	99.37	2105

The proposed ARMpred algorithm evaluation metric number of rules, accuracy and execution time is compared with FACA [192], ECARG [193], and ECARG2 [193].

Table 5.8 and Figure 5.2 show comparisons for the UCI dataset based on number of rules generated.

Table 5. 8 Number of Generated Rule Comparison

Dataset	FACA	ECARG	ECARG2	ARMpred
Lenses	**5**	7	8	7
Post-Operative	12	**11**	27	31
Tic-Tac-Toe	12	**6**	33	9

The comparison outcomes indicate that the ECARG method outperforms other algorithms in terms of reducing the number of rules for the Post-operative and Tic-Tac-Toe datasets.

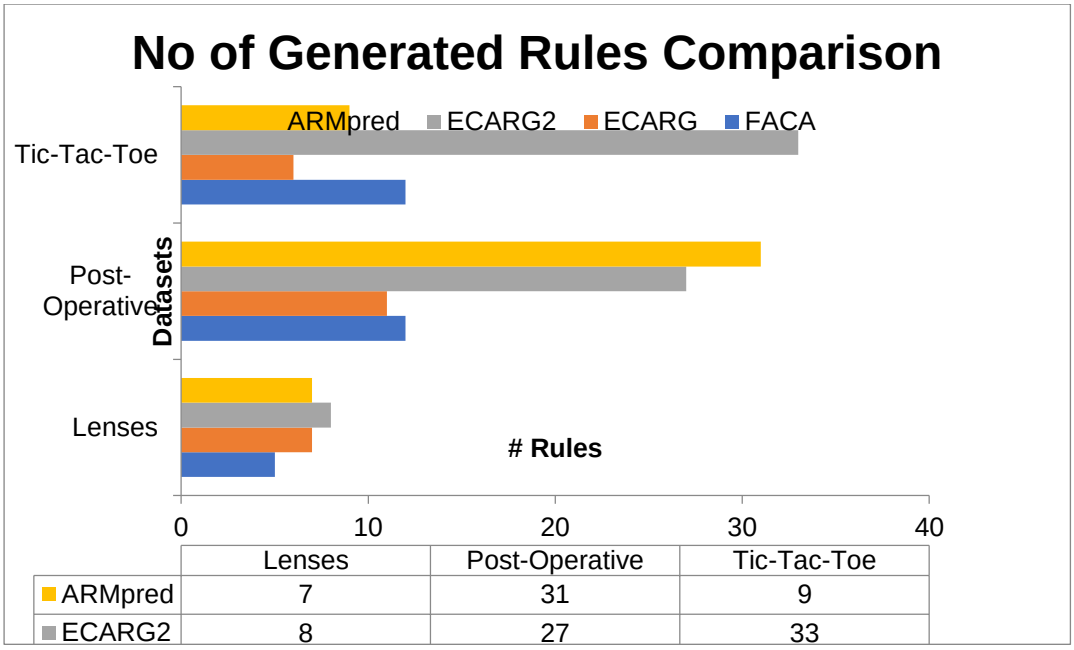


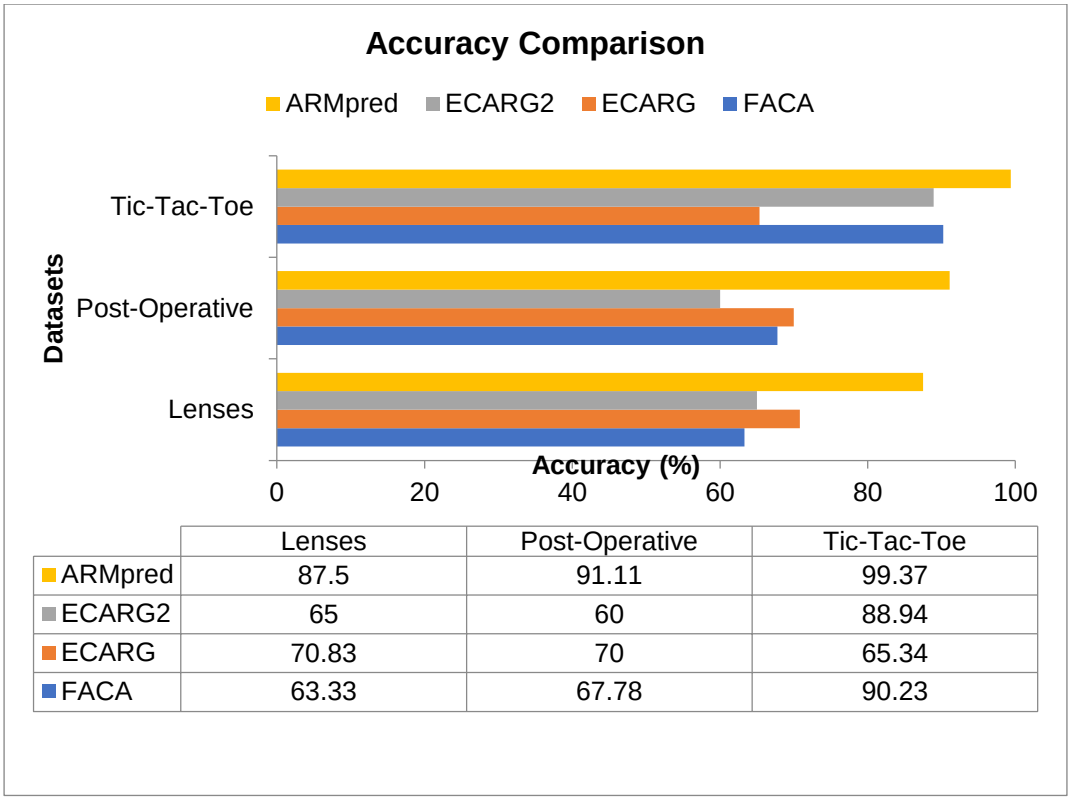
Figure 5. 2 No of Generated Rules Comparison

Table 5.9 and Figure 5.3 show the accuracy comparison for the UCI dataset.

Table 5. 9 Accuracy (%) Comparison

Dataset	FACA	ECARG	ECARG2	ARMpred
Lenses	63.33	70.83	65.00	**87.5**
Post-Operative	67.78	70	60.00	**91.11**
Tic-Tac-Toe	90.23	65.34	88.94	**99.37**

The proposed algorithm increases the accuracy compared to other algorithms.



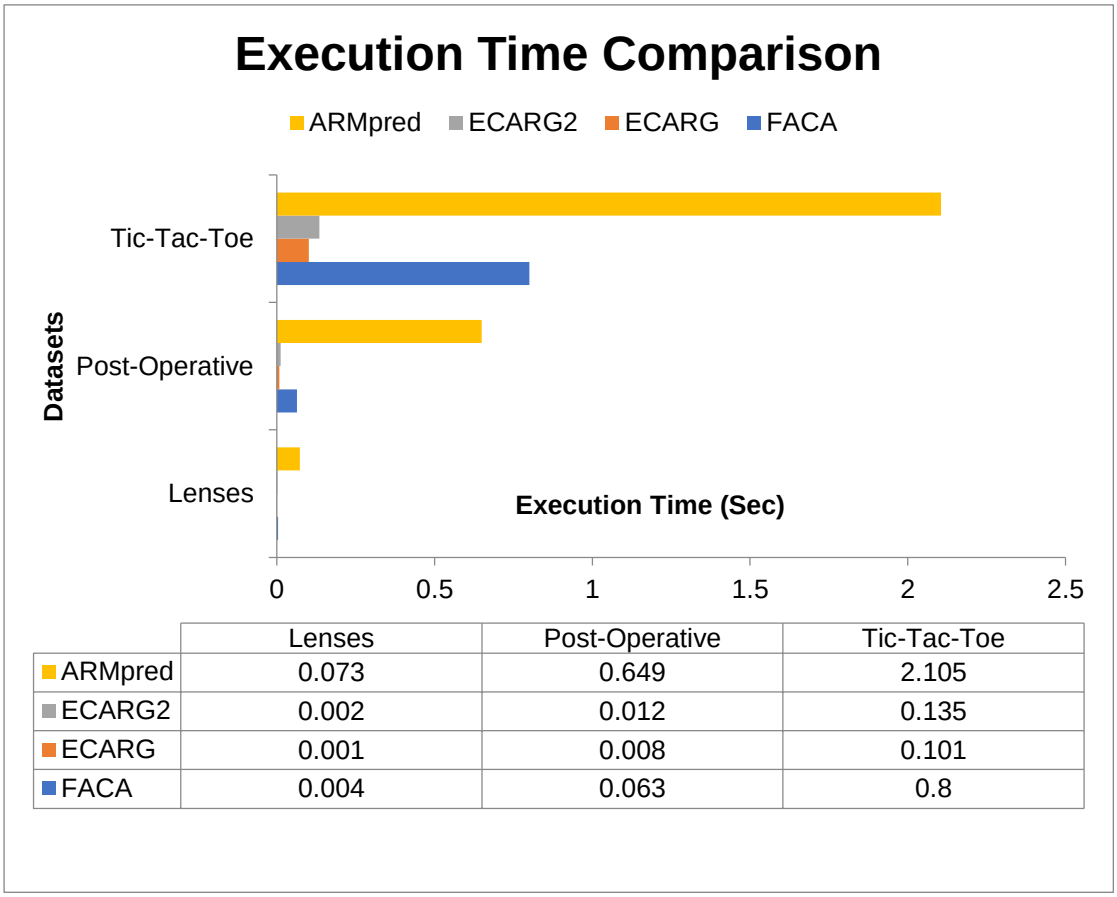
	Lenses	Post-Operative	Tic-Tac-Toe
ARMpred	87.5	91.11	99.37
ECARG2	65	60	88.94
ECARG	70.83	70	65.34
FACA	63.33	67.78	90.23

Figure 5. 3 Accuracy Comparison for UCI Dataset

Table 5.10 and Figure 5.4 shows the execution time comparison for the UCI dataset.

Table 5. 10 Execution Time (Sec) Comparison

Dataset	**FACA**	**ECARG**	**ECARG2**	**ARMpred**
Lenses	0.004	**0.001**	0.002	0.073
Post-Operative	0.063	**0.008**	0.012	0.649
Tic-Tac-Toe	0.800	**0.101**	0.135	2.105



	Lenses	Post-Operative	Tic-Tac-Toe
ARMpred	0.073	0.649	2.105
ECARG2	0.002	0.012	0.135
ECARG	0.001	0.008	0.101
FACA	0.004	0.063	0.8

Figure 5. 4 Execution Time Comparison

5.6 Summary

The enormous number of rules in AC techniques is a challenge acquired from association rule mining. Due to the massive classifiers that can be developed, which are difficult to regulate by end users, this problem limits the usage of AC in application areas. Some rule pruning methods are used to remove low minimum frequency rules. However, it may result in less performance. In this chapter, the efficient prediction algorithm ARMpred is explained. It is based on generated rules from clustered data through AC, and it chooses the MIR and LIR for prediction using two minimum support values. The proposed algorithm improves the accuracy. It also efficiently classifies the minority class label.