ASSOCIATIVE CLASSIFICATION IN MULTI-LABEL CLASSIFICATION: AN INVESTIGATIVE STUDY

Raed Alazaidah[1], Mohammed Amin Almaiah[2] and Mo'ath Al-Luwaici[3]

(Received: 9-Mar.-2021, Revised: 24-Apr.-2021, Accepted: 5-May-2021)

ABSTRACT

Multi-label classification (MLC) is a very interesting and important domain that has attracted many researchers in the last two decades. Several single-label classification algorithms that belong to different learning strategies have been adapted to handle the problem of MLC. Surprisingly, no Associative Classification (AC) algorithm has been adapted to handle the MLC problem, where AC algorithms have shown a high predictive performance compared with other learning strategies in single-label classification. In this paper, a deep investigation regarding utilizing AC in MLC is presented. An evaluation of several AC algorithms on three multi-label datasets with respect to five discretization techniques revealed that utilizing AC algorithms in MLC is very promising compared with other algorithms from different learning strategies.

KEYWORDS

Prediction, Machine learning, Multi-label classification, Associative classification, Learning strategies.

1. INTRODUCTION

Classification is a very interesting task in data mining that involves assigning the class label of an unseen instance as accurately as possible, based on a labeled historical training set [1]-[2].

In general, classification could be divided into three main types the first type of which is called binary classification and comprises only two class labels. The second type is called multi-class classification and comprises more than two class labels in a dataset. Both binary classification and multi-class classification have been known as a conventional single label classification [3]. In single-label classification, class labels are considered to be mutually exclusive; that is, each instance in the dataset is associated with only one class label [4]. The third type is called MLC. MLC does not assume labels in the dataset to be mutually exclusive and hence, an instance in a multi-label dataset could be associated with more than one class label at the same time [5].

MLC has several distinguishable features over single-label classification. First, class labels in MLC are not considered to be mutually exclusive as in single-label classification and hence, class labels in MLC do have some kind of correlations and dependencies [6]. Second, the problem search space of a single-label classification problem is quite limited when compared with the large problem search space of the MLC problem [7]. The problem search space of the MLC problem equals n^q, where q represents the total number of the class labels in the dataset. On the other side, the problem search space of binary classification equals 2 and for multi-class classification equals q. Finally, the complexity of MLC is very high compared with the complexity of single label classification [8]-[9].

Two main approaches are being used to handle MLC. The first approach adapts a single-label classification algorithm to handle a multi-label dataset, while the second approach transforms the multi-label dataset into one single-label dataset or more and then, this approach applies one single-label classifier or more on the transformed datasets, where the outputs of the single-label classifiers on the transformed datasets are aggregated to form the final prediction [10]. Regardless of the approach being used to handle multi-label datasets, the choosing step of the single-label classification algorithm (base classifier) is crucial in determining the accuracy of the proposed MLC algorithm [11].

Many base classifiers have been utilized in MLC, whether by adapting the single-label base classifier to handle multi-label datasets or by applying them to the transformed versions of the multi-label dataset.

1. R. Alazaidah is with the AI Department, Faculty of Science and IT, Irbid National University. Email: raeddiab@yahoo.com
2. M. A. Almaiah is with the Department of Computer Networks and Communications, College of Computer Sciences and Information Technology, King Faisal University, Al-Ahsa, Saudi Arabia.
3. M. Al-Luwaici is with the Institute of Engineering Mathematics, Uni. Malaysia Perlis (UniMAP), Malaysia. Email: wese5@yahoo.com

These base classifiers follow several learning approaches, such as decision trees, neural networks, fuzzy-based learning, lazy learning, statistical learning, support machine learning and several other learning approaches.

Surprisingly, the AC approach, which has been proven to produce high accurate rules and has the ability of discovering hidden knowledge that could not be discovered by other learning strategies [12]-[13], has been weakly utilized in MLC. According to [14], no AC algorithm has the ability of generating multi-label rules and hence, no AC algorithm can handle the problem of MLC. Nevertheless, few research studies that attempted to handle the problem of MLC could be found in the literature. Unfortunately, most of these attempts could not be recognized as effective MLC algorithms, as explained in Section 2, part C.

Therefore, this paper is interested in investigating the applicability of the AC learning approach in solving the problem of MLC, either by adapting one of the AC algorithms to handle MLC problems or by utilizing AC algorithms in classifying the transformed versions of the multi-label datasets and then, aggregating the outputs of these classifiers to generate multi-label rules.

Specifically, this paper aims to meet two main objectives. The first is to evaluate several AC algorithms on three multi-label datasets, with respect to five discretization techniques. The evaluation procedure considers two criteria: accuracy metric and running time. The second objective is to compare the performance of the most promising AC algorithms with other algorithms from several learning strategies, based on the accuracy metric, to determine the applicability of AC in solving the problem of MLC.

The rest of the paper is organized as follows: Section 2 presents some related work, while Section 3 introduces the empirical analysis. Section 4 presents the conclusion and lists some possible future work.

2. LITERATURE REVIEW

In this section, a brief-yet comprehensive-overview of the MLC domain is presented in sub-section A. Also, a quick review of AC learning approach and the considered AC algorithms is introduced in sub-section B. Finally, attempts to utilize AC in MLC are presented in sub-section C.

A. Overview of MLC

MLC is a general type of classification that allows the examples (instances) in a dataset to be associated with more than one class label at the same time [15], [6]. Hence, the goal in MLC is to learn a function from a set of instances, where each instance could be associated with one or more class labels [16].

MLC was motivated at first by text categorization and medical diagnosis [17]. Recently, more scholars have paid great attention toward the problem of MLC; due to its importance in real-world applications [18]. In many domains, where single-label classification failed to solve the classification problem, MLC did [19]. For example, single-label classification may tag an email message as either a *work* or a *research project* but not both, whereas the fact is, the message could be tagged as both *work* and *research project* simultaneously, which MLC does. Nowadays, MLC is increasingly required by modern applications, such as music categorization into emotions [20], semantic video annotation [21], direct marketing [22] and protein function classification [23].

Two main general approaches are being used to handle MLC problems. The first approach is called the Problem Transformation Method (PTM), while the second approach is called the Algorithm Adaptation Method (AAM) [24]. The former transforms a multi-label dataset into a single-label dataset by using different transformation methods, such as Least Frequent Label (LFL), Most Frequent Label (MFL) or by choosing any label randomly [24]. Then, any of the shelf single-label classifiers could be used to classify the transformed dataset [25]. The latter adapts a specific single-label classification algorithm to handle a multi-label dataset [26]. Using PTMs is preferable over using AAMs; because the former are simpler, more general and not domain-specific like AAMs [27].

The step of choosing the base classifier is vital in both PTMs and AAMs. In fact, in datasets with low cardinality, such as Scene, Genbase and Emotions, the accuracy of the base classifier highly affects the final accuracy of the multi-label prediction step. Also, the accuracy of the base classifier in the high-cardinality dataset, like Yeast and Emotions, highly affects the classification step by determining the prediction step of the other labels that have been discarded due to the transformation step.

Therefore, several single-label classification algorithms have been utilized in the domain of MLC as base classifiers, such as C4.5, KNN, PART and several other single-label classification algorithms. Surprisingly, very few research studies have utilized AC in MLC [14], [28]-[29]. The next sub-section briefly overviews the AC learning approach.

B. Associative Classification

Associative Classification (AC) is a learning approach that integrates the task of mining association rules with the task of classification [30]. Recently, AC has attracted many researchers for two main reasons. First, AC is capable of producing higher accurate rules than other learning approaches. Second, AC generates rules that are easier to be understood by the different types of users [31]. Thus, several classification algorithms have been proposed under the AC approach of learning. Even though these AC-based algorithms have shown high predictive performance in conventional single-label classification problems, unfortunately, they have never been adapted to handle a MLC problem [14].

In general, any AC algorithm comprises three phases. In the first phase, the algorithm searches the training data for any associations between the attributes' values and the class labels. The discovered associations are generated as Class Association Rules (CARs) in an "IF-THEN" format [32]-[33]. After generating the complete set of CARs, pruning and ranking procedures are used to prune weak rules according to some specific thresholds, such as *Support* and *Confidence* and rank the remaining strong rules according to their *Support*, *Confidence* and the number of the conditions in the antecedent of the rule or any other ranking criteria (Phase 2). The final output of the second phase is the classifier, which comprises a set of CARs. Lastly, the classifier is tested against a new and independent dataset to verify its effectiveness in predicting new unseen instances [34]. Figure 1 shows the main general phases for any AC algorithm and Table 1 shows some main concepts and definitions related to the AC.

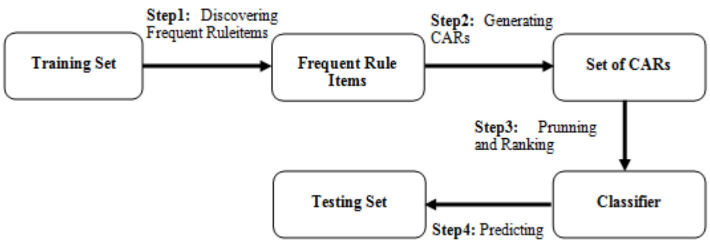

Figure 1. General steps for AC algorithms.

Table 1. Main definitions and concepts related to AC learning approach.

Concept	Definition
Item	An association between an attribute in the dataset and its value (A_i,a_i) or a combination of several attributes' values (A_1,a_1), (A_7,a_7), (A_9,a_9).
Rule	An "IF-Then" rule that has a combination of items in the antecedent and one class label only in the consequent.
Actual Occurrence (AccOccur)	Number of cases in the training dataset that matches the antecedent of a rule.
Support Count (SuppCount)	Number of cases in the training dataset that match the antecedent of a rule and belong to a specific class label.
Minimum Support (MinSupp)	A user predefined threshold. A rule r passes the minsup threshold if SuppCount(r)/n >= MinSupp, where n: number of instances in the training set.
Minimum Confidence (MinConf)	A user predefined threshold. A rule r passes the MinConf threshold if SuppCount(r)/AccOccur(r) >= MinConf.
Frequent Item	An item in the training dataset that passes the MinSupp threshold.

In general, AC-based algorithms start with discovering frequent items that comprise only a single value; i.e., items <A1,a_1>, <A2,a_2> and <A2,x_1>. Any item that passes the user predefined MinSupp threshold is said to be a frequent single item.

For example, in Table 2, if the MinSupp equals 0.4, then the SuppCount will be 4, because there are 10 instances (cases) in the dataset. Therefore, the following are the single frequent items: <A1,a_1>, <A1,a_2>, <A2,x_1> and <A2,x_2>. After that, based on the discovered single frequent items, a new pass over the dataset is carried out to discover frequent triples of items and so on. Thus, most AC algorithms perform several passes over the training set to generate the frequent items that satisfy the user predefined MinSupp threshold.

The next step is to generate the complete set of CARs that satisfy the MinConf threshold based on the discovered frequent items. For example, the following rule could be generated from Table 2, considering that MinConf=0.8: <A1,a1> and <A2,x1> ➔ C1.

Finally, after generating all CARs, a ranking and pruning step is applied on the discovered CARs to keep the most accurate CARs and remove the others.

Table 2. Transactional training dataset.

Row ID	A1	A2	Class
1	a1	x1	C1
2	a1	x2	C1
3	a1	x1	C1
4	a2	x2	C3
5	a2	x1	C4
6	a2	x2	C2
7	a2	x2	C2
8	a1	x1	C1
9	a2	x2	C2
10	a1	x1	C1

Several research studies have shown that AC has two distinguishable features over other methods and approaches of classification [14], [35]. The first one is its simplicity in representing the knowledge in the form of "IF-THEN" rules and its high interpretability. The second distinguishable feature is its great ability to find hidden and additional information, which leads to minimizing the error rate of the classifier and hence highly improving the classification step.

Classification Based on Associations (CBA), which is one of the first algorithms that combined the tasks of Association Rule Mining (ARM) and Classification, was proposed in [36]. Since then, many other algorithms have been proposed based on the concept of merging ARM with classification. CBA managed to utilize the *Apriori* algorithm [37] in a classification dataset through applying three main steps. In the first step, any continuous attribute (if any) in the dataset is discretized. The second step of CBA involves generating all CARs. CARs consider only those rules that have any combination of items in the left-hand side (antecedent) and only one of the classes in the right-hand side (consequent). CARs are chosen based on user-defined thresholds called *Support* and *Confidence*, in which the value of the *Support* threshold is usually very low and the value of the *Confidence* value is high. The third step aims to build a single-label classifier based on the previously discovered CARs [30].

CBA was improved later in [38] by eliminating two weaknesses related to the original CBA. The first weakness is using one value for the *minsup* threshold, which might cause imbalanced class distribution. This weakness has been tackled in the adapted version through using multiple *minsup* thresholds. The second weakness of the original CBA is the exponential growth of the number of rules generated by CBA. This weakness has been tackled by merging CBA with a decision tree as in C4.5, which has led to more accurate rules. The adapted version of CBA has been called CBA2 or msCBA, short for multiple support CBA.

Although CBA2 has shown excellent performance in single-label classification when compared with other algorithms that follow other learning strategies [30], unfortunately, CBA2 does not have the capability to handle multi- label datasets. CBA2 assumes that only one class label is associated with an input instance. Thus, it produces single-label rules with one class label as a consequence of the rule. Hence, to adapt the CBA2 algorithm to handle multi-label datasets, this assumption should be avoided. Also, the CBA2 algorithm captures the associations between features (attributes) and class labels globally, where local associations and dependencies are proven to have a better performance than global associations and dependencies [39].

Yin and Han (2003) [40] proposed an AC algorithm that has been called CPAR, short for Classification based on Predictive Association Rules. CPAR guarantees the generation of more rules, because the training set is allowed to be covered by several rules instead of one single rule, which leads to an improvement in the classification accuracy. CPAR managed to do that by enhancing the First-Order Inductive Learner (FOIL) and considering all the positive cases associated with the generated rule instead of discarding them as in other AC algorithms. Also, CPAR can generate simultaneous multiple rules at the same time by considering the value of all attributes with the largest FOIL-gain instead of considering only one attribute value as in FOIL.

The classification based on Multiple Association Rules (CMAR) algorithm is another AC algorithm that was proposed in [41]. CMAR was the first AC-based algorithm that utilized the FP-growth technique to capture the hidden associations among the features and the class labels. CMAR used a prefix tree data structure called C-tree to save the learned rules. An extensive evaluation based on 26 UCI datasets revealed that CMAR has a competitive performance compared with the CBA and C4.5 algorithms.

In [42], a new AC-based algorithm was presented. The algorithm was called FCRA, short for finding Fuzzy Classification Rules based on the *Apriori* algorithm. FCRA proposed a new data mining technique that captures fuzzy classification rules based on the *Apriori* algorithm. FCRA utilizes a genetic algorithm to automatically determine the threshold of the minimum fuzzy support. An evaluation of the FCRA algorithm on Iris dataset revealed its superior performance compared with other classification algorithms.

A fuzzy-based AC algorithm that enhances the understandability of the generated classifier by reducing the total number of the classification rules was presented in [43]. Classification with Fuzzy Association Rules (CFAR) utilizes the concept of fuzzy logic in solving the so-called "sharp boundary" problem in ARM techniques with quantitative attributes' domains. CFAR has been compared against CBA and showed a better performance in terms of understandability represented by the total number of the generated classification rules.

C. Utilizing AC in MLC

One of the most popular algorithms that utilizes AC to handle the problem of MLC is the Multiclass Multilabel Associative Classification (MMAC) algorithm [29]. MMAC comprises three steps. First, it transforms the multi-label dataset into a single-label dataset, using *copy* as a problem transformation method. Second, it trains a single-label associative classifier to predict a single-label using "IF-THEN" rules. Finally, it merges the predictions of rules that have the same antecedent to form a rule with more than one label in the consequence of the rule. It is worth mentioning that all the datasets that have been used to evaluate MMAC are single label datasets and MMAC has never been tested against multi-label datasets. Also, MMAC assumes class labels to be mutually exclusive and ignores any dependencies among labels, which makes it unsuitable for large datasets with high number of instances and labels.

In [44], a new multi-label algorithm based on AC was introduced and dubbed the Multi-label Classifier based on Associative Classification (MCAC). The algorithm uses a novel rule discovery method that generates and discovers multi-label rules from a single label dataset, without performing the learning step in the dataset. These multi-label rules represent vital information that is usually ignored by most previous AC algorithms. As in MMAC, this algorithm has been tested against single-label datasets and never considered the dependencies among labels.

In [45], a Correlatives Lazy Associative Classifier (CLAC) algorithm was introduced. CLAC is based on two approaches of classification: lazy learning that delays the reasoning process until a new test instance is given and associative classification that merges the association rule mining task with the classification task. In CLAC, the CARs do not have more than one label and consequently, CLAC assigns a value to each CAR based on its *Support* and *Confidence* and the associated class label. Then, CLAC adds the predicted class of the test instance to the instance as a new feature and uses the new test instance with the added feature (class label) to predict a new class label and so on until no further class label can be found. CLAC was evaluated against three textual datasets and achieved better performance compared with the BoosTexter algorithm [46].

In [47], another algorithm that followed the approach of AC was presented. The algorithm produced multi-label association rules by considering all the labels with a probability greater than or equal to (0.5). In fact, their algorithm is similar to MMAC in all of its steps with only one difference in the evaluation of the algorithm. MMAC has been evaluated using single-label datasets, while this algorithm has been evaluated using only one multi-label dataset (Scene). The authors concluded that using AC with MLC will lead to good performance, but generalizing this conclusion is difficult *via* an experiment on only one dataset with specific features and characteristics.

3. EMPIRICAL ANALYSIS

In this section, a comprehensive description of the research conducted is presented. At first, a description of the considered multi-label datasets and the settings of the six AC classifiers considered is introduced. Then, an evaluation of the results of the six AC classifiers is presented. Finally, a discussion regarding the evaluation results is provided.

The accuracy of the classification task in the domain of MLC is still low when compared to other types of classification, like the binary classification and the multi-class classification. Therefore, the main evaluation metric in this investigation study is the accuracy metric. Also, since all AC algorithms highly depend on the discretization technique being used, this paper studies and attempts to identify the most appropriate discretization technique which leads to the best accuracy results. Finally, the running time for the considered algorithms is considered to get the complete picture regarding the significance of utilizing the approach of AC in handling the problem of MLC.

A. Settings and Datasets

Three multi-label datasets with different characteristics are considered in this paper. Each dataset has been transformed into a single-label dataset based on a novel transformation method called High Standard Deviation First (HSDF) [11], in which the label space of the multi-label dataset is extracted first. Then, for each label (item) in the extracted label space, the Predictive Apriori [48] algorithm is applied to capture all positive pairwise associations in the form (IF X=1, THEN Y=1). After that, the standard deviation of the accuracies of the captured positive associations is computed for each label. Finally, the labels are ordered based on the computed standard deviation in a descending order and the input dataset is transformed into a single-label dataset based on the order determined earlier.

Table 3 shows the main characteristics of the considered multi-label datasets in this paper. All datasets are available in Mulan, a multi-label dataset repository [49]. Datasets could be downloaded from http://mulan.sourceforge.net/datasets-mlc.html. Finally, it is worth mentioning that the training datasets and the testing datasets have been chosen according to the datasets author recommendation, where 2/3 of the dataset has been used as a training set and 1/3 of the dataset has been used as a testing set.

Table 3. Datasets characteristics.

Dataset	Instances	Attributes	Labels	LCard	Domain
Scene	2712	294	6	1.074	Image
Emotions	593	72	6	1.868	Media
Flags	194	19	7	3.392	Image

Six AC-based classifiers have been considered in this paper. These classifiers are: CBA, CBA2, CMAR, CPAR, FCRA and CFAR. These classifiers have been used with their default settings as they have been implemented in KEEL [50]. KEEL is an open source Java software that can be used in a wide range of data mining tasks. KEEL is short for Knowledge Extraction based on Evolutionary Learning.

Each classifier has been trained on each transformed version of the considered dataset five times and each time a different discretization technique is used. Five discretization techniques are considered in this paper: Chi2-D [51], Bayesian-D [52], Ameva-D [53], 1R-D [54] and E-Chi2-D [55].

B. Evaluation of Several AC-based Classifiers and other Classifiers from Different Learning Approaches

Table 4 to Table 6 show the results of the evaluation of the six AC classifiers on the three considered multi-label datasets using accuracy metric. Accuracy measures the percentage of those labels that were correctly predicted, with respect to the total number of labels and averaged over all instances. Accuracy is computed using the following equation:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \tag{1}$$

where TP=number of the true positive predictions, TN=number of the true negative predictions, FP=number of the false positive predictions and FN=number of the false negative predictions.

Table 4 shows the accuracy rates of the six different AC-based classifiers on the *Scene* dataset, with respect to 5 discretization techniques. The *Scene* dataset comprises 2712 instances and 294 attributes.

Table 4. Accuracy rates of the six AC algorithms on the *Scene* dataset.

Algorithm	**Chi2-D**	**Bayesian-D**	**Ameva-D**	**1R-D**	**E-Chi2-D**	**Average**
CBA2	0.840	**0.991**	0.801	0.810	0.643	**0.817**
CBA	0.774	0.817	0.830	0.753	0.633	0.761
CMAR	0.776	0.669	0.758	0.712	0.615	0.706
CPAR	0.730	0.753	0.742	0.669	0.607	0.700
FCRA	0.559	0.591	0.562	0.523	0.541	0.555
CFAR	0.273	0.136	0.270	0.145	0.027	0.170
Averag	0.659	0.660	**0.661**	0.602	0.511	

According to Table 4, CBA2 has the best accuracy average on the Scene dataset. Considering the discretization techniques, it can be clearly noted from the table that Ameva-D is the best discretization technique among the 5 considered techniques. Nevertheless, Chi2-D and Bayesian-D show nearly equivalent results to Ameva-D. Finally, the highest accuracy was observed with the CBA2 algorithm when using Bayesian-D as a discretization technique.

Table 5 shows the accuracy rates for the considered AC classifiers on the *Emotions* dataset, with respect to 5 discretization techniques. The *Emotions* dataset comprises 593 instances and 72 attributes.

Table 5. Accuracy rates of the six AC algorithms on the *Emotions* dataset.

Algorithm	**Chi2-D**	**Bayesian-D**	**Ameva-D**	**1R-D**	**E-Chi2-D**	**Average**
CBA2	**0.966**	0.877	0.815	0.455	0.953	**0.813**
CBA	0.598	0.526	0.613	0.447	0.624	0.562
CMAR	0.529	0.396	0.529	0.258	0.526	0.448
CPAR	0.603	0.560	0.562	0.429	0.598	0.550
FCRA	0.294	0.388	0.452	0.416	0.342	0.378
CFAR	0.209	0.209	0.209	0.209	0.209	0.209
Average	0.533	0.493	0.530	0.369	**0.542**	

Table 5 clearly shows that CBA2 has the highest accuracy among the six AC classifiers on the *Emotions* dataset, especially when using Chi2-D discretization technique.

For the discretization techniques, E-Chi2-D shows the best results, with a competitive performance from the Chi2-D and Ameva-D techniques. The best accuracy has been observed with the CBA2 algorithm when using Chi2-D as a discretization technique.

Table 6 shows the accuracy rates of the six different AC-based classifiers on the *Flags* dataset, with respect to 5 discretization techniques. The *Flags* dataset comprises 194 instances and 19 attributes.

Table 6. Accuracy rates of the six AC algorithms on the *Flags* dataset.

Algorithm	**Chi2-D**	**Bayesian-D**	**Ameva-D**	**1R-D**	**E-Chi2-D**	**Average**
CBA2	**0.912**	0.855	0.865	0.835	0.855	**0.864**
CBA	0.798	0.752	0.768	0.737	0.752	0.761
CMAR	0.768	0.721	0.747	0.680	0.721	0.727
CPAR	0.608	0.572	0.592	0.603	0.572	0.589
FCRA	0.510	0.510	0.510	0.510	0.510	0.510
CFAR	0.185	0.185	0.185	0.185	0.185	0.185
Average	**0.630**	0.599	0.611	0.592	0.599	

Table 6 clearly shows that CBA2 has the best accuracy on the *Flags* dataset, especially when using Chi2-D as a discretization technique. Considering the discretization techniques, Chi2-D has the best results, with a competitive performance from the Ameva-D technique. The best accuracy has been observed with CBA2 algorithm when using Chi2-D as a discretization technique.

Based on the accuracy results for the six AC-based classifiers on the three multi-label datasets, the conclusion can be made that CBA2 algorithm is the best AC algorithm in handling multi-label datasets. Table 7 shows the running time in seconds (time needed to build the classifier) for the six AC classifiers on the three datasets, with respect to the discretization technique being used.

Table 7. Running time for several AC classifiers on the considered datasets.

Dataset	**Algorithm**	**Chi2-D**	**Bayesian-D**	**Ameva-D**	**1R-D**	**E-Chi2-D**	**Average**
Scene	**CBA2**	5	6	3	8	4	5.2
	CBA	7	105	21	15	13	32.2
	CMAR	773	2488	1895	3722	691	1913.8
	CPAR	1	1	1	1	0	0.8
	FCRA	204	106	212	7263	5620	2681
	CFAR	11	1	36	100	1508	331.2
	Average	**166.8**	451.2	361.3	1851.5	1306.0	
Emotions	**CBA2**	0	0	0	0	1	0.2
	CBA	0	0	0	0	0	0
	CMAR	0	0	0	0	0	0
	CPAR	0	0	0	0	1	0.2
	FCRA	12	54	13	81	12	34.4
	CFAR	0	0	0	1	0	0.2
	Average	**2.0**	9.0	2.2	13.7	2.3	
Flags	**CBA2**	47	6	42	10	13	23.6
	CBA	3	1	3	2	2	2.2
	CMAR	1	1	2	2	1	1.4
	CPAR	0	0	0	0	0	0
	FCRA	42	62	32	99	71	61.2
	CFAR	2	2	2	5	5	3.2
	Average	15.8	**12.0**	13.5	19.7	15.3	

In general, CPAR shows the best running times on the three datasets considering the five discretization techniques. Among the discretization techniques, Chi2-D has the best running time on the *Scene* and *Emotions* datasets, while it has an acceptable running time on the *Flags* datasets. The CBA2 algorithm has an acceptable running time among the six considered AC classifiers.

Considering the accuracy and the running-time criteria in the era of distributed computing and high-speed processors, the conclusion can be drawn that CBA2 is the best AC classifier to be adapted to handle the problem of MLC. To make this conclusion reasonable, Figure 2 presents a comparison between the best accuracy results maintained for CBA2 algorithm and several other algorithms that belong to different learning approaches on the three datasets considered in this paper. CBA2 was compared to 12 algorithms that belong to six different learning strategies, where the decision tree learning approach was represented by the C4.5 [56] and DT-GA [57] algorithms. From the neural network learning approach, two algorithms have been considered: GANN [58] and NNEP [59] and fuzzy learning approach was represented by FURIA [60] and WF [61] algorithms. Also, two algorithms that belong to lazy learning have been considered: KNN [62] and KNN-Adaptive [63]. The statistical learning approach was represented by Logistic [64] and LDA [65] algorithms. Finally, the Support Vector Machine (SVM) learning approach was represented by the C-SVM [66] and SMO [67] algorithms.

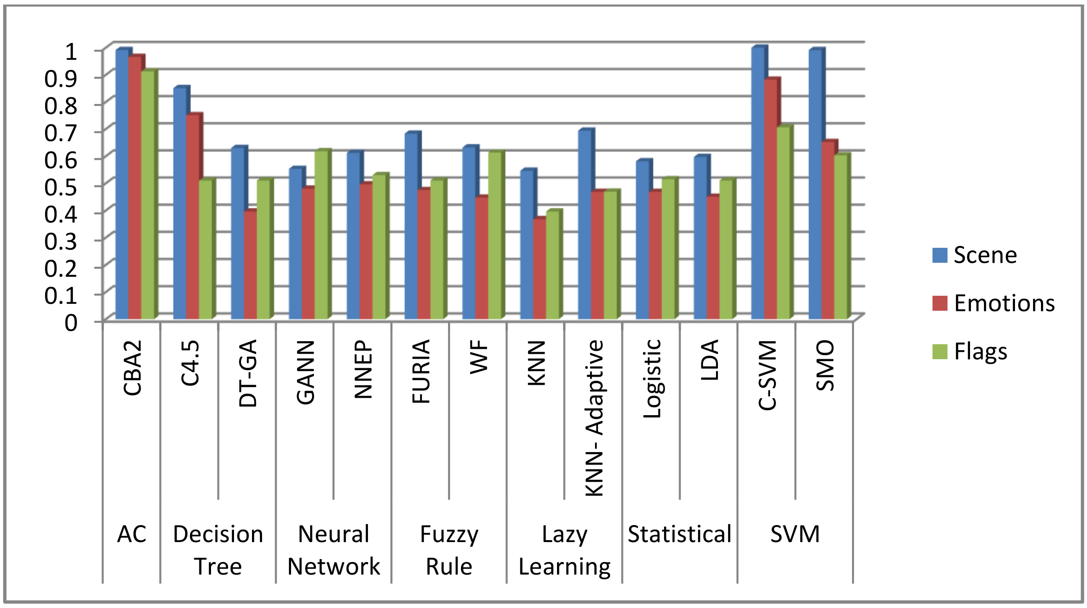

Figure 2. Accuracy rates of CBA2 and several other algorithms on the three datasets.

From Figure 2, clearly the CBA2 algorithm has a superior performance on the *Emotions* and *Flags* datasets. Also, the performance of the CBA algorithm is excellent on the *Scene* dataset, where it has the second-best accuracy after the C-SVM algorithm.

What is distinguishable about CBA2 is that it maintains the same level of performance regardless the characteristics of the datasets, which makes it an excellent choice to handle different multi-label datasets with different characteristics.

To summarize, CBA2 is better than the 12 other algorithms that belong to 6 learning approaches based on the accuracy metric. This fact reveals the significance of adapting the CBA2 algorithm to handle the problem of MLC.

C. Results' Discussion

In General, CBA2 shows a superior performance among the considered AC classifiers on the three datasets based on the accuracy metric. The accuracy of CBA2 has been greatly affected by the discretization technique being used. The results showed that Chi2-D is the most appropriate discretization technique to be used with CBA2 to handle multi-label datasets.

With respect to running time, CPAR has the best running time among the six AC classifiers. Other AC classifiers such as CBA, CBA2 and CMAR, have acceptable running times. Nevertheless, the accuracy of the classification task is more significant than the complexity of the multi-label classifier and its running time, especially in an era of distributed computing and high-efficiency processors. Therefore, CBA2 will be the most promising AC classifier to be adapted to handle the problem of MLC.

Also, a significant issue in determining the applicability of AC algorithm in handling the problem of MLC is the total number of generated rules [14], [30].

Table 8 shows the total number of rules generated by the best four AC classifiers on the three considered datasets with respect to the best three discretization techniques.

Table 8 shows that the total number of rules varies across the four algorithms as well as across the three discretization techniques. The CBA algorithm has the lower values on the three datasets, which makes it an appropriate choice to handle the problem of MLC with respect to the size of the generated classifier. Nevertheless, the accuracy of CBA is less than the accuracy of CBA2. Hence, a trade-off must made between the accuracy results and the size of the classifier results. However, if powerful pruning techniques are utilized, CBA2 will be the best choice to handle MLC problems. Therefore, future work should investigate the most appropriate pruning techniques to be used with AC classifiers to handle MLC datasets that usually suffer from high-dimensionality problems [3], [68].

Table 8. Total number of rules generated by different AC classifiers.

Dataset	Algorithm	Chi2-D	Bayesian-D	Ameva-D	Average
Scene	**CBA2**	235	330	219	261
	CBA	196	**155**	201	**184**
	CMAR	1554	1276	1598	1476
	CPAR	698	1127	841	889
	Average	**671**	722	715	
Emotions	**CBA2**	248	204	171	208
	CBA	78	**41**	71	**63**
	CMAR	195	141	259	198
	CPAR	755	673	487	638
	Average	319	265	**247**	
Flags	**CBA2**	104	91	90	95
	CBA	73	64	**60**	**66**
	CMAR	481	421	443	448
	CPAR	247	213	210	223
	Average	226	**197**	201	

Finally, based on the accuracy of the several AC classifiers on the three datasets, with respect to the accuracy of other algorithms from different learning approaches and strategies, the assumption can be made that AC approach could be more appropriate to be used in the domain of MLC than other learning approaches, especially in a form of ensemble classifiers with ensemble discretization techniques.

4. Conclusion and Future Work

In this paper, an investigation regarding the applicability of AC in solving the problem of MLC has been presented. Six different AC-based classifiers have been evaluated on three multi-label datasets, with respect to five well-known discretization techniques.

Based on the evaluation results, it can be concluded that AC learning approach achieved a superior performance with respect to the accuracy metric compared with the six learning approaches which have been considered in this paper, which indicates that adapting AC to handle the problem of MLC is a promising research work.

Among the considered AC classifiers, CBA2 has shown the best accuracy on the three considered datasets. Also, determining the discretization technique that is optimal to handle multi-label datasets is a crucial decision, where Chi2-D has shown to have an excellent performance when compared with other discretization techniques.

Future work could be done in several areas. First, the CBA2 algorithm could be adapted to handle multi-label datasets. Second, other promising future work is to propose a new MLC algorithm based on an ensemble of several AC classifiers. Third, an investigation regarding the best pruning technique to be used with multi-label datasets that suffer from high number of attributes will be good for future investigation.

REFERENCES

[1] L. Al Qadi, H. El Rifai, S. Obaid and A. Elnagar, "A Scalable Shallow Learning Approach for Tagging Arabic News," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 06, no. 03, pp. 263-280, September 2020.

[2] L. T. Nguyen, B. Vo, T. Mai and T. L. Nguyen, "A Weighted Approach for Class Association Rules," Proc. of Modern Approaches for Intelligent Information and Database Systems, pp. 213-222, Part of the Studies in Computational Intelligence Book Series, vol. 769, Springer, Cham, 2018.

[3] E. Gibaja and S. Ventura, "A Tutorial on Multilabel Learning," ACM Computing Surveys, vol. 47, no. 3, pp. 1-39, 2015.

[4] R. Sousa and J. Gama, "Multi-label Classification from High-speed Data Streams with Adaptive Model Rules and Random Rules," Progress in Artificial Intelligence, vol. 7, pp. 177-187, 2018.

[5] R. Alazaidah, F. K. Ahmad and M. F. M. Mohsen, "A Comparative Analysis between the Three Main Approaches that are Being Used to Solve the Problem of Multi Label Classification," International Journal of Soft Computing, vol. 12, no. 4, pp. 218-223, 2017.

[6] J. Huang, G. Li, S. Wang, Z. Xue and Q. Huang, "Multi-label Classification by Exploiting Local Positive and Negative Pairwise Label Correlation," Neurocomputing, vol. 257, pp. 164-174, 2017.

[7] R. Alazaidah, F. K. Ahmad, M. F. M. Mohsen and A. K. Junoh, "Evaluating Conditional and Unconditional Correlations Capturing Strategies in Multi Label Classification," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 10, no. 2-4, pp. 47-51, 2018.

[8] Y. Qu, G. Yue, C. Shang, L.Yang, R. Zwiggelaar and Q. Shen, "Multi-criterion Mammographic Risk Analysis Supported with Multi-label Fuzzy-rough Feature Selection," Artificial Intelligence in Medicine, vol. 100, Article ID: 101722, DOI: 10.1016/j.artmed.2019.101722, 2019.

[9] R. Alazaidah, F. K. Ahmad, M. F. M. Mohsin and W. A. AlZoubi, "Multi-label Ranking Method Based on Positive Class Correlations," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 6, no. 4, pp. 377-391, Dec. 2020.

[10] R. Alazaidah, F. Thabtah and Q. Al-Radaideh, "A Multi-label Classification Approach Based on Correlations among Labels," International Journal of Advanced Computer Science and Applications, vol. 6, no. 2, pp. 52-59, 2015.

[11] R. Alazaidah, F. Ahmad and M. Mohsin, "Multi Label Ranking Based on Positive Pairwise Correlations among Labels," The International Arab Journal of Information Technology, vol. 17, no. 4, DOI: 10.34028/iajit/17/4/2, 2019.

[12] M. Atzmueller, N. Hayat, M. Trojahn and D. Kroll, "Explicative Human Activity Recognition Using Adaptive Association Rule-based Classification," Proc. of the IEEE International Conference on Future IoT Technologies (Future IoT), pp. 1-6, DOI: 10.1109/FIOT.2018.8325603, Eger, Hungary, 2018.

[13] Y. Chengxin, G. Yan, Y. Jianguo and R. Xiaoting, "A New Recommendation System on the Basis of Consumer Initiative Decision Based on an Associative Classification Approach," Industrial Management & Data Systems, vol. 118, no. 1, pp.188-203, 2018.

[14] N. Abdelhamid, A. A. Jabbar and F. Thabtah, "Associative Classification Common Research Challenges," Proceedings of the 45th IEEE International Conference on Parallel Processing Workshops, pp.432-437, Philadelphia, USA, 2016.

[15] G. Corani and M. Scanagatta, "Air Pollution Prediction *via* Multi-label Classification," Environmental Modelling & Software, vol. 80, pp. 259-264, 2016.

[16] E. C. Gonçalves, A. Plastino and A. A. Freitas, "A Genetic Algorithm for Optimizing the Label Ordering in Multi-label Classifier Chains," Proceedings of the 25th IEEE International Conference on Tools with Artificial Intelligence, pp. 469-476, Herndon, USA, 2013.

[17] X. Li, J. Ouyang and X. Zhou, "Centroid Prior Topic Model for Multi-label Classification," Pattern Recognition Letters, vol. 62, pp. 8-13, 2015.

[18] S. Ali and A. Majid, "Can–Evo–Ens: Classifier Stacking Based Evolutionary Ensemble System for

Prediction of Human Breast Cancer Using Amino Acid Sequences," Journal of Biomedical Informatics, vol. 54, pp. 256-269, 2015.

[19] E. A. Tanaka, S. R. Nozawa, A. A. Macedo and J. A. Baranauskas, "A Multi-label Approach Using Binary Relevance and Decision Trees Applied to Functional Genomics," Journal of Biomedical Informatics, vol. 54, pp. 85-95, 2015.

[20] K. Trohidis, G. Tsoumakas, G. Kalliris and I. P. Vlahavas, "Multi-label Classification of Music into Emotions," Proceedings of the 9th International Conference on Music Informative Retrieval, pp. 325-330, Philadelphia, PN: Drexel University, 2008.

[21] A. Dimou, G. Tsoumakas, V. Mezaris, I. Kompatsiaris and I. Vlahavas, "An Empirical Study of Multi-label Learning Methods for Video Annotation," Proceedings of the 7th IEEE International Workshop on Content-based Multimedia Indexing, pp. 19-24, Chania, Greece, 2009.

[22] T. Li, C. Zhang and S. Zhu, "Empirical Studies on Multi-label Classification," Proc. of the 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06), DOI: 10.1109/ICTAI.2006.55, Arlington, USA, 2006.

[23] Z. Barutcuoglu, R. E. Schapire and O. G. Troyanskaya, "Hierarchical Multi-label Prediction of Gene Function," Bioinformatics, vol. 22, no. 7, pp. 830-836, 2006.

[24] J. Lee, H. Kim, N. R. Kim and J. H. Lee, "An Approach for Multi-label Classification by Directed Acyclic Graph with Label Correlation Maximization," Information Sciences, vol. 351, pp. 101-114, 2016.

[25] M. Alluwaici, A. K. Junoh, F. K. Ahmad, M. F. M. Mohsen and R. Alazaidah, "Open Research Directions for Multi Label Learning," Proc. of the IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), pp. 125-128, DOI: 10.1109/ISCAIE.2018.8405456, Penang, 2018.

[26] S. Xu, X. Yang, H. Yu, D. J. Yu, J. Yang and E. C. Tsang, "Multi-label Learning with Label-specific Feature Reduction," Knowledge-based Systems, vol. 104, pp. 52-61, 2016.

[27] J. Read, B. Pfahringer, G. Holmes and E. Frank, "Classifier Chains for Multi-label Classification," Machine Learning, vol. 85, no. 3, Article no. 333, 2011.

[28] F. Thabtah, "Challenges and Interesting Research Directions in Associative Classification," Proceedings of the 6th IEEE Int. Conference on Data Mining Workshops, pp. 785-792, Hong Kong, China, 2006.

[29] F. A. Thabtah, P. Cowling and Y. Peng, "MMAC: A New Multi-class, Multi-label Associative Classification Approach," Proceedings of the 4th IEEE International Conference on Data Mining, pp. 217-224, Brighton, UK, 2004.

[30] N. Abdelhamid and F. Thabtah, "Associative Classification Approaches: Review and Comparison," Journal of Information & Knowledge Management, vol. 13, no. 3, pp. 1-30, 2014.

[31] Y. Yang, A. Stein, V. A. Tolpekin and Y. Zhang, "High-resolution Remote Sensing Image Classification Using Associative Hierarchical CRF Considering Segmentation Quality," IEEE Geoscience and Remote Sensing Letters, vol. 15, no. 5, pp. 754-758, 2018.

[32] P. B. Shekhawat and S. S. Dhande, "A Classification Technique Using Associative Classification," International Journal of Computer Applications, vol. 20, no. 5, pp. 20-28, 2011.

[33] Y. W. C. Chien and Y. L. Chen, "Mining Associative Classification Rules with Stock Trading Data: A GA-based Method," Knowledge-based Systems, vol. 23, no. 6, pp. 605-614, 2010.

[34] F. Thabtah, Q. Mahmood, L. McCluskey and H. Abdel-Jaber, "A New Classification Based on Association Algorithm," J. of Information & Knowledge Management, vol. 9, no. 1, pp. 55-64, 2010.

[35] R. AlShboul, F. Thabtah, N. Abdelhamid and M. Al-Diabat, "A Visualization Cybersecurity Method Based on Features' Dissimilarity," Computers & Security, vol. 77, pp. 289-303, 2018.

[36] B. Liu, W. Hsu and Y. Ma, "Integrating Classification and Association Rule Mining," Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, pp.80-86, American Association for Artificial Intelligence, 1998.

[37] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," Proceedings of the 20th International Conference on Very Large Databases, pp. 487-499, Santiago, Chile, 1994.

[38] B. Liu, Y. Ma and C. K. Wong, "Improving an Association Rule Based Classifier," Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery, pp. 504-509, Berlin, Heidelberg: Springer-Verlag, 2009.

[39] S. J. Huang, Z. H, Zhou and Z. Zhou, "Multi-label Learning by Exploiting Label Correlations Locally," Proceedings of the 26th AAAI Conf. on Artificial Intelligence, pp. 949-955, Palo Alto, CA: AAAI, 2012.

[40] X. Yin and J. Han, "CPAR: Classification Based on Predictive Association Rules," Proceedings of the 2003 SIAM International Conference on Data Mining, pp. 331-335, Society for Industrial and Applied Mathematics, Philadelphia, PN, USA, 2003.

[41] W. Li, J. Han and J. Pei, "CMAR: Accurate and Efficient Classification Based on Multiple Class-association Rules," Proc. of the IEEE Int. Conf. on Data Mining, pp. 369-376, San Jose, USA, 2001.

[42] Y. C. Hu, R. S. Chen and G. H. Tzeng, "Finding Fuzzy Classification Rules Using Data Mining Techniques," Pattern Recognition Letters, vol. 24, no. 1-3, pp. 509-519, 2003.

[43] Z. Chen and G. Chen, "Building an Associative Classifier Based on Fuzzy Association Rules," International Journal of Computational Intelligence Systems, vol. 1, no. 3, pp. 262-273, 2008.

[44] N. Abdelhamid, A. Ayesh and W. Hadi, "Multi-label Rules Algorithm Based Associative Classification," Parallel Processing Letters, vol. 24, no. 1, pp. 1-21, 2014.

[45] A. Veloso, W. Meira, M. Gonçalves and M. Zaki, "Multi-label Lazy Associative Classification," Proc. of the European Conference on Principles of Data Mining and Knowledge Discovery, pp. 605-612, Berlin, Heidelberg:Springer-Verlag, 2007.

[46] R. E. Schapire and Y. Singer, "BoosTexter: A Boosting-based System for Text Categorization," Machine Learning, vol. 39, no. 2-3, pp. 135-168, 2000.

[47] X. Li, D. Qin and C. Yu, "ACCF: Associative Classification Based on Closed Frequent Itemsets," Proceeding of the 5th IEEE International Conference on Fuzzy Systems and Knowledge Discovery, pp. 380-384, Jinan, China, 2008.

[48] J. Han, J. Pei, Y. Yin and R. Mao, "Mining Frequent Patterns without Candidate Generation: A Frequent-pattern Tree Approach," Data Mining and Knowledge Discovery, vol. 8, no. 1, pp. 53-87, 2004.

[49] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek and I. Vlahavas, "Mulan: A Java Library for Multi-label Learning," Journal of Machine Learning Research, vol. 12, no. July, pp. 2411-2414, 2011.

[50] I. Triguero, S. González, J. M. Moyano, S. García, J. Alcalá-Fdez, J. Luengo et al., "KEEL 3.0: An Open Source Software for Multi-stage Analysis in Data Mining," International Journal of Computational Intelligence Systems, vol. 10, no. 1, pp. 1238-1249, 2017.

[51] H. Liu and R. Setiono, "Feature Selection *via* Discretization," IEEE Transactions on Knowledge and Data Engineering, vol. 9, no. 4, pp. 642-645, 1997.

[52] X. Wu, "A Bayesian Discretizer for Real-valued Attributes," The Computer Journal, vol. 39, no. 8, pp. 688-691, 1996.

[53] L. Gonzalez-Abril, F. J. Cuberos, F. Velasco and J. A. Ortega, "Ameva: An Autonomous Discretization Algorithm," Expert Systems with Applications, vol. 36, no. 3, pp. 5327-5332, 2009.

[54] R. C. Holte, "Very Simple Classification Rules Perform Well on Most Commonly Used Datasets," Machine Learning, vol. 11, no. 1, pp. 63-90, 1993.

[55] C. T. Su and J. H. Hsu, "An Extended Chi2 Algorithm for Discretization of Real Value Attributes," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 3, pp. 437-441, 2005.

[56] J. R. Quinlan, "Combining Instance-based and Model-based Learning," Proceedings of the 10th International Conference on Machine Learning, pp. 236-243, Burlington, MA: Morgan Kauffman, 1993.

[57] D. R. Carvalho and A. A. Freitas, "A Hybrid Decision Tree/Genetic Algorithm Method for Data Mining," Information Sciences, vol. 163, no. 1-3, pp. 13-35, 2004.

[58] G. F. Miller, P. M. Todd and S. U. Hegde, "Designing Neural Networks Using Genetic Algorithms," Proceedings of the 3rd International Conference on Genetic Algorithms, pp. 379-384, Hillsdale, IL: Morgan Kaufmann, 1989.

[59] F. J. Martínez-Estudillo, C. Hervás-Martínez, P. A. Gutiérrez and A. C. Martínez-Estudillo, "Evolutionary Product-unit Neural Network Classifiers," Neurocomputing, vol. 72, no. 1-3, pp. 548-561, 2008.

[60] J. Hühn and E. Hüllermeier, "FURIA: An Algorithm for Unordered Fuzzy Rule Induction," Data Mining and Knowledge Discovery, vol. 19, no. 3, pp. 293-319, 2009.

[61] T. Nakashima, G. Schaefer, Y. Yokota and H. Ishibuchi, "A Weighted Fuzzy Classifier and Its Applica-

tion to Image Processing Tasks," Fuzzy Sets and Systems, vol. 158, no. 3, pp. 284-294, 2007.

[62] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21-27, 1967.

[63] J. Wang, P. Neskovic and L. N. Cooper, "Improving Nearest Neighbor Rule with a Simple Adaptive Distance Measure," Pattern Recognition Letters, vol. 28, no. 2, pp. 207-213, 2007.

[64] S. Le Cessie and J. C. Van Houwelingen, "Ridge Estimators in Logistic Regression," Applied Statistics, vol. 41, no. 1, pp. 191-201, 1992.

[65] G. J. McLachlan, Discriminant Analysis and Statistical Pattern Recognition, vol. 544. John Wiley & Sons, ISBN: 978-0-471-69115-0, 2004.

[66] C. Cortes and V. Vapnik, "Support-vector Networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.

[67] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya and K. R. K. Murthy, "Improvements to Platt's SMO Algorithm for SVM Classifier Design," Neural Computation, vol. 13, no. 3, pp. 637-649, 2001.

[68] R. Alazaidah and F. K. Ahmad, "Trending Challenges in Multi Label Classification," International Journal of Advanced Computer Science and Applications, vol. 7, no. 10, pp. 127-131, 2016.

ملخص البحث:

يُعّد التّصنيف متعدّد العناوين (MLC) مجالاً مثيراً ومُهماً جذب اهتمام الكثير من الباحثين في العقدين الأخيرين. فقد تمّ تكييف عدّة خوارزميّات في تصنيف مفردة العنوان تنتمي الى استراتيجيّات تعلّم مختلفة واستخدامها للتّعامل مع مشكلة التّصنيف متعدّد العناوين. ومن المفاجيء أنّه لم يتمّ استخدام أي خوارزمية تصنيف تشاركي للتّعامل مع مشكلة التّصنيف مُتعدّد العناوين؛ علماً بأنّ خوارزميّات التّصنيف التّشاركي أظهرت أداءً توقعياً عالياً مقارنة بغيرها من استراتيجيّات التعلّم في مجال التّصنيف مُفرد العنوان. هذه الورقة عبارة عن استقصاءٍ عميق لاستخدام التّصنيف التّشاركي في مجال التّصنيف متعدّد العناوين. فقد جرى تقييم عدّة خوارزميات تصنيف تشاركي على ثلاث مجموعات بيانات متعددة العناوين باستخدام خمس تقنيات تجريد، وبينت النتائج أن استخدام خوارزميّات التّصنيف التّشاركي في مجال التّصنيف متعدّد العناوين هو مجال واعد إذا قورن باستخدام خوارزميّات أخرى من استراتيجيّات تعلّم مختلفة.