Modern Applied Science; Vol. 17, No. 1; 2023
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

An Intelligent Technique to Predict the Autism Spectrum Disorder Using Big Data Platform

Jaber A. Alwidian[1]

[1] The Group Securities, Doha, Qatar

Correspondence: Jaber Alwidian, The Group Securities, Doha, Qatar. E-mail: algo@thegroup.com.qa

Received: March 13, 2023 Accepted: April 2, 2023 Online Published: April, 4, 2023

doi:10.5539/mas.v17n1p28 URL: https://doi.org/10.5539/mas.v17n1p28

Abstract

Autism or autism spectrum disorder (ASD) is considered a psychiatric disorder. It is a condition that puts constraints on the use of linguistic, cognitive, communicative, and social skills and abilities. Recently, many data mining techniques have been developed to help autism patients by discovering the main features of the condition and the correlation between them. In this paper, we employ the association classification (AC) technique as a data mining approach to predict whether or not an individual has an autism. The Intelligent Classification Based on Association rules (ICBA) algorithm is proposed for finding the correlations between the features to decide whether an individual has autism in its early stage, especially in childhood. The ICBA algorithm incorporates the chi-square method to select the best feature to make the decision, in addition to proposing new techniques in all phases and increasing number of folds to 2size of data/10. The proposed algorithm is compared against four well-known AC algorithms in terms of accuracy to evaluate their behavior in the prediction task using big data platform. The results show a better performance for the ICBA algorithm in most experiments. Moreover, all of the considered algorithms had an increased level of accuracy when the chi-square method was used.

Keywords: classification, association rules, association classification (AC), autism spectrum disorder (ASD), data Mining, big data, big data platform

1. Introduction

Data mining can be described as the extraction of unknown information from a huge amount of data. It provides various different approaches, such as classification, association rule, clustering, and regression, to discover hidden insights. The main aim of the classification process is to predict the class value of any given unknown instance. Meanwhile, the association process discovers the correlations between attributes in the dataset. In the past few years, researchers have adopted a new technique called association classification (AC) to classify the accuracy of the classification process (Abdelhamid et al., 2014; Abdelhamid et al., 2015; Abdallat et al., 2019). The AC technique aims to build a classifier from a large labeled dataset, referred to a training data, in order to predict the class value of unseen instances, referred to as test data (Abu-Mansour et al.,2012; Tan et al., 2006; Hadi, 2013; Abdelhamid et al., 2015; Salah et al., 2019).

All AC algorithms function act on the basis of three steps: rule generation, pruning, and classification/prediction. The rule generation step aims to generate rules from the dataset at represent the correlation between the items in the dataset. The pruning step reduces the number of weak rules to enhance the accuracy of the classifier. Finally, the classification step is used to predict the unknown class value for any given instance (Hadi, 2013; Taware et al., 2015, Abuqabita et al., 2019).

The AC technique is popular for two main reasons, namely the classifier's high throughput accuracy rate and the understandability of the rules that are generated by the classifier. A simple rule can be described in the form "$A \rightarrow C$", where A is a conjunction of a set of items and C is the class label. Unfortunately, using the AC mining technique has some weaknesses, including generating a huge number of rules, which consumes more time and memory than traditional data mining techniques. Moreover, the user estimates the support and confidence measures, while these estimated thresholds (Dou et al,. 2019; Hadi, 2013; Shorfuzzaman et al,. 2019, Al-Fayoumi et al., 2020) affect the rule generation, pruning, and ranking processes.

In the literature, many studies have shown that AC techniques outperformed traditional classifiers due to the number of rules that can be generated during the rule generation stage and the way in which these rules can be

easily understood to select the most important ones. However, the number of rules and their types may play a dominant role in the prediction phase (Liu et al., 1998; Wa'el et al., 2010; Alwidian et al., 2012; Abdelhamid et al., 2014; Ma et al., 2014; Abdelhamid et al., 2015; Alazaidah et al., 2015; Taware et al., 2015; Shahin et al., 2019). Sorting the generated association rules is one of the most critical issues in deciding which rules have the highest importance and which ones have the lowest importance and hence eliminated. Support and confidence measures are the main measures used to differentiate the association rules (Tan et al., 2006; Hadi, 2013; Alwidian e al., 2020).

The big data solution offers two strengths for any machine learning approach that are 1) increasing size of data exponentially in the training phase may enhance the performance of the machine learning approach and 2) increasing number of computations exponentially on normal size of data leads to enhance the training and testing phases (Alwidian and Hadi, 2012; Xing et al., 2015). In our proposed solution we employ the second strength to obtain accurate measurements.

In this paper, a statistical measure is investigated and tested to see how it would affect the accuracy of AC technique(s). in addition to generate the accuracy level based on exponential number of folds to make the measure more accurate. A set of experiments are conducted using autism datasets, which are selected from the UCI repository to evaluate the most common AC algorithms: Classification Based on Association Rules (CBA), Multi-class Classification based on Association Rule (MCAR), Fast Associative Classification Algorithm (FACA), and Fast Classification Based on Association rules (FCBA). All previously mentioned algorithms are compared against the proposed Enhanced FCBA algorithm (ICBA) in terms of accuracy in relation to autism patients.

The paper is organized as follows: Section 2 presents the background on AC. Section 3 gives details of autism. The related work is presented in section 4. Section 5 presents the proposed big data platform. The proposed technique is described in details in section 6. Extensive experiments and their results are presented in Section 7. Finally, the conclusions and future research suggestions are presented in Section 8.

2. AC Background

In Alwidian et al. (2018), the AC approach combined the association rules and classification task: for example, if a rule such as $A_1 \rightarrow Class_1$, then $Class_1$ is a class value. The training dataset T has m distinct attributes $A_1, A_2 \cdots A_m$ and C as a list of class values. A training object in Tcan be a set of attributes, $A_1, A_2 \cdots A_i$ A1, and a class $(Class_i)$ and the items described as an attributes, A_i, and values, V_i, where an $itemset$ is a set of combined items contained in a training object.

A rule item r is formed as $< itemset, C_i >$ where $C_i$is the class value. The actual occurrence $(actoccrr)$ of a rule item r in $T$is the number of tuples in T that match the itemsets defined in r, where the support-count $(supp\_count)$ of rule item r is the number of tuples in T that match $r's$ itemsets, and belong to a C_i for r, as shown in Equation 1.

$$\mathrm{Supp_{cout}} = \mathrm{r} \cup \mathrm{C_i} \qquad \text{Equation (1)}$$

A rule item r exceeds the minimum-support value if $(\mathrm{supp}\_count(r)/|T|) \geq minimum\_support$, where $|T|$ is the set of tuples in T, as shown in Equation 2.

$$\mathrm{Support} = \frac{(\mathrm{r} \cup \mathrm{C_i}).\,\mathrm{count}}{|\mathrm{T}|} \qquad \text{Equation (2)}$$

A rule item r exceeds minimum-confidence value if $\left(\mathrm{supp}\_count(r)/actoccrr(r)\right) \geq minimum\_confidence$, as shown in Equation 3.

$$\mathrm{Confidence} = \frac{(\mathrm{r} \cup \mathrm{C}).\mathrm{count}}{\mathrm{r.count}} \qquad \text{Equation (3)}$$

Any rule item r that exceeds the $minimum\_support$ value will be a frequent rule item and a is described as: (A_{i1}, V_{i1}) ^ (A_{i1}, V_{i1}) ^ ... ^ $(A_{1m}, V_{im}) \rightarrow C_i$.

3. Autism Spectrum Disorder (ASD) Background

ASD is a brain development disorder that limits communication and social behaviors (Bolton et al., 1994; Thabtah, 2017). A number of tools are used for ASD diagnosis. Examples of clinical diagnosis approaches are Autism Diagnostic Interview (ADI) [Lord et al., 1994) and Autism Diagnostic Observation Schedule-Revised (ADOS-R) (Lord et al., 2014]. To enhance the accuracy of ASD diagnosis, researchers recently adopted machine learning approaches (Bone et al., 2014; Duda et al., 2016; Wall et al., 2012a; Wall et al., 2012b). The main goals

of these approaches are to:

(1) Improve the classification accuracy

(2) Reduce the screening time

(3) Identify the smallest number of ASD codes to reduce the complexity of this problem.

Data mining offers automated classification models for ASD that are effective and efficient. These models combine various mathematical and search methods adopted from the field of computer science (Thabtah, 2007; Thabtah, 2017). Researchers have recently developed a number of data mining techniques for the ASD issue, e.g. support vector machine (Platt, 1998), decision trees (Quinlan, 1993), rule neural network (Mohammad et al., 2014), and classifiers (Abdelhamid and Thabtah, 2014). ASD diagnosis is regarded as a typical data mining classification problem, in that known classified instances can be used to build a model. The diagnosis of a new instance (ASD, No-ASD) can then be predicted using this technique.

Currently, data scientists use existing open source software to achieve this; WEKA (Hall et al., 2009) is an example of such software. The processed dataset is firstly loaded and the data mining algorithm is then applied. Various measures can be used to determine the effectiveness of the selected data mining method for predicting the diagnosis. Examples include accuracy, false positive rates, false negative rates, the model building time, and true negative rate. Data mining software packages often incorporate such evaluation measures.

4. Related Work

Different AC algorithms were developed to increase the classifiers accuracy and the building time model based on using one of the rule generation techniques and prediction methods. These algorithms include CBA (Liu et al., 1998; Alsahlee et al., 2019), Classification based on Multiple Class Association Rules (CMAR) (Li et al., 2001) and MCAR (Thabtah et al., 2005; Alwedyan et al., 2011). They have common steps in the way they work, while they vary in their rule generation process.

New AC algorithms for the multi-class prediction process and adopting different types of algorithms were developed based on dynamic and greedy algorithms. These types of algorithms lead to propose many other solutions such as: CAEP (Dong et al., 1999), Classification based on Predictive Association Rules (CPAR) (Yin and Han, 2003), CAAR (Xu et al., 2004), Negative Rules (Antonie and Zaïane, 2004; Alwidian et al., 2016), Live and Let Live (L3) (Baralis et al., 2004; Alwidian et al., 2020), Multi-class Multi-label Associative Classification (MMAC) (Thabtah et al., 2004; Hadi et al., 2013), 2-PS (Qian et al., 2005), Class Based Associative Classification Approach (CACA) (Tang and Liao, 2007), Associative Classification Based on Closed Frequent Itemsets (ACCF) (Li et al., 2008), Boosting Association Rules (BCAR) (Yoon and Lee, 2008), and Associative Classifier with Negative Rules (ACN) (Kundu et al., 2008).

Spark platform used to evaluate six machine-learning algorithms on five health datasets in (Hadi et al., 2010; Nagarajan and Babu, 2019). The evaluation process was in term of accuracy and computational time that show well performance for the random forest and logistic regression algorithms in term of accuracy while the Naïve Bayes was the best in term of computational time.

In Ajayi et al. (2019) the big data technologies used for health safety and risks analytics with very large size of dataset. The solution focused on building big data platform to serve this size of data and the lifecycle of health risk analytics, while the architecture prototype interfaced different technology artefacts was implemented in Java programming language to predict the likelihoods of health hazards occurrence. The proposed architecture was able to find relevant features and enhance the explanatory capacities and preliminary prediction accuracies.

In Liu et al. (1998), the CBA algorithm was proposed to merge the classification task with the association rules. This algorithm functions in three phases: rule generation, pruning, and prediction. In the rule generation phase, the Apriori algorithm was implemented to identify the most frequent $itemset$ that represents the Class Association Rules (CARs) which passes minimum-support and minimum-confidence measures. The following steps explain how this is done.

(1) Find the candidate single $itemset$. Table 1 depicts a dataset sample for two training objects. In this dataset, we have three single items, namely v1, v2, and v3.

Table 1. Dataset (T) for two objects

Attribute $A(at_A)$	Attribute $B(at_B)$	Class (C)
v_1	v_2	C_1
v_1	v_3	C_2

(2) Find the frequent single $itemset$. Select the items for which the support is greater than or equal to a given minimum-support of the candidate set, where the support of an item can be calculated using Equation 4.

$$Support = \frac{(X \cup Y).count}{n} \quad \text{Equation (4)}$$

where x is the attribute, y is the name of the class, and n is the number of rows in the dataset.

(3) Find the two- $itemset$ candidate rules (i.e. each rule should have two items on the left-hand side, for example $(v_1, v_2 \rightarrow C_1)$.

(4) Find the frequent two- $itemset$ that satisfies the minimum-support.

(5) Repeat to find the next $itemset$ until the set is empty.

(6) Generate the CARs from the produced set based on selecting the rules with confidence values greater than or equal a given minimum-confidence. The confidence of an item can be calculated using Equation 5.

$$\text{Confidence} = \frac{(\text{X} \cup \text{Y}).\,\text{count}}{\text{X}.\,\text{count}} \quad \text{Equation (5)}$$

After generating the rules, the M1 method will be used in the pruning phase to choose the best rules that cover the entire dataset. Finally, to predict the class value for any given instance, the class of the first rule that can match this instance will be assigned as its own predicted class.

Li et al. (2001) developed a new association classification algorithm (CMAR). This algorithm was developed based on adopting new approaches in rule generation and classification, which are considered the two main steps in this algorithm. In the rule generation step, FP-tree and CR-tree were employed to generate rules. The classification step in the CMAR algorithm finds the value class for its input by finding all the rules that can predict this input and then evaluates all of these rules to predict the class value. At the end, CMAR was compared with some AC algorithms and the results showed that CMAR outperformed other algorithms.

Thabtah et al. (2005) proposed the MCAR algorithm to overcome the CBA's dataset multi-scanning process for generating the rules. In MCAR, the single itemsets are selected using the Tid-list approach. In addition, the occurrences for each rule are kept, facilitating the next itemset generation step without scanning many times.

In Alwidian et al. (2016), the Apriori algorithm was optimized using general rule generation to overcome the long time needed in the generation phase to achieve incremental application. The authors proposed the FCBA algorithm and compared this with a set of AC algorithms in terms of accuracy, recall, precision, F1, and building time model measures.

A new FACA algorithm was proposed in Hadi et al. (2016). The Diffset method used to generate rules to enhance the efficiency of the classifier. It also sorts the generated association rules according to the minimum number of values on the left-hand side. The FACA algorithm proposed a multi-rules method in the prediction step to enhance the accuracy level of the classifier. In this phase, this algorithm splits the pass rules to a set of groups based on the class and then selects the class that has strongest rules. The authors evaluated the efficiency of their algorithm by comparing it in terms of set measures with well-known AC algorithms.

Table 2 shows the main stages and the internal techniques for the CBA, MCAR, FACA, and FCBA algorithms. In the rule discovery stage, all of these algorithms generate the same rules based on minimum-support and minimum-confidence as estimated measures; the main difference between them is the data structures that are used to store the entire data. In the CBA algorithm, the rules are generated by visiting the database directly without any changes to its structure that would lead to more time being spent at this stage. The MCAR and FACA algorithms, meanwhile, convert the database to lists to solve the multi-scan database problem, which requires a long time for the rule generation process.

Ranking is the most critical stage in these algorithms. It sorts the generated rules, based on a suggested set of measures, from the highest priority to the lowest. Based on the pruning technique in the next stage, some of these rules will then be eliminated and the others retained. Thus, the number and type of rules can be different in these algorithms for the same dataset. Finally, at the prediction stage, the type of prediction method that is used plays a very important role in increasing or decreasing the accuracy level.

Table 2. Main stages for the CBA, FACA, MCAR and FCBA algorithms

Name	Rule generation	Ordering	Pruning	Prediction method	Data layout
CBA	APRIORI ALGORITHM	CONFIDENCE MEASURE SUPPORT MEASURE FIRST RULE	COVERAGE OF DATABASE METHOD	MAXIMUM MATCH	HORIZONTAL
FACA	DIFF-SET	CONFIDENCE MEASURE SUPPORT MEASURE CLASS CARDINALITY	COVERAGE OF DATABASE METHOD	MULTIPLE RULE	VERTICAL
MCAR	TID-LIST	CONFIDENCE MEASURE SUPPORT MEASURE CLASS CARDINALITY FREQUENCY DISTRIBUTION	COVERAGE OF DATABASE METHOD	MAXIMUM MATCH	VERTICAL
FCBA	APRIORI ALGORITHM	SUPPORT MEASURE CONFIDENCE MEASURE FIRST RULE	COVERAGE OF DATABASE METHOD	MAXIMUM MATCH	HORIZONTAL

According to Hadi et al. (2016), Nguyen et al. (2016), and Thabtah et al. (2011), these algorithms have the following weaknesses.

(1) The CBA algorithm needs more than one scan for the dataset, thus requiring more memory and time.

(2) The MCAR and CBA algorithms do not split the dataset based on the classes, and this can lead to the generation of more unnecessary rules, negatively affecting the classifier speed.

(3) The FACA algorithm prefers more specific rules than general rules, affecting the accuracy of the prediction method.

(4) All of these algorithms used the minimum-support and minimum-confidence measures assigned by the user, which are support and confidence. This, if there are any rules that have confidence or support values less than the minimum-confidence and minimum-support, these will not be selected in the generated rules, i.e. if minimum-support = 0.2 and minimum-confidence = 0.6, rules with support = 0.6 and confidence = 0.55 will not be selected.

These weaknesses motivated us to propose a new algorithm to serve autism patients. This algorithm generates a set of rules by using the harmonic mean (HM) measure to enhance the accuracy of the classifier.

5. Big Data Platform

Our proposed solution builds a big data platform to enhance the WEKA performance in the training phase and testing based on 2size of data folds where, size of data represents number of records in the dataset (i.e. if we have dataset of 1000 records, then number of folds will be 21000). The huge number of combinations that could be produced form this assumption leads to use parallel programming technique that already embedded in Spark apache.

Our big data platform is Cloudera platform that contains some of selected components to serve the main functionality of our proposed solution as shown in Figure 1. Distributed WEKA has been integrated with SPARK apache to enhance the model building time for the machine learning algorithms that will be evaluated based on huge number of folds to make enhance the accuracy measure (Meng et al., 2016).

Furthermore, apache Hive is used to store our data above the Hadoop Distributed File System (HDFS) that uses MapReduce and Yet Another Resource Negotiator (YARN) apaches to run multi task and manage the resources at the same time within the environment (Vavilapalli et al., 2013). Finally, Hadoop User Experience (HUE) apache employed to investigate the data on the HDFS by using a user-friendly interface.

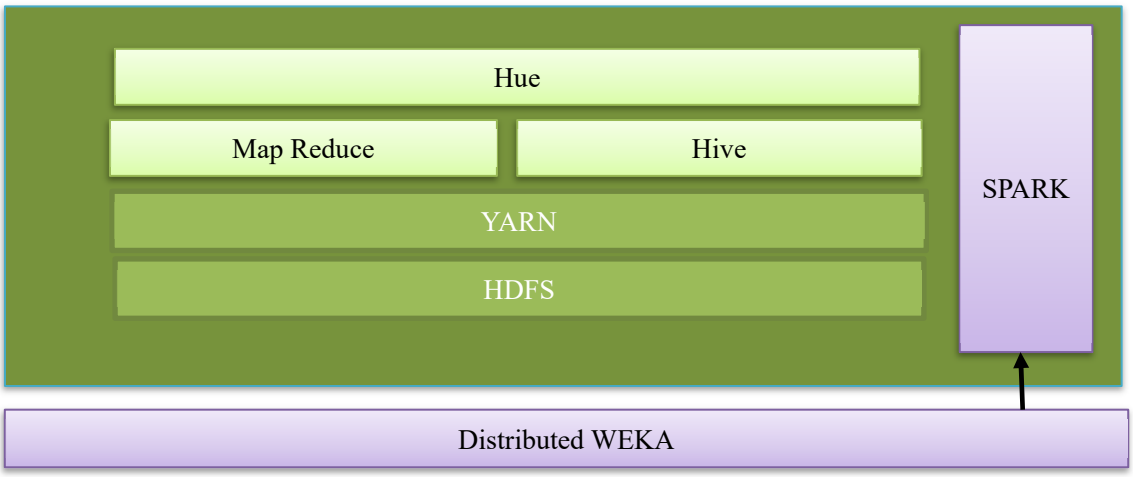

Figure 1. Our proposed big data platform

6. Proposed Model

The proposed Intelligent Classification Based on Association (ICBA) algorithm aims to overcome the estimated measures that occurred in the association classification algorithm, thus increasing the accuracy of classifier. Moreover, this algorithm uses the incremental application that is needed to rebuild the classifier for each new instance in order to reflect the changes on the classifier, thus enhancing the accuracy measure.

Assumption 1: We assume that the ICBA algorithm differentiates between the attributes in the selected dataset based on the weight assigned to each attribute by using the chi-square method to eliminate the weak attributes that affect the accuracy of the classifier.

Assumption 2: We assume that the ICBA algorithm differentiates between the generated rules in the selected dataset based on the HM value for each rule. In addition, it will generate general rather than specific rules, improving the accuracy model and covering a large portion of the dataset. Furthermore, the general rules work properly with the voting prediction method.

6.1 Detailed Description of the ICBA Algorithm

The ICBA algorithm contains four phases, as shown in Figure 2 and Algorithm 1: 1) Preprocessing phase; 2) Rule generation phase; 3) Pruning phase; an Fig 2. ICBA stages

```
Algorithm 1: ICBA
1 Dataset D with T training objects
2 ICBA (D, T)
3 {
4 D' = chi-square (D)
5 Divide (D')       [Training-data, Test-data] // D' is divide into training data and test data
6 S = empty set
7 S = Generate-Rule (Training-data, n, minimum-support, minimum-confidence)
8 Rule-Pruning (S)       [rules]
9 Prediction (rule-set, test-data)
10 }
```

6.1.1 Preprocessing Phase by Chi-square Method

The ICBA algorithm employs the chi-square (χ2) test to select the features in any given dataset based on statistical measures to show the dependencies between the features. Chi-square is a very commonly used method (Wall et al., 2012b). It evaluates the strongest features by finding the value of the chi-square statistic with regard to the class value. The initial hypothesis H0 is that the two features are independent, and this is tested using the chi-square equation:

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Where Oij is the actual frequency and Eij is the estimated frequency, asserted by the null hypothesis. The greater the value of χ2, the greater the evidence contradicting the hypothesis H0.

6.1.2 Rule Generation Phase

After applying the chi-square method as the feature selection technique to select the best attributes from the original dataset based on the dependencies between these attributes, the ICBA algorithm begins the rule generation process.

Algorithm 2 uses the D' and T' as input for this phase, where D' is the dataset selected by the chi-square method and T' is the training data. The first step in this algorithm is to compute the minimum HM value based on the given minimum-support and minimum-confidence. The ICBA algorithm then generates the single itemset and computes the support, confidence, and HM values for each item (Line 6).

In the next step, the ICBA algorithm generates the single item rules: the generated rules should have an HM value greater than or equal to the minimum HM value (Line 9) and the others will be used to generate the next item rules. Finally, the ICBA algorithm evaluates the remaining items based on the support and confidence: the items that have support and confidence less than the minimum-support and minimum-confidence will be eliminated (Line 12) and the others will be used to generate the itemset. This process will be repeated until S is empty (Line 20).

A good reason to employ the HM measure in the first phase in the association classification technique is to overcome the problem of the given measures that are used by the AC techniques. In AC algorithms, rules that have confidence or support less than the estimated measures, even by very slight values, will be eliminated. As an example, if the minimum-confidence = 0.5 and the minimum-support = 0.3, then if there are rules with support = 0.29 and confidence = 0.8 (or vice versa), these rules will be eliminated. Therefore, the HM measure is used by the ICBA algorithm to produce a harmonic value.

Furthermore, using the HM measure instead of support and confidence will lead to the generation of general rules. For example, if we have three rules such as $a \rightarrow T$, $a, b \rightarrow T$ and $a, b, c \rightarrow T$, we can observe that $a \rightarrow T$ covers $a, b \rightarrow T$ and $a, b, c \rightarrow T$, so there is no need to generate these rules.

Algorithm 2: Rule generation

1 Input: Dataset D' with T' training objects

Minimum-support and Minimum-confidence

2 $\text{Minimum HM value} = 2 * \frac{\text{Minimum-} * \text{Minimum-confidence}}{\text{Minimum} + \text{Minimum-confidence}}$

3 $\text{item\_size} = 0$

4 item_size = item_size+1

5 Find the candidate item_size itemset S

6 For each item in S

$\text{Support (item)} = \text{suppcount (item)}/\text{T'}$

$\text{Confidence (item)} = \text{suppcount (item)}/\text{actoccr (item)}$

$$\text{HM value (item)} = 2 * \frac{\text{support(item)} * \text{confidence(item)}}{\text{support(item)} + \text{confidence(item)}}$$

7 End For

8 For each item in S

9 If HM value(item) >= Minimum HM value

10 Then

11 Add to CARs

12 Else If support (item) < Minimum-support && confidence (item) < Minimum-confidence

13 Remove from the list

14 Else

15 Leave it in the itemset S

16 End For

17 If the itemset S is not empty

6.1.3 Pruning Phase

Algorithm 3 presents how the ICBA algorithm prunes the generated rules based on the HM value. All selected rules are sorted in ascending order based on their HM value; any rules with the same HM values are sorted by confidence value, support value, and the first generated, respectively (Line 2). The first occurrence refers to the rule that has been produced first. Finally, our algorithm removes the conflicting rules based on the class majority criteria, where the majority class is the one with maximum frequency in the dataset.

Algorithm 3: Pruning phase

Pruning (S)

1 {

2 Sort (S) // Sort the generated rules based on the HM values, confidence, support, and first occurrence in descending order

3 If there are conflicting rules

4 Select the rule that has majority class and remove the others

5 Return [rule-set]

6 }

6.1.4 Prediction Phase

The ICBA algorithm predicts the class of an unknown instance i by selecting the rules that match the instance from the rule-set and categorizing these rules based on the class name. The category that has the most rules will be assigned to the instance. If there is more than one category with the same number of rules, the default class will be assigned, where the default class in this context points to the class that has maximum frequency in the

dataset.

6.1.5 Running Example

This example demonstrates how the ICBA algorithm works, and can be applied to any domain under the same phases. To begin, assume there is a dataset (T) for weather as shown in Table 3, a $minimum\_confidence = 0.5$ with a $minimum\_support = 0.2$ and $minimum\ HM\ value = 0.285714$, where

$$Minimum\ HM\ value = 2 * \frac{\text{Minimum-support}*\text{Minimum-confidence}}{\text{Minimum-support}+\text{Minimum-confidence}} = 2 * \frac{0.2*0.5}{0.1+0.5} = 0.285714$$

Table 3. Weather dataset T

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	FALSE	No
Sunny	Hot	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Rainy	Mild	High	FALSE	Yes
Rainy	Cool	Normal	FALSE	Yes
Rainy	Cool	Normal	TRUE	No
Overcast	Cool	Normal	TRUE	Yes
Sunny	Mild	High	FALSE	No
Sunny	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	FALSE	Yes
Sunny	Mild	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Mild	High	TRUE	No

(1) Preprocessing phase

In the preprocessing phase we apply the chi-square method to rank the weather dataset, and the result are shown in Table 4.

Table 4. Ranking values for weather dataset attributes

Ranked attributes:	Name of attribute
3.547	Outlook
2.8	Humidity
0.933	Windy
0.57	Temperature

The cut-off point value of 0.8 is used in the chi-square method, which means that the temperature attribute will be removed and the weather dataset will contain only three attributes, as shown in Table 5.

Table 5. Weather dataset after the ranking process

Outlook	Humidity	Windy	Play
Sunny	High	FALSE	No
Sunny	High	TRUE	No
Overcast	High	FALSE	Yes
Rainy	High	FALSE	Yes
Rainy	Normal	FALSE	Yes
Rainy	Normal	TRUE	No
Overcast	Normal	TRUE	Yes
Sunny	High	FALSE	No
Sunny	Normal	FALSE	Yes
Rainy	Normal	FALSE	Yes
Sunny	Normal	TRUE	Yes
Overcast	High	TRUE	Yes
Overcast	Normal	FALSE	Yes
Rainy	High	TRUE	No

(2) Rule Generation Phase

In this phase, the ICBA algorithm computes the HM measure for each value in the dataset based on the support and confidence values for each candidate rule, as shown in Table 6.

Table 6. Support, confidence, and HM values for the candidate rules

No.	Candidate rule	Support	Confidence	HM measure
1	Sunny → no	0.21	0.6	0.311111
2	Sunny → yes	0.14	0.4	0.207407
3	Overcast → no	0	0	0
4	Overcast → yes	0.285	1	0.44358
5	Rainy → no	0.14	0.4	0.207407
6	Rainy → yes	0.21	0.6	0.311111
7	High → no	0.285	0.57	0.38
8	High → yes	0.21	0.43	0.282188
9	Normal → no	0.07	0.143	0.093991
10	Normal → yes	0.43	0.857	0.572665
11	False → no	0.14	0.25	0.179487
12	False → yes	0.43	0.75	0.54661
13	True → no	0.21	0.5	0.295775
14	True → yes	0.21	0.5	0.295775

It is obvious from Table 6 that rules 2, 3, 5, 8, 9, and 11 are less than the minimum HM value (0.285), so these will be evaluated using the support and confidence values. The rule 8 is the only rule that passes the evaluation process, which means that this rule will be used in the next generation process, while the remaining rules will be removed from the list. Furthermore, the next generation process stops as it contains only one rule.

(3) Pruning Phase

The ICBA algorithm sorts the rules in the CARs based on the minimum HM value, confidence, support, and first occurrence respectively in ascending order, as shown in Table 7.

Table 7. Sorted rules in CARs

Order	Candidate rule	Support	Confidence	HM measure
6	Sunny → no	0.21	0.6	0.311111
3	Overcast → yes	0.285	1	0.44358
5	Rainy → yes	0.21	0.6	0.311111
4	High → no	0.285	0.57	0.38
1	Normal → yes	0.43	0.857	0.572665
2	False → yes	0.43	0.75	0.54661
8	True → no	0.21	0.5	0.295775
7	True → yes	0.21	0.5	0.295775

Regarding the issue of conflicting rules, we can observe there is a conflict between rules 7 and 8 in Table 7, and the ICBA algorithm eliminates rule 8 based on the majority class criteria. As a final result, the CARs contain only seven rules, as shown in Table 8.

Table 8. Final sorted rules in CARs

Order	Candidate rule	Support	Confidence	HM measure
6	Sunny → no	0.21	0.6	0.311111
3	Overcast → yes	0.285	1	0.44358
5	Rainy → yes	0.21	0.6	0.311111
4	High → no	0.285	0.57	0.38
1	Normal → yes	0.43	0.857	0.572665
2	False → yes	0.43	0.75	0.54661
7	True → yes	0.21	0.5	0.295775

The following illustrates the prediction phase:

(1) For the instance "Overcast, High, False," the rules that can classify this instance are: $Overcast \rightarrow yes$, $High \rightarrow no$, and $False \rightarrow yes$. Two of these rules give "yes" and one gives "no." Thus, the class for this instance is "yes" and it is correct as shown in Table 3.

(2) For the instance "Sunny, High, False," the rules that can classify this instance are: $Sunny \rightarrow no$, $High \rightarrow no$, and $False \rightarrow yes$. Two of these rules give "no" and one gives "yes." Thus, the class for this instance is "no" and it is correct as shown in Table 3.

7. Experimental Results

The CBA, MCAR, FACA, and ECBA algorithms are compared against ICBA in terms of accuracy, precision, and recall. We use autism datasets from the UCI repository (Shirabad and Menzies, 2005). To obtain fair results and reduce the error rate, a 270-fold cross-validation process is employed for all experiments where, 70 is number of records or instances in the dataset divided by 10.

All experiments are performed on cluster with 31 nodes (1 master and 30 workers). Specifications of the node is a 3GHz i7 with 32GB main memory and 1TB storage. The CBA, MCAR, FACA, and ECBA algorithms are implemented by their respective authors. The parameters of all algorithms are set as pairs for $minimum\_support$ and $minimum\_confidence$ as follows: (0.1, 0.5), (0.2, 0.5), (0.3, 0.5) and (0.4, 0.5). The ICBA algorithm is executed using the Java programming language under the WEKA tool (Hall et al., 2009).

7.1 Dataset

To test our proposed algorithm, an autism dataset is used from the UCI repository. The dataset contains 21 attributes and 704 instances, where 515 instances have no autism and 189 instances have autism, as follows: $A1\_Score$, $A2\_Score$, $A3\_Score$, $A4\_Score$, $A5\_Score$, $A6\_Score$, $A7\_Score$, $A8\_Score$, $A9\_Score$, $A10\_Score$, age, $gender$, $ethnicity$, $jaundice$, $autism$, $country\_of\_res$, $used\_app\_before$, $result$, age_desc, $relation$ and $class/ASD$. Figures 3, 4, 5, and 6 visualize the distribution of the autism dataset attributes.

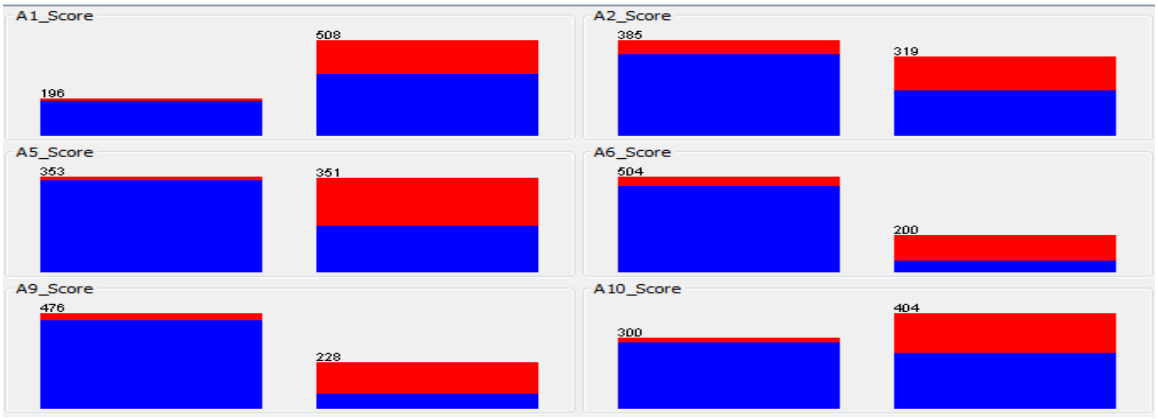

Figure 2. Distribution of $A1\_Score$, $A2\_Score$, $A5\_Score$, $A6\_Score$, $A9\_Score$ and $A10\_Score$ attributes

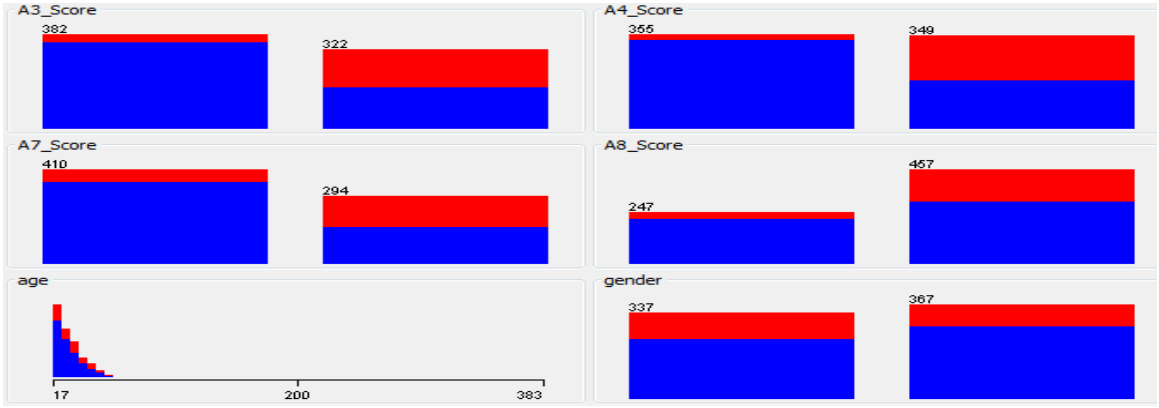

Figure 3. Distribution of $A3\_Score$, $A4\_Score$, $A7\_Score$, $A8\_Score$, age and $gender$ attributes

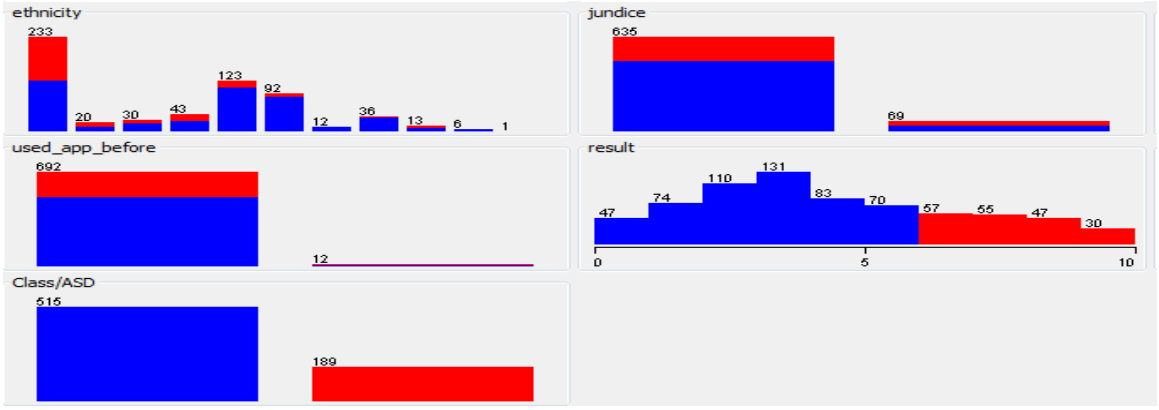

Figure 4. Distribution of $ethnicity$, $jaundice$, $used\_app\_before$, $result$ and $class/ASD$ attributes

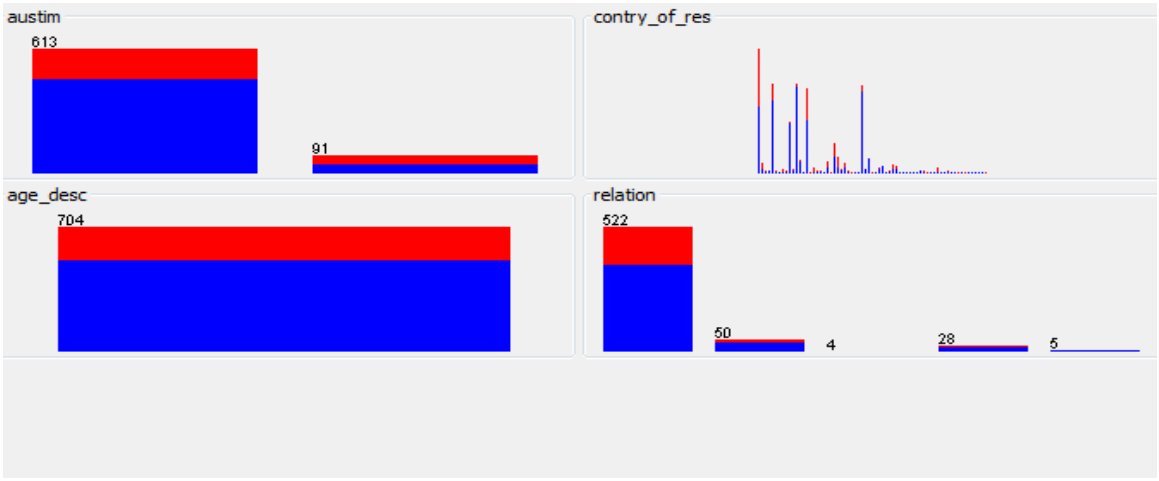

Figure 5. Distribution of *autism*, *country_of_res*, *age_desc* and *relation* attributes

The ICBA algorithm employs the chi-square method to show the correlations between the attributes and their importance, as shown in Table 9.

Table 9. Chi-square scores for autism attributes

Score	Attribute name
704	Result
284.3854	A9_Score
246.8025	A6_Score
203.0151	A5_Score
182.126	country_of_res
155.4773	A4_Score
136.9604	A3_Score
104.8481	A10_Score
86.9455	A7_Score
80.2737	Ethnicity
68.2588	A2_Score
62.3619	A1_Score
39.5966	A8_Score
22.1591	Autism
7.3462	Jaundice
4.5483	Gender
1.3653	used_app_before
1.0624	Relation
0	age_desc
0	Age

According to the chi-square scores, we choose the cut-off point of 10, which leads us to eliminate six attributes: *jaundice*, *gender*, *used_app_before*, *relation*, *age*_desc, and *age*. Therefore, the autism dataset will contain 15 attributes with strong correlation.

7.2 Experiments and Analytical Results

7.2.1 Experiment I: AC Algorithms against ICBA Algorithm Using the Chi-square Method

We compare the ICBA algorithm against four AC algorithms – CBA, MCAR, FACA, and FCBA – based on the

accuracy measure. All these algorithms are tested on the autism dataset after applying the chi-square method on the dataset as the preprocessing phase described in the previous section.

Different values for the $minimum\_support$ and $minimum\_confidence$ are selected to generate four extensive experiments. These values are (0.1, 0.5), (0.2, 0.5), (0.3, 0.5), and (0.4, 0.5), as shown in Table 10 and Figures 6, 7, 8, and 9.

Figure 6 shows the performance of all the considered AC algorithms with $minimum\_support = 0.1$ and $minimum\_confidence = 0.5$. In this experiment, the ICBA algorithm outperforms the AC algorithms in term of accuracy, where the MCAR algorithm in second place and the CBA algorithm in last position.

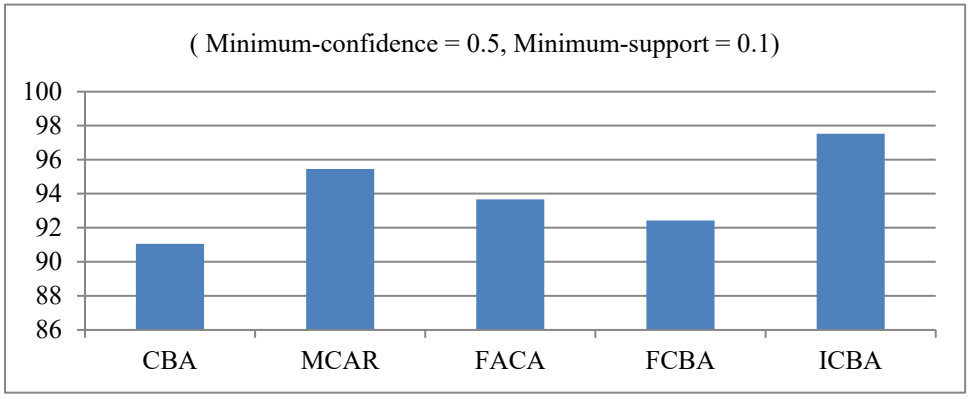

Figure 6. Accuracy of CBA, MCAR, FACA, FCBA, and ICBA algorithms

In the second run, the ICBA and FCBA algorithms are in first place with accuracy of 95.4545%, while the CBA and MCAR are in second place with accuracy of 94.8864%, as shown in Figure 7.

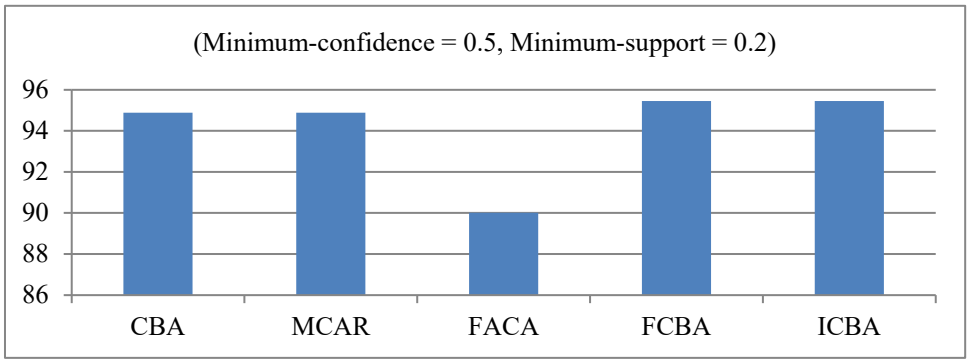

Figure 7. Accuracy of CBA, MCAR, FACA, FCBA, and ICBA algorithms

The best accuracy value for the ICBA occurs in the third experiment: it is in first place with a value 98.6223%. The MCAR is in the second place with accuracy of 91.0511%, while the CBA has the lowest accuracy value of 73.1534%, as shown in Figure 8.

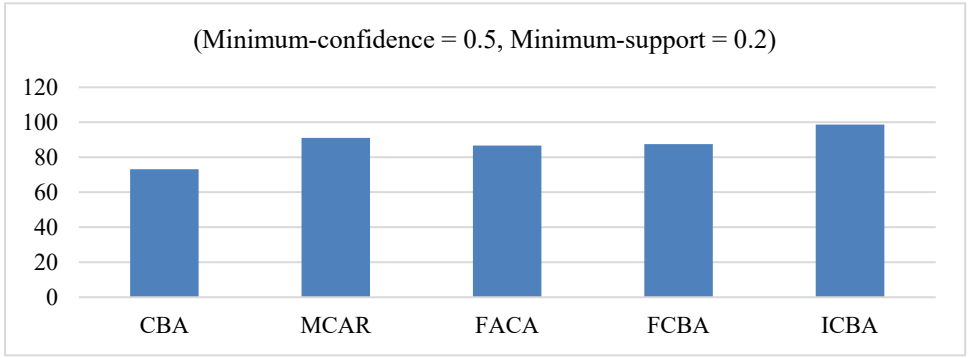

Figure 8. Accuracy of CBA, MCAR, FACA, FCBA, and ICBA

In the final experiment in this section, the ICBA achieves first place with accuracy of 93.608%, as shown in Figure 9. However, if this is compared with the previous experiments, this value is the lowest value for the ICBA algorithm owing to the small number of rules that are generated in the classifier that satisfy the high $minimum\_support$ value.

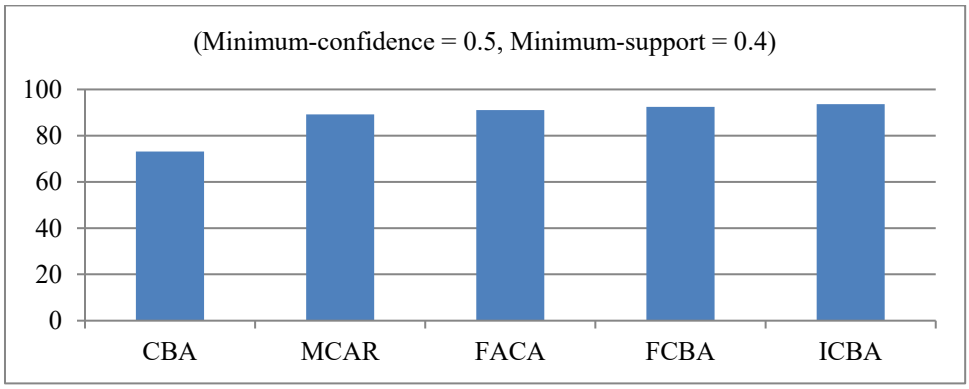

Figure 9. Accuracy of CBA, MCAR, FACA, FCBA, and ICBA

Table 10 summarizes all of these experiments. The ICBA algorithm outperforms the other AC algorithms due to the type of rules that are generated in the classifier. Most of the rules are general rules as mentioned in assumption 1. Assumption 1 employs the HM measure in the first phase, which helps the classifier to generate the rules in the CARs directly.

Table 10. Performance of CBA, MCAR, FACA, FCBA, and ICBA based on accuracy

VALUES OF (MINIMUM-SUPPORT, MINIMUM-CONFIDENCE)	CBA	MCAR	FACA	FCBA	ICBA
(0.1, 0.5)	91.0511	95.4545	93.6642	92.4260	97.5201
(0.2, 0.5)	94.8864	94.8864	90.0086	95.4545	95.4545
(0.3, 0.5)	73.1534	91.0511	86.6477	87.4862	98.6223
(0.4, 0.5)	73.1534	89.2045	91.0511	92.4260	93.608

7.2.2 Experiment II: AC Algorithms with/without Chi-square

To show the impact of using the chi-square method on the considered AC algorithms, we compare the original AC algorithms that do not use the chi-square method with those that use chi-square. In this experiment, we use different values for the $minimum\_support$ and $minimum\_confidence$ from those in Experiment I to generate extensive analysis for these algorithms.

Figure 10 shows the improvement in the performance of the CBA algorithm in the first three runs with the use of the chi-square method, while achieving the same accuracy value when applied with $minimum\_support = 0.4$ and $minimum\_confidence = 0.5$. This experiment reflects the positive impact on the CBA algorithm in term of accuracy.

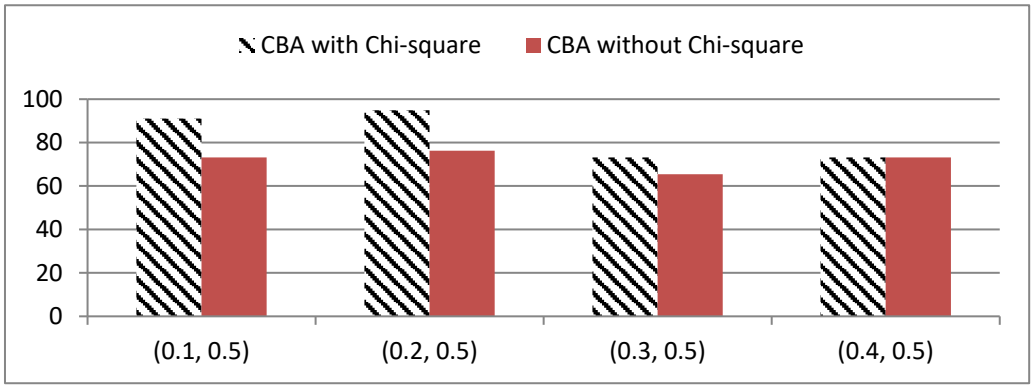

Figure 10. Accuracy of original and modified CBA algorithm

Meanwhile, the MCAR algorithm is enhanced when using the chi-square method in all runs except the third one, with the same accuracy value, as shown in Figure 11.

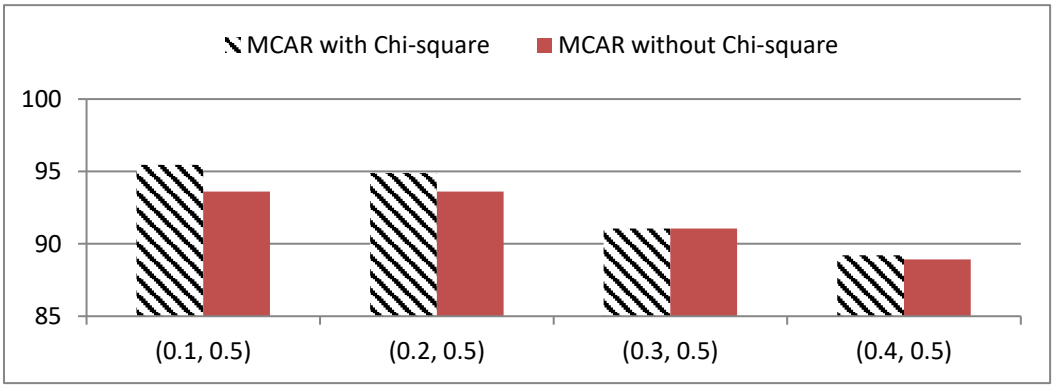

Figure 11. Accuracy of original and modified MCAR algorithm

The FACA algorithm also achieves improved performance when the chi-square method is used. When the *minimum_support* value is 0.1, 0.3, and 0.4, the FACA algorithm achieves better performance using the chi-square method than without using this method, as shown in Figure 12.

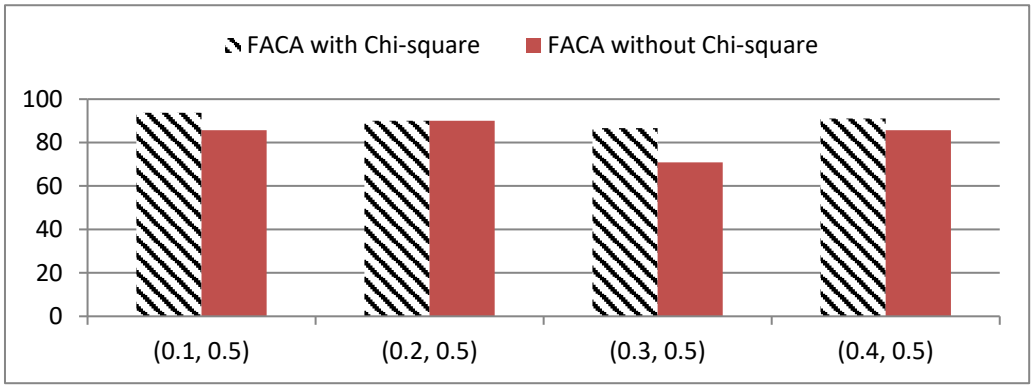

Figure 12. Accuracy of original and modified FACA algorithm

The most striking results are shown in Figure 13, which shows the improved performance of the FCBA algorithm in all runs when using the chi-square method.

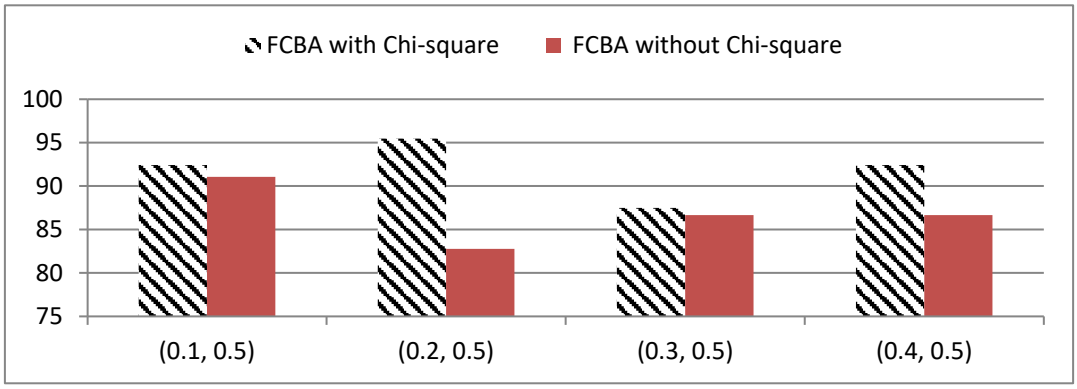

Figure 13. Accuracy of original and modified FCBA algorithm

Our proposed algorithm is affected when the chi-square method eliminates the preprocessing phase. The ICBA algorithm is negatively affected in three runs, while it achieves the same accuracy value in the second run, as shown in Figure 14.

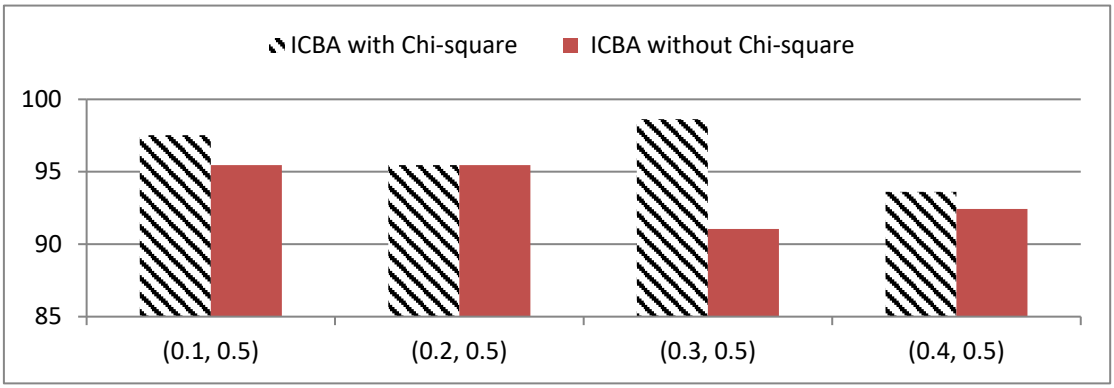

Figure 14. Accuracy of original and modified ICBA algorithm

Furthermore, the ICBA algorithm outperforms the considered AC algorithms in most runs both with and without the chi-square method, due to the type of rules generated in the classifier and the voting technique used in the prediction phase, as shown in Table 11.

Table 11. Performance of original and modified CBA, MCAR, FACA, FCBA, and ICBA algorithms based on accuracy (%)

Values of (Minimum-support, Minimum-confidence)	CBA with chi-square	CBA without chi-square	MCAR with chi-square	MCAR without chi-square	FACA with chi-square	FACA without chi-square	FCBA with chi-square	FCBA without chi-square	ICBA with chi-square	ICBA without chi-square
(0.1, 0.5)	91.0511	73.1534	95.4545	93.608	93.6642	85.6624	92.4260	91.0511	97.5201	95.4545
(0.2, 0.5)	94.8864	76.2405	94.8864	93.608	90.0086	90.0086	95.4545	82.7603	95.4545	95.4545
(0.3, 0.5)	73.1534	65.420	91.0511	91.0511	86.6477	70.8862	87.4862	86.6477	98.6223	91.0511
(0.4, 0.5)	73.1534	73.1534	89.2045	88.9205	91.0511	85.6624	92.4260	86.6477	93.608	92.4260

To identify the reasons behind the good performance achieved using the chi-square method in the AC algorithm, we show the top rules that are generated in the ICBA algorithm in the final classifier both with and without the chi-square method (Figures 15 and 16).

Figure 15 shows the top rules that are used in the classifier without using the chi-square method. We can observe from that the generated rules contain many of attributes that are eliminated using the chi-square method due their weak relationship with other attributes and their possible effect on the classification process, such as age, $age_$desc, $jaundice$ and $used\_app\_before$. Figure 16 shows how the ICBA algorithm eliminates these weak attributes from the top rules that are generated in the classifier, leading to increased accuracy of the classifier.

Furthermore, most of the rules that are generated in the ICBA classifier have a small number of attributes, which reflects the correctness of assumption 2. Assumption 2 illustrates how the ICBA algorithm generates general rather than specific rules that could cover a huge number of instances in the dataset.

```
Classification Rules (ordered):
========================

1.      A1_Score=0 0 0 austim=no 5 0 jundice=no 4 0  ==> Class/ASD=NO     conf:(0.95),  (156),  (0),

2.      A1_Score=0 0 0 austim=no 5 0 jundice=no 4 0 age_desc=18 and more 7 0  ==> Class/ASD=NO     conf:(0.95),  (156),  (1),

3.      A1_Score=0 0 0 austim=no 5 0 jundice=no 4 0 age='All' 1 0  ==> Class/ASD=NO     conf:(0.95),  (154),  (2),

4.      A1_Score=0 0 0 austim=no 5 0 jundice=no 4 0 age='All' 1 0 age_desc=18 and more 7 0  ==> Class/ASD=NO     conf:(0.95),  (154),  (3),

5.      A1_Score=0 0 0 austim=no 5 0 jundice=no 4 0 used_app_before=no 6 0  ==> Class/ASD=NO     conf:(0.95),  (153),  (4),
```

Figure 15. The top rules generated by the ICBA algorithm without using the chi-square method

```
Classification Rules (ordered):
=========================

1.      result='(-inf-6.5]' 12 0  ==> Class/ASD=NO     conf:(1),  (515),  (0),

2.      result='(-inf-6.5]' 12 0 austim=no 11 0  ==> Class/ASD=NO     conf:(1),  (467),  (1),

3.      A6_Score=0 5 0 result='(-inf-6.5]' 12 0  ==> Class/ASD=NO     conf:(1),  (452),  (2),

4.      A9_Score=0 8 0 result='(-inf-6.5]' 12 0  ==> Class/ASD=NO     conf:(1),  (441),  (3),

5.      A6_Score=0 5 0 result='(-inf-6.5]' 12 0 austim=no 11 0  ==> Class/ASD=NO     conf:(1),  (410),  (4),

6.      A9_Score=0 8 0 result='(-inf-6.5]' 12 0 austim=no 11 0  ==> Class/ASD=NO     conf:(1),  (401),  (5),
```

Figure 16. The top rules generated by the ICBA algorithm using the chi-square method

Figures 17, 18, and 19, show how the general rules could give accurate prediction in the classifier. Figure 17 presents the distribution of $A3\_score$, $jaundice$ attributes on the class attribute that has two values ("No" that represents blue color and "YES" that represents red color). Where, class (no) could predicted if $A3\_score$ attribute has value with label 0 and $jaundice$ attribute has value with label 0 as shown in lower left angle in the figure.

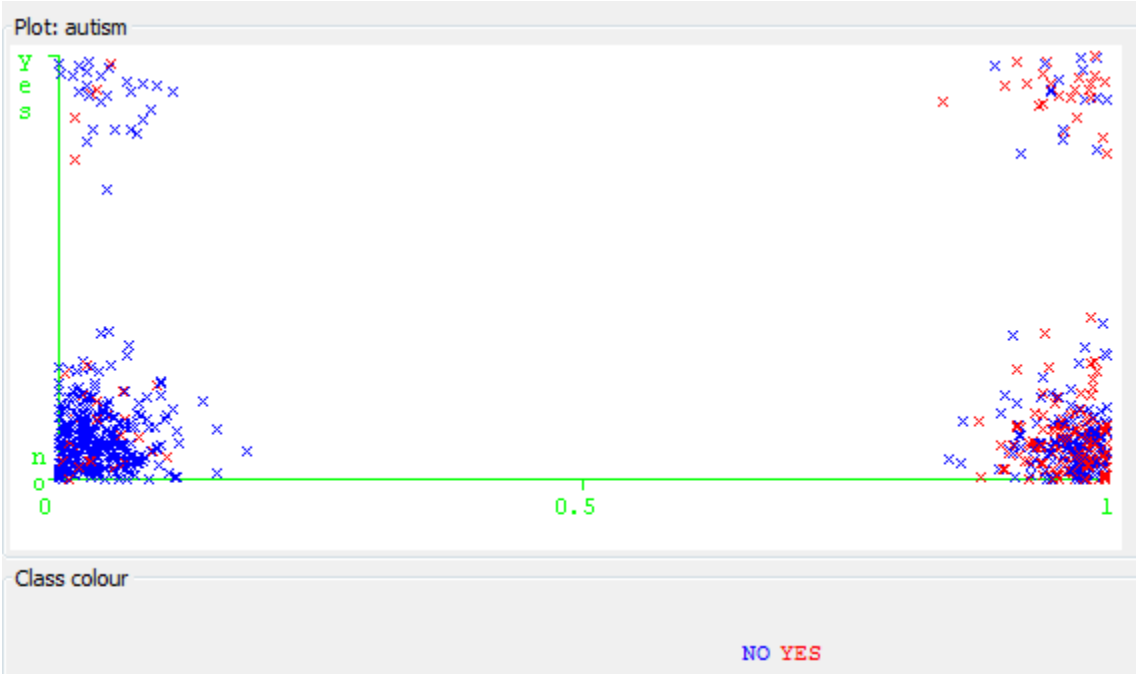

Figure 17. Distribution of attributes A3_score and jaundice based on class value

The same issue emphasized in Figures 18 and 19, where Figure 18 generates two rules: $\text{autism(no)}, \text{result}(0) \rightarrow \text{class(no)}$ and $\text{autism(no)}, \text{result}(10) \rightarrow \text{class(yes)}$.

While Figure 19 generates one rule: $\text{Relation(self)}, \text{A10\_score}(0) \rightarrow \text{class(no)}$. Finally, the visualization process clarifies how the generated rules with small number of attributes have high confidence and support values, and this leads to enhance the accuracy level of the classifier.

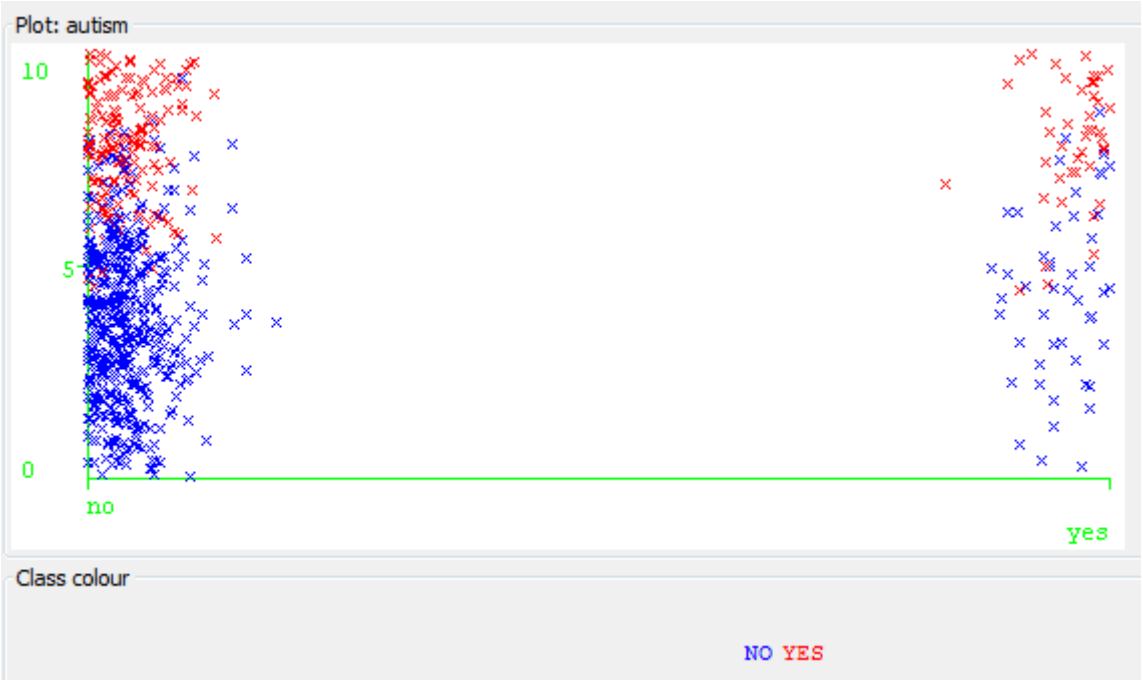

Figure 18. Distribution of attributes (Autism and result) based on class value

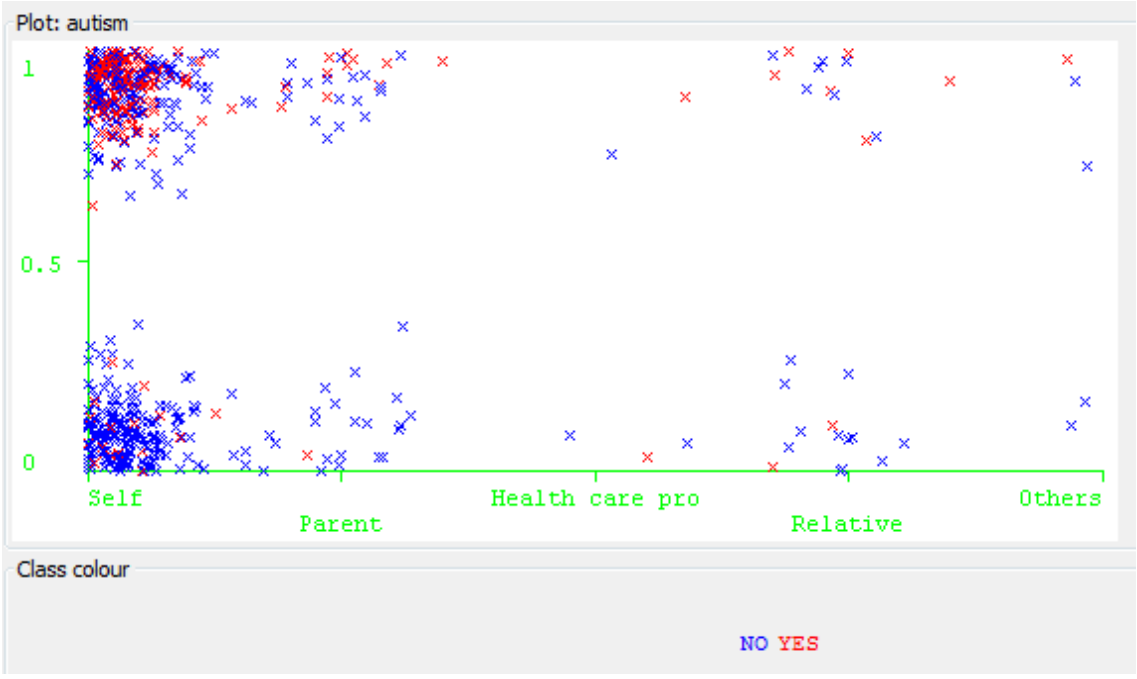

Figure 19. Distribution of attributes (Relation and A10_score) based on class value

8. Conclusions and Future Work

Data mining techniques can be used to improve the decision-making process in many critical areas, such as the medical field, website phishing, text analysis, social media, and many others. The AC techniques are of the most

important techniques in data mining that use association rules in the classification process to enable more accurate decisions to be taken in many areas. The main challenge faced by this technique is that of obtaining a high level of accuracy. The proposed ICBA algorithm was built on two main factors: differentiating between the attributes by using a statistical approach, and generating general rules by using the harmonic mean measure to select the stronger association rules. Both of these factors contribute to the improved decision-making process in the field of ASD discovery in terms of accuracy measures. The proposed algorithm was compared against four well-known AC algorithms in terms of accuracy to evaluate their behavior in the prediction task using big data platform. It is worth mentioning that the proposed algorithm showed outstanding performance in all experiments.

References

Abdallat, A. A., Alahmad, A. I., & AlWidian, J. A. (2019). Hadoop mapreduce job scheduling algorithms survey and use cases. *Modern Applied Science, 13*(7), 1-38. https://doi.org/10.5539/mas.v13n7p38

Abdelhamid, N., Ayesh, A., & Thabtah, F. (2015). Emerging Trends in Associative Classification Data Mining. *International Journal of Electronics and Electrical Engineering, 3*(1), 50-53. https://doi.org/10.12720/ijeee.3.1.50-53

Abdelhamid, N., & Thabtah, F. (2014). Associative Classification Approaches: Review and Comparison. *Journal of Information and Knowledge Management (JIKM), 13*(3). https://doi.org/10.1142/S0219649214500270

Abu-Mansour, H., Alwidian, J., Hadi, W., & Arabia, R. S. (2012). *Efficient Algorithm for Two Dimensional Pattern Matching Problem (Square Pattern)*. https://doi.org/10.1109/ICITeS.2012.6216622

Abuqabita, F., Al-Omoush, R., & Alwidian, J. (2019). A comparative study on big data analytics frameworks, data resources and challenges. *Modern Applied Science, 13*(7), 1-14. https://doi.org/10.5539/mas.v13n7p1

Ajayi, A., Oyedele, L., Davila Delgado, J. M., Akanbi, L., Bilal, M., Akinade, O., & Olawale, O. (2019). Big data platform for health and safety accident prediction. *World Journal of Science, Technology and Sustainable Development, 16*(1), 2-21. https://doi.org/10.1108/WJSTSD-05-2018-0042

Alazaidah, R., Thabtah, F., & Al-Radaideh, Q. (2015). A Multi-Label Classification Approach Based on Correlations Among Labels. *International Journal of Advanced Computer Science and Applications, 6*(2), 52-59. https://doi.org/10.14569/IJACSA.2015.060208

Al-Fayoumi, M., Alwidian, J., & Abusaif, M. (2020). Intelligent association classification technique for phishing website detection. *International Arab Journal of Information Technology, 17*(4), 488-496. https://doi.org/10.34028/iajit/17/4/7

Alwidian, J., Abu-Mansour, H., & Ali, M. (2012, March). Efficient algorithm for two dimensional pattern matching problem (non-square pattern). In 2012 *International Conference on Information Technology and e-Services* (pp. 1-8). IEEE. https://doi.org/10.1109/ICITeS.2012.6216622

Alwidian, J., Elhassan, A., & Ghnemat, R. (2020). Predicting autism spectrum disorder using machine learning technique. *International Journal of Recent Technology and Engineering, 8*(5), 4139-4143. https://doi.org/10.35940/ijrte.E6016.018520

Alwidian, J., & Hadi, W. (2012, March). Enhancing the results of UCP in cost estimation using new external environmental factors. In 2012 *International Conference on Information Technology and e-Services* (pp. 1-11). IEEE. https://doi.org/10.1109/ICITeS.2012.6216623

Alsahlee, O. H., Al-Zu'bi, A., Alamro, I. I., & Alwidian, J. (2019, July). Distributed and automated machine learning in big data stream analytics. In 2019 *Third World Conference on Smart Trends in Systems Security and Sustainablity (WorldS4)* (pp. 307-313). IEEE. https://doi.org/10.1109/WorldS4.2019.8903932

Alwedyan, J., Hadi, W. E. M., Salam, M. A., & Mansour, H. Y. (2011, April). Categorize arabic data sets using multi-class classification based on association rule approach. In *Proceedings of the 2011 International Conference on Intelligent Semantic Web-Services and Applications* (pp. 1-8). https://doi.org/10.1145/1980822.1980840

Alwedyan, J., Wa'el Musa Hadi, M. A. Salam, & Hussein Y. Mansour. (2011). Categorize Arabic data sets using multi-class classification based on association rule approach. *Proceedings of ISWSA 2011,* Amman, Jordan, 18. https://doi.org/10.1145/1980822.1980840

Alwidian, J., Hammo, B.H., & Obeid, N. (2018). WCBA: Weighted classification based on association rules algorithm for breast cancer disease. *Applied Soft Computing, 62*, 536-549. https://doi.org/10.1016/j.asoc.2017.11.013

Alwidian, J., Hammo, B., & Obeid, N. (2016). FCBA: Fast Classification Based on Association Rules Algorithm. *International Journal of Computer Science and Network Security (IJCSNS), 16*(12), 117.

Alwidian, J., Hammo, B., & Obeid, N. (2016). Enhanced CBA algorithm based on apriori optimization and statistical ranking measure. In *Proceeding of 28th international business information management association (IBIMA) conference on vision* (Vol. 2020, pp. 4291-4306).

Alwidian, J., Rahman, S. A., Gnaim, M., & Al-Taharwah, F. (2020). Big Data Ingestion and Preparation Tools. *Modern Applied Science, 14*(9), 12-27. https://doi.org/10.5539/mas.v14n9p12

Antonie, M.L., & Zaïane, O.R. (2004). An associative classifier based on positive and negative rules. In: *Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, June, 2004, 64–69. ACM. https://doi.org/10.1145/1008694.1008705

Baralis, E., Chiusano, S., & Garza, P. (2004). On support thresholds in associative classification. In: *Proceedings of the 2004 ACM Symposium on Applied Computing,* March, 2004, 553–558. ACM. https://doi.org/10.1145/967900.968016

Bolton, P., Macdonald, H., Pickles, A., Rios, P., Goode, S., Crowson, M., Bailey, A., & Rutter, M. (1994). A case-control family history study of autism. *Journal of Child Psychology & Psychiatry, 35*, 877-900. https://doi.org/10.1111/j.1469-7610.1994.tb02300.x

Bone, D., Goodwin, M.S., Black, M.P., Lee, C.-C., Audhkhasi, K., & Narayanan, S. (2014). Applying Machine Learning to Facilitate Autism Diagnostics: Pitfalls and Promises. *Journal of Autism and Developmental Disorders,* 1-16. https://doi.org/10.1007/s10803-014-2268-6

Dong, G., Zhang, X., Wong, L., & Li, J. (1999). CAEP: Classification by aggregating emerging patterns. In: *Discovery Science, December,* 1999, 30-42. Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-46846-3_4

Dou, Y., Fang, H., Ju, X., Zhang, H., Zhang, T., & Li, D. (2019, March). Reliability analysis method of aerospace equipment system based on big data. In *Fifth Symposium on Novel Optoelectronic Detection Technology and Application* (Vol. 11023, p. 110234U). International Society for Optics and Photonics. https://doi.org/10.1117/12.2519534

Duda, M., Ma, R., Haber, N., & Wall, D. P. (2016). Use of machine learning for behavioral distinction of autism and ADHD. *Translational Psychiatry, 9*(6), 732. https://doi.org/10.1038/tp.2015.221

Hadi, W. (2013). EMCAR: Expert Multi Class Based on Association Rule. *International Journal of Modern Education and Computer Science, 5*(3), 33-41. https://doi.org/10.5815/ijmecs.2013.03.05

Hadi, W., Aburub, F., & Alhawari, S. (2016). A new fast associative classification algorithm for detecting phishing websites. *Applied Soft Computing, 48*, 729-734. https://doi.org/10.1016/j.asoc.2016.08.005

Hadi, W., Al-Widian, J., & Alhawari, S. (2013). An integrated model for knowledge management and electronic customer relationship management. *Journal of American Science, 9*(11), 440-447.

Hadi, W. E. M., Salam, M. A., & Al-Widian, J. A. (2010, June). Performance of NB and SVM classifiers in Islamic Arabic data. In *Proceedings of the 1st International Conference on Intelligent Semantic Web-Services and Applications* (pp. 1-6). https://doi.org/10.1145/1874590.1874604

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I.H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter, 11*(1), 10-18. https://doi.org/10.1145/1656274.1656278

Jr S, Pappa C, Freitas A, & Kaestner C. (2014). Automatic text summarization with genetic algorithm-based attribute selection. *Adv Artif Intell– IBERAMIA Springer, 2004*, 305-14. https://doi.org/10.1007/978-3-540-30498-2_31

Li, W., Han, J., & Pei, J. (2001). CMAR: Accurate and efficient classification based on multiple class-association rules. In: Data Mining, ICDM 2001, *Proceedings IEEE International Conference,* November, 2001, 369-376. IEEE.

Li, X., Qin, D., & Yu, C. (2008). ACCF: Associative classification based on closed frequent itemsets. In: *Fuzzy Systems and Knowledge Discovery,* FSKD'08, Fifth International Conference, October, 2008, 380–384. IEEE. https://doi.org/10.1109/FSKD.2008.396

Liu, B., Hsu, W., & Ma, Y. (1998). Integrating classification and association rule mining. In: *Proceedings of the*

fourth international conference on knowledge discovery and data mining, August, 1998, New York, NY, 80–86.

Lord, C., Risi, S., … Lambrecht, L. (2000). The Autism Diagnostic Observation Schedule-Generic: a standard measure of social and communication deficits associated with the spectrum of autism. *Journal of Autism and Developmental Disorders, 30*, 205-223.7. https://doi.org/10.1023/A:1005592401947

Lord, C., Rutter, M., & Le Couteur, A. (1994). Autism Diagnostic Interview-Revised: A revised version of a diagnostic interview for caregivers of individuals with possible pervasive developmental disorders. *Journal of Autism and Developmental Disorders, 24*, 659-685. https://doi.org/10.1007/BF02172145

Ma, B., Zhang, H., Chen, G., Zhao, Y., & Baesens, B. (2014). Investigating Associative Classification for Software Fault Prediction: An Experimental Perspective. *International Journal of Software Engineering and Knowledge Engineering, 24*(01), 61-90. https://doi.org/10.1142/S021819401450003X

Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., ... Xin, D. (2016). Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research, 17*(1), 1235-1241.

Nagarajan, G., & Babu, L. (2019). An Empirical Comparison of Six Supervised Machine Learning Techniques on Spark Platform for Health Big Data. In *Smart Intelligent Computing and Applications* (pp. 299-307). Springer, Singapore. https://doi.org/10.1007/978-981-13-1927-3_32

Mohammad, R., Thabtah, F., & McCluskey L. (2014). Predicting Phishing Websites based on Self-Structuring Neural Network. *Journal of Neural Computing and Applications,* (3), 1-16. Springer. https://doi.org/10.1007/s00521-013-1490-z

Nguyen, D., Nguyen, L., Vo, B., & Pedrycz, W. (2016). Efficient mining of class association rules with the itemset constraint. *Knowledge-Based Systems, 103*, 73-88. https://doi.org/10.1016/j.knosys.2016.03.025

Platt, J. (1998). Fast training of support vector machines using sequential optimization. In: *B.* Scholkopf, C. Burges, A. Smola (Eds.), *Advances in Kernel Methods – Support Vector Learning*. Cambridge, MA: MIT Press, pp. 185-208. https://doi.org/10.7551/mitpress/1130.003.0016

Qian, T., Wang, Y., Long, H., & Feng, J. (2005). 2-ps based associative text classification. In: *Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery*, August, 2005, pp. 378–387. Berlin, Heidelberg: Springer. https://doi.org/10.1007/11546849_37

Quinlan, J. (1993). *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann.

Salah, H., Al-Omari, I., Alwidian, J., Al-Hamadin, R., & Tawalbeh, T. (2019). Data streams curation for better machine learning functionality and result to serve IoT and other applications: a survey. *Journal of Computer Science, 15*(10), 1572-1584. https://doi.org/10.3844/jcssp.2019.1572.1584

Shahin, D., Hannen Ennab, R. S., & Alwidian, J. (2019). Big Data Platform Privacy and Security, A Review. *IJCSNS, 19*(5), 24.

Shorfuzzaman, M., Hossain, M. S., Nazir, A., Muhammad, G., & Alamri, A. (2019). Harnessing the power of big data analytics in the cloud to support learning analytics in mobile learning environment. *Computers in Human Behavior, 92*, 578-588. https://doi.org/10.1016/j.chb.2018.07.002

Tan, P.N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining* (Vol. 1). Boston: Pearson Addison Wesley.

Tang, Z., & Liao, Q. (2007). A New Class Based Associative Classification Algorithm. *IMECS, 36*(2), 685-689.

Thabtah, F. (2007). A Review of Associative Classification Mining. *Journal of Knowledge Engineering Review, 22*(1), 37-65. https://doi.org/10.1017/S0269888907001026

Thabtah, F. (2017, May). Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment. In *Proceedings of the 1st International Conference on Medical and Health Informatics 2017* (pp. 1-6). ACM. https://doi.org/10.1145/3107514.3107515

Thabtah, F., Cowling, P., & Peng, Y. (2005). MCAR: multi-class classification based on association rule. In: *Computer Systems and Applications*, the 3rd ACS/IEEE International Conference, January, 2005, pp. 33–40. IEEE.

Thabtah, F., Hadi, W., Abdelhamid, N., & Issa, A. (2011). Prediction Phase in Associative Classification Mining. *International Journal of Software Engineering and Knowledge Engineering, 21*(06), 855-876. https://doi.org/10.1142/S0218194011005463

Vavilapalli, V. K., Murthy, A. C., Douglas, C., Agarwal, S., Konar, M., Evans, R., ... Saha, B. (2013, October). Apache hadoop yarn: Yet another resource negotiator. In *Proceedings of the 4th annual Symposium on Cloud Computing* (p. 5). ACM. https://doi.org/10.1145/2523616.2523633

Wa'el Musa Hadi, Ma'an Salam, & Al-Widian, J. A. (2010). Performance of NB and SVM classifiers in Islamic Arabic data. In *ISWSA* (p. 14). https://doi.org/10.1145/1874590.1874604

Wall, D.P., Kosmicki, J., Deluca, T.F., Harstad, L., & Fusaro, V.A. (2012a). Use of Machine Learning to Shorten Observation-Based Screening and Diagnosis of Autism. *Translational Psychiatry,* (2). https://doi.org/10.1038/tp.2012.10

Wall, D.P., Dally, R., Luyster, R., Jung, J.Y., & Deluca, T.F. (2012b). Use of Artificial Intelligence to Shorten the Behavioural Diagnosis of Autism. *PLoS ONE, 7*, e43855. https://doi.org/10.1371/journal.pone.0043855

Xing, E. P., Ho, Q., Dai, W., Kim, J. K., Wei, J., Lee, S., ... Yu, Y. (2015). Petuum: A new platform for distributed machine learning on big data. *IEEE Transactions on Big Data, 1*(2), 49-67. https://doi.org/10.1109/TBDATA.2015.2472014

Xu, X., Han, G., & Min, H. (2004). A novel algorithm for associative classification of image blocks. In: *Computer and Information Technology*, CIT'04, Fourth International Conference, September, 2004, 46–51. IEEE.

Yin, X., & Han, J. (2003). CPAR: Classification based on Predictive Association Rules. In: *Proceedings of the 2003 SIAM International Conference on Data Mining*, May, 2003, 331-335. https://doi.org/10.1137/1.9781611972733.40