FCBA: Fast Classification Based on Association Rules Algorithm

Jaber Alwidian, Bassam Hammo, and Nadim Obeid

Department of Computer Science King Abdulla II School for Information Technology The University of Jordan, Amman

Summary
Many critical applications – such as medical diagnosis, text analysis, website phishing, and many others – need an artificial automated tool to enhance the decision-making process. Employing association rules in the classification process is one technique in the data-mining field for making more accurate and critical decisions. This is known as the association classification (AC) technique .However, most of the AC algorithms are not scalable as they are affected by the size of the dataset. Furthermore, the issue of the algorithm's level of accuracy versus the time needed to build the model is critical; some AC algorithms have a high level of accuracy but take a long time to build a model, while the others take short time to build a model but have a low level of accuracy. To address these problems, we propose in this paper, a Fast Classification Based on Association Rules (FCBA) algorithm based on new internal and external pruning methods to generate association rules using an enhanced Apriori algorithm. We compare our proposed algorithm with four well-known AC algorithms, namely the CBA, CMAR, MCAR and FACA algorithms, based on 11 UCI datasets. Most of the datasets are medical and of different sizes. This allows us to evaluate the scalability and accuracy of the algorithms. Our extensive experimental study shows that the FCBA algorithm is more scalable than the others. In addition, the FCBA algorithm outperforms the others with regard to accuracy and the time taken to build the model. FCBA is ranked first in 64% and second in 36% of datasets, with an average time of less than 0.01 seconds. Thus, it achieves the highest accuracy and the fastest average time to build the model, in comparison with the other algorithms. In the medical datasets, FCBA performs better, wins in 67% of datasets and is second place in 33%, with an average time of less than 0.01 seconds.

Keywords:
Data mining, Association Classification, Apriori, Medical diagnosis

1. Introduction

Data mining is a subfield of computer science that aims to discover hidden patterns or knowledge in large datasets, and to transform this extracted information into a more suitable form to enhance the decision-making process in many fields. Data mining includes a set of techniques for different purposes, such as classification, clustering, regression, association rules, and association classification (AC) (Abdelhamid et al., 2014; Ma et al., 2014; Taware et al., 2015). In this paper, we will shed light on the AC technique and show how it can be used to enhance the decision-making process.

The understandability and simplicity of the rules that can be generated using the AC technique and its positive effect on the accuracy of the classification process or decision-making process make the AC technique attractive for researchers. However, the AC mining technique does have a disadvantage, namely that it generates a very large number of rules, requiring more memory and time than the classical data-mining techniques. Furthermore, many of the AC algorithms do not behave in a stable way in all datasets, so may not be scalable owing to this negative aspect (Tan et al., 2006; Hadi, 2013; Abdelhamid et al., 2015).

To evaluate our proposed model and set of well-known AC algorithms, we focused on two critical evaluation measures: accuracy and building time for the model. The accuracy reflects the level of enhancement of the decision-making process, while the building time for the model is considered the main challenge for data mining. This is particularly the case for the AC technique for all real-world applications that depend on an incremental learning approach. In the incremental learning approach, any new instance can affect the classification model and its accuracy. Thus, most of these applications – such as online transactions, banking, medicine, retail marketing, and stock market exchanges – suffer from rapid changes in the data that require the classification model to be rebuilt within an acceptable timeframe for any changes in the data, in order to obtain more accurate results (Gupta et al., 2005; Alnababteh et al., 2014).

In the incremental learning approach, once a new instance is added to the dataset, the AC technique deals with this situation in one of two ways. In the first, the AC technique uses the original dataset without including the new instance and its effect on the dataset. Therefore, the classifier cannot reflect the latest changes on the dataset, and this leads to a reduction in the accuracy of the classifier that’s means, the support of the rules in this type of applications is the most important (i.e. if we have a set of confident rules then the rules that have highest support will get highest priority than the others). The second scenario involves rebuilding the classifier for each new instance or class. However, this scenario needs a full scan of the training dataset to reflect new changes in the classifier, and the process needs to be executed rapidly if it is to be acceptable (Nababteh et al., 2010; Alnababteh et al., 2014).

Manuscript received December 5, 2016
Manuscript revised December 20, 2016

The main aim of this paper is to build a more efficient intelligent AC technique and apply it to a set of UCI datasets, most of which will be selected from the medical field, for example breast-cancer, iris, and liver disorder datasets. A comprehensive experimental study using UCI datasets will be presented to evaluate and compare well-known association rule-based classification techniques with our proposed technique in terms of accuracy, F1, recall, precision, and building time for the model. Furthermore, our study aims to meet the following objectives:

- Conduct a comprehensive and significant study on some aspects of AC data-mining techniques.
- Develop scalable and accurate AC algorithms.
- Produce extensive experimental results to evaluate the proposed AC technique with regard to different datasets from the UCI repositories.

The main of the AC concepts are presented in section 2. In Section 3, some relevant works are discussed. We describe our proposed model in detail in Section 3. Extensive experimental results are given in Section 4. Finally, the conclusion and discussion of future work are presented in section 6.

2. AC Background

The AC technique is a combination of the association rules and classification techniques. The association rules technique aims to discover a correlation or association between attributes, while the classification process is responsible for predicting the class label. Thus, the AC technique represents the second generation of the association rules technique, and is designed to find the correlation between attributes and classes. For example, in a rule such as $At_1, At_2 \rightarrow C_1$, C_1 must be a class attribute, while At_1 and At_2 are attribute values. This rule can be interpreted as meaning that if At_1 and At_2 attribute values occur together for any object; this object can be classified as C_1,which represents the class value (Liu et al., 1998; Abdelhamid et al., 2014; Abdelhamid et al., 2015).

The formal description of the AC problem is stated by Thabtah et al. (2006). We use the dataset (T) shown in Table 1.

Table 1: Dataset sample (T) with four training objects

Training object	Attribute 1 (At_1)	Attribute 2 (At_2)	Attribute 3 (At_3)	Class (C)
1	v_1	v_3	v_5	C_1
2	v_1	v_3	v_6	C_1
3	v_1	v_3	v_6	C_2
4	v_2	v_4	v_7	C_3

In the AC problem, the association rules are employed in the classification process. If a rule states that $At_1 \rightarrow C_1$, then C_1 has to be a class attribute. The training data set T has *m* distinct attributes (At_1, At_2… At_m), and C is a list of classes. Attributes could be categorical or continuous. In the case of categorical attributes, all possible values are mapped to a set of positive integers, while continuous attributes use any discretization method. A row or a training object in T can be described as a combination of attribute names At_i and values v_i, plus a class denoted by C_j, and the item can be described as an attribute name At_i and value v_i.As shown in Table 1, (At_1, v_1) is an item; an itemset is a set of items contained in a training object, for example, (At_1, v_1) (At_2, v_3).

A ruleitem *r* is of the form < itemset, C_i>, where c_i is the class; as an example, in Table 1, training object 1 contains < (At_1, v_1) (At_2, v_3), C_1> as a ruleitem. The actual occurrence (actoccr) of a ruleitem *r* in T is the number of rows in T that match the itemsets defined in *r*; thus, for (At_1, v_1) (At_2, v_3) as itemset, the actoccr = 3. Based on that, the support count (suppcount) of ruleitem *r* is the number of rows in T that match *r*'s itemsets, and belong to a class C_i for *r*, as shown in equation (1).

$$\mathbf{Suppconut = r \cup c_i} \qquad (1)$$

The suppcount for < (At_1, v_1) (At_2, v_3), C_1> ruleitem is 2, which means there are two occurrences of this ruleitem.

A ruleitem *r* passes the minsupp threshold if (suppcount(*r*) / |T|) >= minsupp, where |T| is the number of instances in T, as shown in equation (2).

$$\mathbf{Support} = \frac{\mathbf{(r \cup c_i).count}}{\mathbf{|T|}} \qquad (2)$$

In Table 1, the number of training objects in the dataset T is 4; since the suppcount of the ruleitem < (At_1, v_1) (At_2, v_3), C_1> is 2, the support equals 4/2=2.

A ruleitem *r* passes the minconf threshold if (suppcount(*r*)/actoccr(*r*))>= minconf, as shown in equation (3).

$$\mathbf{Confidence} = \frac{\mathbf{(r \cup c_i).count}}{\mathbf{r.count}} \qquad (3)$$

For < (At_1, v_1) (At_2, v_3), C_1> ruleitem, the suppcount=2 and actoccr=3 so, confidence=2/3.

Any ruleitem *r* that passes the minsupp threshold is said to be a frequent ruleitem, and an actual class association rule is represented in the form: (At_{i1}, v_{i1}) ^ (At_{i2}, v_{i2}) ^…. ^ (At_{1m}, v_{im})$\rightarrow C_j$, where the antecedent of the rule is an itemset and the consequent is a class.

Using an efficient intelligent AC technique to make strategic decisions can reduce time, effort, and risk for any organization. Over the past few years, this has motivated many researchers to become involved in this area, focusing on applications such as medicine, mail order, phishing websites, supermarkets, insurance fraud, telemarketing, and many others, in order to enhance the decision-making process. This has entailed the need to generate new demands in the association and classification problem to produce more accurate results than those obtained using traditional data-mining techniques. To serve these critical applications, the PROMISE (Shirabad and Menzies, 2005) and NASA MDP (Metrics Data Program) (Chapman et al., 2004) repositories have published datasets that are available to all scholars, without fees.

The Classification Based on Association Rules (CBA) algorithm was proposed in (Liu et al., 1998) to employ the association rules in the classification task that produced new generation in the classification process that's called AC technique. This algorithm was built on three phases: rule generation, pruning, and prediction. In the rule generation phase, Apriori algorithm was used to generate the frequent itemset that represent the class association rules (CARs) where, all of these frequent itemset should pass two estimated measures (minimum support and minimum confidence), as shown in following steps:

1. Generate the candidate single itemset. And then generate the frequent single itemset, based on selecting the items that have support greater than or equal to the estimated minimum support. The Support for any item can be calculated by equation (4).

   $$\boldsymbol{Support} = \frac{(\boldsymbol{X}\cup\boldsymbol{Y}).\boldsymbol{count}}{\boldsymbol{n}} \qquad (\mathbf{4})$$

   Where, *x* is the attribute, *y* the name of the class, and *n* the number of rows in the dataset.
2. Generate the candidate 2-itemset.
3. Generate the frequent 2-itemset that passes the minimum support.
4. Repeat to find all next itemsets until the set is empty.
5. Finally, The CARs should be generated from the frequent sets by selecting the rules that have confidence greater than or equal to the estimated minimum confidence, where the confidence of an item can be calculated by equation (5).

$$\mathbf{Confidence} = \frac{(\mathbf{X}\cup\mathbf{Y}).\mathbf{count}}{\mathbf{X}.\mathbf{count}} \qquad (5)$$

After finding the CARs using the Apriori algorithm, the M1 method is used in the pruning phase to select the best rules that cover the entire database. Finally, the prediction phase predicts the class for any given unknown input, the class of the first rule that can match this input will be assigned as its predicted class.

3. Related Works

(Li et al., 2001) proposed Classification based on Multiple Association Rules (CMAR) as a new association and classification algorithm, based on creating a combination of association rules and classification techniques, like other AC algorithms, such as CBA. The novelty of this algorithm is that it adopts new approaches in the rule-generation and classification phases, which are consider the main two phases in this algorithm. In the rule-generation phase, FP-tree and CR-tree are employed to generate rules instead of the Apriori algorithm; in all of these algorithms the first step is the same, namely to find the frequent single itemset. Moreover, minimum support and confidence play the main role in the CMAR and CBA algorithms. The classification phase in the CMAR algorithm depends on finding the label class for its input by finding all rules that can be matched with this input and then analyzing all of these rules to predict the class. In the final step, CMAR is compared with CBA and C4.5 based on the accuracy measure, and the result shows that CMAR performed better than the others.

(Thabtah et al., 2005) proposed the Multi-class Classification based on Association Rule (MCAR) algorithm to overcome the main problem of the CBA algorithm, which is the multi-scanning of the dataset to generate the rules. In MCAR, a single itemset will be generated using the traditional procedure from the CBA algorithm. In addition, the occurrence positions for each item are stored, facilitating the next itemset-generation process without extra scanning of the dataset.

In (Hadi et al., 2016) proposed a new Fast Associative Classification Algorithm (FACA) for predicting phishing websites. In this algorithm, Diffset method has been employed in the rule generation process to enhance the building time model. Furthermore, it sorts the generated rules according to least number of attributes in the left hand side, confidence, support and rule generated first respectively. Finally, FACA employs multiple rules in prediction phase to enhance the accuracy of the classifier. In particular, this algorithm divides the matched rules to set clusters based on their classes and then select the class that has maximum number of rules. The authors compared their algorithm with CBA, CMAR, MCAR and new Enhanced Class Association Rule (ECAR) (Hadi, 2015) algorithms on phishing websites dataset.

CBA, CMAR, MCAR and FACA algorithms employ different data structures in the rule generation process like Apriori, FP-tree, TID-list, and Diffset structures to enhance the building time model. In addition, they use two well-known measures: support and confidence to generate, rank and prune the rules by using different prioritization procedures without taking the application domain in their consideration. All of these issues motivated us to propose a new AC algorithm for specific application domain i.e. incremental applications that need high speed for the building time model and high accuracy for the classification process (Petko et al., 2003; Chang et al., 2005; Nababteh et al., 2010; Alnababteh et al., 2014; Hadi, 2015; Hadi et al., 2016).

In this type of applications, the support measure for the rule is more important than the confidence and this is not mean the rule should not be confident i.e. the confidence of the rule should be greater than or equal to the minimum confidence measure and then the rules that have highest support will get highest priority in all phases. Furthermore, fast building time model is required to enhance the accuracy of classification process (Nababteh et al., 2010; Alnababteh et al., 2014) thus, we proposed a new internal pruning method that enhances the Apriori algorithm speed and accuracy level based on the features of this type of application domain.

Various experimental studies have found that AC techniques perform better than the traditional classifiers owing to the small number of rules that can be produced by the traditional techniques. Moreover, AC techniques produce a large number of important rules that cannot be generated by traditional classifiers, and these rules can enhance the entire classification process (Liu et al., 1998; Ma et al., 2014; Abdelhamid et al., 2014; Abdelhamid et al., 2015; Alazaidah et al., 2015; Taware et al., 2015). Thus, many rule-generation techniques have been proposed with various enhancements, such as the number of rules that are generated during the rule-generation phase, the quality of the rules, and the time taken to generate the rules. Some of these techniques are: Apriori (Agrawal and Srikant, 1994), Direct Hashing and Pruning (DHP) (Park et al., 1995), Fast Distributed Mining of association rules (FDM) (Cheung et al., 1996a), Generalized Sequential Pattern (GSP) (Srikant and Agrawal, 1996), DIC (Brin et al., 1997), Pincer-Search (Lin and Kedem, 1998), CARMA (Hidber, 1999), CHARM (Zaki and Hsiao, 1999), Depth Project (Agrawal et al., 2000), FP-Growth (Han et al., 2000), ECLAT (Zaki, 2000), Diffset (Zaki and Gouda, 2003), PRICE (Wang and Tjortjis, 2004), Scaling Apriori (Prakash and Parvathi, 2010), TopSeqRules (Fournier-Viger and Tseng, 2011), Frequent Pattern Growth Association Rule Mining (FPG-ARM) (Rao and Gupta, 2012), TNR (Fournier-Viger and Tseng, 2012).

In recent years, many researchers have been involved in this research area, focusing on various applications, and this had led to a generation of new demands in relation to association and classification problems, such as creating fuzzy relations between items and classes. To meet these demands, a new generation of single label rules association has been proposed that is called multi-class rules association. Abdelhamid et al. (2014), Abdelhamid (2015), and Alazaidah (2015) proposed Multi-label Classifiers-based Associative Classification (MCAC) to find the relation between any object and all classes, and represented this relation by a ratio. For example, if we have *x* as an object and T/F as two classes, and we need to find the relation between *x* and all classes, by using these algorithms the output will be in the following form: *x*→T, strength of this relation 0.7, and for x→F 0.3. We can observe that the total strength of relations between *x* and the two classes is 1, and this is the main condition in all multi-label rules algorithms. All of these algorithms can be applied on different datasets and give a high level of accuracy in the classification process compared with known AC algorithms.

A new kind of AC algorithm was proposed in (Cheung et al., 1996b; Cheung et al., 1997; Tsai et al., 1999; Petko et al., 2003; Chang et al., 2005; Nababteh et al., 2010; Alnababteh et al., 2014; Hadi, 2015; Hadi et al., 2016) to serve all real-world applications that depend on the incremental learning approach and the need for very fast AC algorithms. However, in the AC problem, researchers have paid little attention to the incremental learning issue. In addition, in view of the fact that classification is a critical task in data mining and has a large number of critical applications that collect data periodically, there is a great interest in enhancing existing classification methods to handle the incremental learning issue.

Three approaches to addressing the association and classification problem have been summarized by Abdelhamid et al. (2015), based on three different motivations, as follows:

- Minimize the number of rules in the candidate set rules to reduce the pruning and classification time. This can be solved by using Immune Systems-Based AC and test data training as two main approaches to addressing the AC problem.
- Many critical applications have a greater need for a high level of accuracy in the classification process than for a faster speed, so calibration AC is suggested as the third approach in this paper. This depends not only on the accuracy of the rules that will be generated in the rule-generation phase, but also on the accurate computation of the membership of the classes, which can lead to increased accuracy of the classification process.
- Minimum support and confidence are two measures in most of AC algorithms. Both of them are given by users, thus give small or large threshold may affect the number and quality of the rules that can be generated, and this will play a very important role in decreasing the accuracy and time of the classification process. For this reason, non-confidence-based learning has been proposed as a fourth approach that aims to use a different technique to solve this problem.

4. Proposed model

We propose an AC algorithm called Fast Classification Based on Association Rule (FCBA). The FCBA aims to optimize the time spent building the model and the classification accuracy rate for the CBA algorithm. Unlike a CBA algorithm that generates rules using the original Apriori algorithm, the FCBA employs a new ranking method that enhances the speed of the Apriori algorithm by adding a new efficient internal pruning mechanism for the rule-generation process.

Our proposed model has three main stages: rule-generation, pruning, and prediction. Firstly, we will show how the pruning stage can affect the rule-generation stage based on a new internal pruning process. After all frequent rules have been generated, the FCBA algorithm ranks these rules according to the following proposed pruning procedure:

1. The rule with higher support measure is given a higher rank.
2. If the support measure values of two or more rules are equal, then the rule with the first occurrence is given a higher rank.
3. The FCBA then prunes the generated rules, like the CBA algorithm, based on the database coverage method (Liu et al., 1998).

Our pruning procedure depends on two assumptions:

Assumption 1: Our algorithm prefers the general rules rather than specific ones to cover most of the dataset entries. In other words, if we have three rules: $A \rightarrow T$, $A, B \rightarrow T$ and $A, B, C \rightarrow T$ that's means $A \rightarrow T$ cover $A, B \rightarrow T$ and $A, B, C \rightarrow T$ so, $A \rightarrow T$ will be selected and the others will be eliminated and no need reach for this level in the rule generation process. We believe that this assumption helps to enhance the efficiency and performance of this algorithm in terms of time and accuracy.

Assumption 2: Our algorithm prefers to differentiate between the support and confidence measures based on type of application to get better results. Thus, in our algorithm the support will get higher priority than the confidence measure to serve the incremental applications (i.e. all generated rules should be confident but in the pruning phase, the rule that has highest support will be placed in the highest rank).

In this pruning method, any single item rule that satisfies the given minimum support and minimum confidence will be added to the CARs, and any rule eliminated that can be generated using this single item because, the support for this rule is greater than or equal to the support of all rules that can be produced in the next step by using this rule. In other words, suppose we have three items, *x*, *y*, and *z*, and two classes, C_1 and C_2. If $x \rightarrow C_1$ is a single frequent item and satisfies the given minimum support and minimum confidence, then the support for this rule will be greater than or equal to the support for $x, y \rightarrow c_1$, $x, z \rightarrow c_1$, and $x, y, z \rightarrow c_1$. With regard to the first occurrence, $x \rightarrow c_1$ as a rule will be added to the CARs and the others will be eliminated. Generating the next frequent items will follow the same procedure.

Depending on this pruning method, the original Apriori algorithm that is used in the rule-generation stage can be enhanced by adding a new intelligent internal pruning process, as shown in Figure 1.

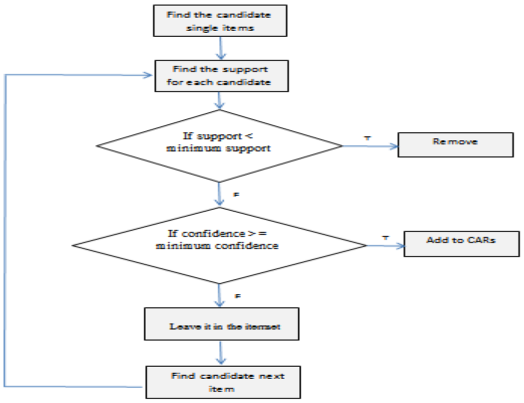
Find the candidate single items
Find the support for each candidate
If support < minimum support
T
Remove
F
If confidence >= minimum confidence
T
Add to CARs
F
Leave it in the itemset
Find candidate next item

Figure 1: Enhanced Apriori algorithm

As is obvious from Figure 1, our proposed optimized Apriori algorithm depends on a new internal Pruning process to generate confident general rules. Our optimized Apriori works based on the following procedure:

```
1.  Input: Dataset T with n training objects
    Minimum support and Minimum confidence
2.  Find the candidate single itemset S.
3.  Find the support for each candidate, the support
    being calculated using equation (4).
4.  For each item:
5.  If the support < minimum support.
6.  Then
7.          Remove from the list
8.  Else
    If confidence>= minimum confidence where the
    confidence of rule is calculated using equation (5)
9.  Then
10.         Add to CARs
11. Else
12.         Leave it in the itemset S.
13. If all items visited
14. Then
15.         Find candidate next itemset $S^-$
16. $S = S^-$
17. If the itemset S is not empty
18. Then
19.         Go to step 2
20. Else
21.         end
```

Finally, the FCBA uses only one rule in the classification process (i.e. the algorithm predicts test instances using the highest support rule that matches the test instance).

To illustrate how the FCBA algorithm works, assume we have the dataset shown in Table 2 with a minimum support = 0.2 and a minimum confidence = 0.5.

Table 2: Dataset sample (T) with five training objects

Training object	Attribute 1(at_1)	Attribute 2(at_2)	Class (C)

1	A1	B1	C2
2	A1	B1	C2
3	A2	B2	C1
4	A1	B1	C1
5	A3	B2	C2

In the first step, our algorithm calculates the support and confidence for each single itemset rule. Then it removes any rule which has support less than the minimum support, any rule has support and confidence greater than or equal the minimum support and confidence will be inserted on CAR list. Otherwise, the rule will be inserted on the candidate list to generate the next item rule, as shown in Table 3.

Table 3: Single itemset rule

Single itemset rule	Support	confidence	status
A1→ C1	0.2	0.33	candidate
A1→C2	0.4	0.67	CAR
A2→C1	0.2	1	CAR
A2→C2	0	0	removed
A3→C1	0	0	removed
A3→C2	0.2	1	CAR
B1→C1	0.2	0.33	candidate
B1→C2	0.4	0.67	CAR
B2→C1	0.2	0.5	CAR
B2→C2	0.2	0.5	CAR

The candidate list contains two single item rules as shown in table 4. In the next step, our algorithm will generates 2-itemset rules by merging the single item rules as shown in table 5.

Table 4: Candidate list

Single itemset rule	Support	confidence	status
A1→ C1	0.2	0.33	candidate
B1→C1	0.2	0.33	candidate

Table 5: 2-itemset rules

Single itemset rule	Support	confidence	status
A1, B1→ C1	0.2	0.33	candidate

The FCBA algorithm generates only one 2-item rule and computes the support and confidence values. This rule has support equal the minimum support but the confidence less than the minimum confidence so, the rule will be candidate. Then, the algorithm stops the generation process because only one rule remained in candidate list and this rule will be removed.

After that, our algorithm sorts all rules in the CAR list according to support and the first occurrence respectively as shown in table 6.

Table 6: CARs list

Order	rules	support
R1	A1→C2	0.4
R2	B1→C2	0.4
R3	A2→C1	0.2
R4	A3→C2	0.2
R5	B2→C1	0.2
R6	B2→C2	0.2

Finally, the m1 method will be employed to select the best rule that cover our dataset as shown in table 7.

Table 7: CARs after pruning

Order	rules	support
1	A1→C2	0.4
2	A2→C1	0.2
3	A3→C2	0.2

This example is evidence that shows the power of our algorithm in reducing the number of unneeded generated rules that leads to enhance the building time model. While, the original Apriori if applied on this example will merge all rules in table 5 that have candidate and CAR statuses, that's means we have 8 rules satisfy these statuses which generate 28 rules in this phase.

5. Experimental results

We performed an extensive analysis to assess the accuracy, precision, recall, F1, and building time for the model. The FCBA algorithm was compared with four well-known AC algorithms – CBA, MCAR, CMAR and FACA – based on a set of experimental results with regard to accuracy, precision, recall, F1, and building time for the model in order to evaluate the scalability and reliability of these algorithms.

We conducted our experiments on a 3GHz i3 PC with a 4GB main memory. Our proposed algorithm is implemented using the Java programming language within the WEKA tool (Hall et al., 2009). The compared algorithms were implemented by the authors. The parameters of these algorithms were: minimum support=0.05 and minimum confidence=0.5.

We used 11 datasets from the UCI repository, mostly from the medical field, to evaluate the performance of all these algorithms and to investigate how the size of the dataset affects the building time for the model, to demonstrate the scalability of these algorithms. The main features of the selected UCI datasets are reported in Table 8.

Table 8: Features of the selected datasets from the UCI repository

Name of Dataset	No. of attributes	No. of instances
Contact-Lenses	5	24
Solar-Flare 1	13	323
Solar-Flare 1	13	1066
Postoperative-Patient	9	90
Shuttle-Landing-control	7	15
Liver-Disorders	7	345
Haberman	4	306
Unbalanced	33	856
Breast-Cancer	10	286
Tae	6	151
Iris	5	150

Table 9 presents a comparison of the average accuracy of CBA, MCAR, CMAR, FACA and FCBA algorithms, with FCBA outperforming the other four algorithms with average accuracy 79.588%. Furthermore, of the 11 datasets, the FCBA achieved the best level of accuracy in 7, which means that the FCBA algorithm was ranked first in 64% of the test datasets. It achieved second place in the remaining 36%, while CBA, MCAR, CMAR and FACA algorithms were ranked first in 55%, 36%, 36% and 54% of the datasets, respectively.

Most of the datasets used in our experimental study were selected from the medical field (Contact-Lenses, Postoperative-Patient, Liver-Disorders, Unbalanced, Breast-Cancer, and Iris). Our proposed algorithm was ranked first in 67% of medical datasets and second in 33%. The CBA and FACA algorithms were ranked first in 50%, and the MCAR and CMAR algorithms in 33%.

The FCBA algorithm outperformed all considered algorithms in term of average of accuracy due to employs the assumptions 1 and 2 that change type of generated rules from specific to general rules and adopts the support measure in rule generation and pruning phases.

Table 9: Evaluation of CBA, MCAR, CMAR, FACA and FCBA algorithms based on accuracy

Datasets	CBA	MCAR	CMAR	FACA	FCBA
Contact-Lenses	66.67	66.67	62.50	68.22	**70.83**
Solar-Flare 1	**97.83**	95.98	95.98	**97.83**	**97.83**
Solar-Flare 2	**99.53**	99.34	99.34	99.34	**99.53**
Postoperative-Patient	58.89	56.67	**71.11**	58.89	**71.11**
Shuttle-Landing-control	**93.33**	**93.33**	**93.33**	**93.33**	**93.33**
Liver-Disorders	56.52	**58.55**	55.94	**57.97**	57.97
Haberman	73.53	73.20	**74.19**	**74.19**	73.53
Unbalanced	**98.60**	**98.60**	**98.60**	**98.60**	**98.60**
Breast-Cancer	**72.38**	68.18	70.629	71.23	**72.38**
Tae	47.02	**50.99**	34.44	47.02	47.02
Iris	**94**	72.67	**94**	**94**	93.33
Average	78.03	75.83	77.28	78.24	**79.59**

Tables 11, 12, and 13 show the comparison between CBA, CMAR, MCAR, FACA and FCBA based on three well-known evaluation measures (F1, precision, and recall), where F1 is calculated based on equation (6):

$$\mathbf{F1} = \frac{\mathbf{2*(Precision*Recall)}}{\mathbf{(Precision+Recall)}} \tag{6}$$

Recall and precision are commonly used for evaluation in machine learning, and are calculated using equations (7) and (8), according to Table 10.

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP+FP}} \tag{7}$$

$$\mathbf{Recall} = \frac{\mathbf{TP}}{\mathbf{TP+FN}} \tag{8}$$

Table 10: Confusion Matrix for Classes

Class	**Predicted as**	
	Actual Class	**Other Classes**
Actual Class	True Positive (TP)	False Negative (FN)
Other Classes	False Positive (FP)	True Negative (TN)

The MCAR algorithm gave a greater level of precision than the other algorithms, producing a smaller number of false positive instances (i.e. if we have two classes, A and B, and we want to compute the precision for class A, the false positive value represents the number of instances that are classified as A, but that are actually not correct). Furthermore, the FCBA algorithm outperformed the others in terms of the recall measure, i.e. the number of correctly classified instances is greater than number of false negative instances, which is equivalent to the accuracy measure. Based on the precision and recall measures, the F1 measure produced the harmonic mean value for our selected algorithms. On this measure, the MCAR algorithm had a higher value.

Table 11: Evaluation of CBA, MCAR, CMAR, FACA and FCBA algorithms based on precision

Datasets	CBA	MCAR	CMAR	FACA	FCBA
Contact-Lenses	0.712	0.820	0.391	0.708	0.708
Solar-Flare 1	0.957	0.962	0.962	0.957	0.957
Solar-Flare 2	0.991	0.991	0.991	0.991	0.991
Postoperative-Patient	0.477	0.539	0.506	0.506	0.506
Shuttle-Landing-control	0.871	0.871	0.871	0.871	0.871
Liver-Disorders	0.507	0.574	0.529	0.529	0.336
Haberman	0.510	0.692	0.708	0.51	0.541
Unbalanced	0.972	0.972	0.972	0.972	0.972
Breast-Cancer	0.696	0.665	0.793	0.665	0.696
Tae	0.348	0.571	0.119	0.348	0.348
Iris	0.940	0.772	0.943	0.934	0.934
Average	0.726	**0.766**	0.708	0.726	0.715

Table 12: Evaluation of CBA, MCAR, CMAR, FACA and FCBA algorithms based on recall

Datasets	CBA	MCAR	CMAR	FACA	FCBA
Contact-Lenses	0.667	0.667	0.625	0.682	0.708
Solar-Flare 1	0.978	0.960	0.960	0.978	0.978
Solar-Flare 2	0.995	0.993	0.994	0.993	0.995
Postoperative-Patient	0.589	0.567	0.711	0.589	0.711
Shuttle-Landing-control	0.933	0.933	0.933	0.933	0.933
Liver-Disorders	0.565	0.586	0.559	0.580	0.580
Haberman	0.735	0.732	0.742	0.742	0.735
Unbalanced	0.986	0.986	0.986	0.986	0.986
Breast-Cancer	0.724	0.682	0.706	0.712	0.724
Tae	0.470	0.510	0.344	0.470	0.470
Iris	0.940	0.727	0.940	0.940	0.933
Average	0.78	0.758	0.773	0.782	**0.795**

Table 13: Evaluation of CBA, MCAR, CMAR, FACA and FCBA algorithms based on F1

Datasets	CBA (F 1)	MCAR (F 1)	CMAR (F 1)	FACA (F 1)	FCBA (F 1)
Contact-Lenses	0.649	0.790	0.481	0.695	0.671
Solar-Flare 1	0.968	0.961	0.961	0.968	0.968
Solar-Flare 2	0.993	0.992	0.992	0.992	0.993
Postoperative-Patient	0.527	0.558	0.591	0.544	0.591
Shuttle-Landing-control	0.901	0.901	0.901	0.901	0.901
Liver-Disorders	0.466	0.573	0.516	0.553	0.425
Haberman	0.623	0.694	0.659	0.604	0.623
Unbalanced	0.979	0.979	0.979	0.979	0.979
Breast-Cancer	0.683	0.673	0.588	0.688	0.683
Tae	0.378	0.509	0.176	0.400	0.378
Iris	0.940	0.730	0.940	0.937	0.933
Average	0.737	**0.760**	0.708	0.751	0.740

To evaluate our selected algorithms with regard to the scalability measure, which reflects how the algorithm can be affected by the increasing size of the dataset (i.e. increasing number of attributes and instances can lead to an increase in the building time of the model for the classifiers). In many applications, such as medical diagnosis, phishing websites, embedded systems and many others, time is very important for making critical decisions within an accepted period. The CBA, MCAR, CMAR, FACA and FCBA algorithms were evaluated based on the building time of the model when executed on small, medium, and large datasets, and the results are shown in Table 14.

Table 14 and Figure 2 show two clusters in relation to the building time of the model and the number of rules that are generated in the classification process. The FCBA, FACA and MCAR algorithms represent one cluster, achieving average times of less than one second and producing small numbers of rules in the rule-generation process, before and after pruning. The CBA and CMAR algorithms are in another cluster, having average times around 400 seconds and generating a huge number of association rules in the rule-generation process, which can affect the time and accuracy. The number and type of rules play the main role in the building time of the model and the accuracy of the classification process, so that if the number of rules is huge, more scanning time and memory will be required. Thus, increasing the number of rules can lead to conflict in the decision-making process that affects the accuracy measure. The FCBA algorithm depends on the generation of rules of a high quality rather than focusing on their quantity. Hence, this algorithm has the smallest number of rules, but these rules are of high quality. In other words, finding a small number of strong rules is better than producing a large number of useless ones that can reduce the accuracy of the classifier. In Table 14, we can observe the gap between the two clusters. However, in the first one the FCBA algorithm outperformed the MCAR, FACA algorithms: the average time of the FCBA algorithm is 0.038s and for the FACA algorithm 0.15s and the MCAR 0.832s, with FCBA also producing fewer rules than MCAR and FACA algorithms, thus requiring less time and memory. Furthermore, the FCBA algorithm won in 8 out of 11 datasets, the MCAR in 7 out of 11 and FACA in 4 out of 11, suggesting that these algorithms should have the same behavior. However, in the large datasets, such as Solar-Flare 1, Solar-Flare 2, and Breast-Cancer, the FCBA algorithm was more scalable than the MCAR and FACA algorithms.

Table 14: Evaluation of CBA, MCAR, CMAR, and FCBA algorithms based on building time for the model

Datasets	CBA	MCAR	CMAR	FACA	FCBA
Contact-Lenses	0.03	**0.00**	0.05	0	**0.00**
Solar-Flare 1	485.33	2.59	483.43	0.07	**0.02**
Solar-Flare 2	746.60	6.28	743.34	1.23	**0.04**
Postoperative-Patient	1.06	**0.04**	0.97	0.06	0.06
Shuttle-Landing-control	0.10	**0.00**	0.01	0.01	**0.00**
Liver-Disorders	0.43	**0.01**	0.39	0.08	0.05
Haberman	0.01	**0.00**	0.01	0	**0.00**
Unbalanced	3069.4	**0.05**	3022.5	0.03	0.09
Breast-Cancer	1.33	0.08	1.08	0.06	**0.06**
Tae	0.10	**0.05**	0.10	0.10	**0.05**
Iris	0.06	**0.05**	0.06	**0**	**0.05**
Average	391.32	0.832	386.54	0.15	**0.038**

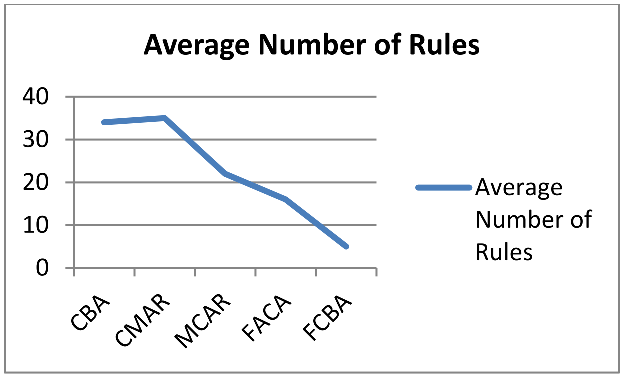

Figure 2: Evaluation of the CBA, MCAR, CMAR, FACA and FCBA algorithms based on the average number of rules in CARs

The FCBA algorithm was faster than the other algorithms and maintained the same performance within all datasets. This means that the FCBA algorithm is more efficient and scalable than the other algorithms, as shown in Figure 3. In the fact, the scalability of this algorithm comes from the new internal pruning method that is proposed for the rule-generation phase and its effect on the external pruning method. The FCBA, CMAR and FACA algorithms performed better than the CBA and CMAR algorithms, and this is clear from Figure 3, when these algorithms were applied to Solar-Flare 1, Solar-Flare 2, and Unbalanced datasets, which are consider to be large datasets. The total average time highlights a large gap between these algorithms, especially between the FCBA and CBA algorithms.

It is obvious; the assumption 1 plays the main role in reducing number of generated rules in the FCBA classifier that leads to enhance the building time model. Specially, the FCBA algorithm employs a new internal pruning method in the original Apriori algorithm that prevents the rule generation process to be continued in each branch contains rule that has support and confidence greater than or equal the inputted ones.

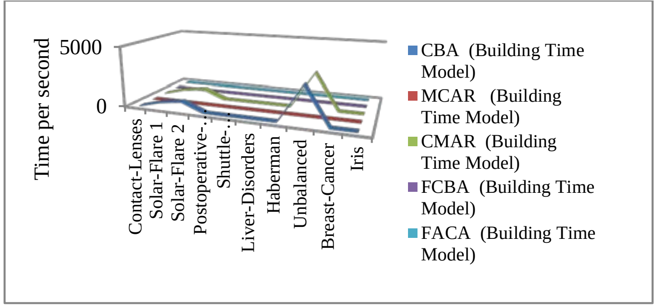

Figure 3: Evaluation of the scalability of the CBA, MCAR, CMAR, FACA and FCBA algorithms based on the building time of the model

6. Conclusion and future works

There are many techniques in the data-mining field that can be used in the decision-making process in many critical areas, such as the medical field, text analysis, website phishing, social media, and many others. One of these techniques is the AC technique that employs the association rules in the classification process to enable more accurate decisions to be taken in many fields. The main challenge faced by this technique is that of achieving a high level of accuracy and speed while at the same time maintaining the efficiency and scalability of the algorithms.

The FCBA algorithm is built on two main features: an extra internal pruning stage for the original Apriori algorithm and new external pruning methods to select more useful association rules. Both of these features contributed to the improved performance of the proposed algorithm in terms of the building time for the model and accuracy measures. It is worth mentioning that the proposed algorithm showed outstanding performance in all datasets, especially medical ones.

This study showed that large changes in the size of the dataset affect the performance of the algorithms significantly. Our proposed algorithm maintained the same performance on all experiments with respect to accuracy and speed. Moreover, the number and type of rules can clearly enhance the speed and accuracy of the classification process.

In future work, we will investigate different pruning and prediction methods and show their impact on the decision-making process in different fields in terms of a set of well-known evaluation measures, such as accuracy, F1, recall, precision, and building time of the model.

References

[1] Abdelhamid, N., Ayesh, A., and Hadi, W. (2014), Multi-Label Rules Algorithm Based Associative Classification. Parallel Processing Letters, 24(01), 1450001-14500021.

[2] Abdelhamid, N., Ayesh, A., and Thabtah, F. (2015), Emerging Trends in Associative Classification Data Mining. International Journal of Electronics and Electrical Engineering, 3(1), 50-53.

[3] Agrawal, R., Aggarwal, C., and Prasad, V. (2000), Depth first generation of long patterns. In Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August, 2000, 108-118. ACM.

[4] Agrawal, R., and Srikant, R. (1994), Fast algorithms for mining association rules. In Proc. 20th Int. Conf. Very Large Data Bases, September, 1994, 487-499.

[5] Alazaidah, R., Thabtah, F., and 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.

[6] Alnababteh, M., Alfyoumi, A., Aljumah, A., and Ababneh, J. (2014). Associative Classification Based on Incremental Mining (ACIM). International Journal of Computer Theory and Engineering, 6(2).

[7] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. (1997), Dynamic itemset counting and implication rules for market basket data. In ACM SIGMOD Record, June, 1997, 255-264. ACM.

[8] Chang, C., Li, Y., and Lee, J. (2005). An Efficient Algorithm for Incremental Mining of Association Rules. In Proc. 15th International Workshop on Research Issues in Data Engineering: Stream Data Mining and Applications, Washington, 2005, 3-10.

[9] Chapman, M., Callis, P., and Jackson, W. (2004), Metrics data program. NASA IV and V Facility, http://mdp.ivv.nasa.gov.

[10] Cheung, D., Han, J., Ng, T., Fu, W., and Fu, Y. (1996a), A fast distributed algorithm for mining association rules. In Parallel and Distributed Information Systems, Fourth International Conference ,December, 1996, 31-42. IEEE.

[11] Cheung, D., Han, J., Ng, V., and Wong, C. (1996b). Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique. In Proc. 12th International Conference on Data Engineering, New Orleans, 1996, 106-114.

[12] Cheung, D., Lee, S., and Kao, B. (1997). A General Incremental Technique for Mining Discovered Association Rules. In Proc. 5th International Conference on Database System for Advanced Applications, Melbourne, 1997, 185-194.

[13] Fournier-Viger, P., and Tseng, V. S. (2011), Mining top-k sequential rules. In Proc. 7th Intern. Conf. on Advanced Data Mining and Applications (ADMA 2011), Beijing, China, December, 2011, 180-194. Springer, Berlin, Heidelberg.

[14] Fournier-Viger, P., and Tseng, V. S. (2012), Mining top-k non-redundant association rules. In Foundations of Intelligent Systems, 20th International Symposium, ISMIS,December, 2012, 31-40. Springer, Berlin, Heidelberg.

[15] Gupta, A., Kumar, N., and Bhatnagar, V. (2005). Incremental classification rules based on association rules using formal concept analysis. In Machine Learning and Data Mining in Pattern Recognition, July, 2005, 11-20. Springer, Berlin, Heidelberg.

[16] W. Hadi, ECAR: a new enhanced class association rule, Adv. Comput. Sci.Technol. 8 (1) (2015) 43–52.

[17] Hadi, W. (2013). EMCAR: Expert Multi Class Based on Association Rule. International Journal of Modern Education and Computer Science, 5(3), 33-41.

[18] Hadi, W. E., Aburub, F., &Alhawari, S. (2016). A new fast associative classification algorithm for detecting phishing websites. Applied Soft Computing, 48, 729-734.

[19] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1), 10-18.

[20] Han, J., Pei, J., and Yin, Y. (2000, May). Mining frequent patterns without candidate generation. In ACM SIGMOD Record,May, 2000, 1-12. ACM.

[21] Hidber, C. (1999), Online association rule mining. In ACM SIGMOD International Conference on Management of Data,June, 1999, 145-156. ACM.

[22] Li, W., Han, J., and 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.

[23] Lin, D. I., and Kedem, Z. M. (1998). Pincer-search: A new algorithm for discovering the maximum frequent set. In Advances in Database Technology—EDBT'98, 6th International Conference on Extending Database Technology, March, 1998, 103-119. Springer, Berlin, Heidelberg.

[24] Liu, B., Hsu, W., and Ma, Y. (1998). Integrating classification and association rule mining. In Proceedings 4th International Conference on Knowledge Discovery and Data Mining, August, 1998, 80-86. New York, NY.

[25] Ma, B., Zhang, H., Chen, G., Zhao, Y., and 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.

[26] Nababteh, M., Al-Shalabi, R., Thabtah, F., and Najeeb, M. (2010). An Incremental Data Insertion Algorithm for Associative Classification Mining. In Proc. 15th International Business Information Management Conference on Knowledge Management and Innovation: A Business Competitive Edge Perspective, Cairo, 2010, 1806-1812.

[27] Park, J. S., Chen, M. S., and Yu, P. S. (1995). An effective hash-based algorithm for mining association rules. In Proc. ACM SIGMOD International Conference on Management of Data, May, 1995, 175-186. ACM.

[28] Petko, V., Rokia, M., Mohamed, R., and Robert, G. (2003). Incremental Maintenance of Association Rule Bases. In Proc. 2nd Intl. Workshop on Data Mining and Discrete Mathematics, San Francisco, 2003, 12 p.

[29] Prakash, S., and Parvathi, R. M. S. (2010). An enhanced scaling Apriori for association rule mining efficiency. European Journal of Scientific Research, 39(2), 257-264.

[30] Rao, S., and Gupta, P. (2012). Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm. IJCST, 3(1), 489-493

[31] Shirabad, J. S., and Menzies, T. J. (2005). The PROMISE repository of software engineering databases. School of Information Technology and Engineering, University of Ottawa, Canada, 24.

[32] Srikant, R., and Agrawal, R. (1996). Mining sequential patterns: Generalizations and performance improvements. In Proceedings of the 5th International Conference on Extending Database Technology, June, 1996, 1-17. Springer, Berlin, Heidelberg.

[33] Tan, P. N., Steinbach, M., and Kumar, V. (2006). Introduction to data mining (Vol. 1). Pearson Addison Wesley, Boston.

[34] Taware, S., Ghorpade, C., Shah, P., Lonkar, N., and Bk, M. (2015). Phish Detect: Detection of Phishing Websites based on Associative Classification (AC).International Journal of Advanced Research in Computer Science Engineering and Information Technology, 4(3), 384-395.

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

[36] Thabtah, F., Cowling, P., and Hammoud, S. (2006). Improving rule sorting, predictive accuracy and training time in associative classification. Expert Systems with Applications, 31(2), 414-426.

[37] Tsai, P., Lee, C., and Chen, A. (1999). An efficient approach for incremental association rule mining. In Proc. 3rd Pacific-Asia Conference on Methodologies for Knowledge Discovery and Data Mining, London, 1999, 74-83.

[38] Wang, C., and Tjortjis, C. (2004). PRICES: an efficient algorithm for mining association rules. In Intelligent Data Engineering and Automated Learning–IDEAL, 2004,352-358. Springer, Berlin, Heidelberg.

[39] Yin, X., and Han, J. (2003). CPAR: Classification based on Predictive Association Rules. In SDM, May, 2003, 331-335.

[40] Zaki, J. (2000). Scalable algorithms for association mining. IEEE Transactions on Knowledge and Data Engineering, 12(3), 372-390.

[41] Zaki, J., & Gouda, K. (2003, August). Fast vertical mining using diffsets. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 326-335). ACM.

[42] Zaki, J., and Hsiao, C. J. (1999). CHARM: An efficient algorithm for closed association rule mining (Vol. 10). Technical Report 99.

Jaber Alwidian is a PhD Candidate in the Department of Computer Science at the University of Jordan. He received his B.Sc. degree in Computer Information System from the University of Philadelphia and M.Sc. degree in Information System from the Jordan University in 2005 and 2010, respectively. He has about seven years of work experience as a lecturer. His research interests are data mining, software engineering and image processing.

Bassam Hammo (Professor) is with The University of Jordan since 2003. He is an active researcher in the field of Arabic Natural Language Processing (ANLP). His research interests include: Arabic historical corpora, morphological analyzers and parsers, data mining and ontologies. He has long been a supporter of free software tools and resources for Arabic language.

Nadim Obeid holds a B.Sc. in Mathematics (Lebanese University, 1979) and a B.Sc. in Business Administration (Lebanese University, 1980). He also holds A Postgraduate Diploma (Essex University, 1982), M.Sc. in Computer Studies (Essex University, 1983) and a Ph.D. in Computer Science (Essex University, 1987). In 1986, he joined the EUROTRA project as a Senior Research Officer and then in 1987, he took the post of a Lecturer in the department of Computer Science at Essex University. He joined Princess Sumaya University for Technology in 1996 and became an associate Professor in 1998. He was promoted in 2002 to the position of professor. In 2004, he joined, as a professor, King Abdullah II School for Information Technology at the University of Jordan. He served as the Deputy Dean of King Abdullah II School for information technology during 2008-2009 and dean during 2010-2012. The areas of research in which he is currently active are: Knowledge Representation, Multi-Agent Systems, Dialogue and Argumentation Systems, Formalisation of Access Control Policies and Data Mining.