Multi-Label Arabic Text Classification: An Overview

Nawal Aljedani[1], Reem Alotaibi[2] and Mounira Taileb[3]
Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—There is a massive growth of text documents on the web. This led to the increasing need for methods that can organize and classify electronic documents (instances) automatically. Multi-label classification task is widely used in real-world problems and it has been applied on different applications. It assigns multiple labels for each document simultaneously. Few and insufficient research studies have investigated the multi-label text classification problem in the Arabic language. Therefore, this survey paper aims to present an extensive review of the existing multi-label classification methods and techniques that can deal with multi-label problem. Besides, we focus on Arabic language by covering the relevant applications of multi-label classification on the Arabic text, and identify the main challenges faced by these studies. Furthermore, this survey presents an experimental comparisons of different multi-label classification methods applied for the Arabic context and points out some baseline results. We found that further investigations are also needed to improve the multi-label classification task in the Arabic language, especially the hierarchical classification task.**

Keywords*—Machine learning; text classification; multi-label classification; Arabic natural language processing; hierarchical classification; Lexicon approach

I. Introduction

With the emergence of unstructured data, social media mining, and a massive growth of text documents on the web, text classification (TC) has become an essential task that can be solved using machine learning. It is used for organizing a huge number of electronic documents (instances) efficiently [1]. In general, it can be formally defined as a supervised machine learning technique where a classification model is trained using a training data which consist of a set of instances and their associated labels (categories). In classification, the objective is to classify unseen instances using the trained model by assigning the appropriate labels to an unseen instance [2].

In the literature, two approaches are used for classification as shown in Fig.1; single-label classification, which is the traditional type of classification, it is concerned with assigning only one predefined label to each instance. In contrast, in multi-label classification (MLC), a set of predefined labels associated with the instance simultaneously [3]. Generally, it is usually inadequate to classify each instance under just one single label, because there are several labels that might be suitable to describe its content concurrently [4]. For example, a news article assigned with “education" label, may assigned with several labels simultaneously e.g., “social" and “technology".

MLC task is widely used in real-world problems and it has been applied on different applications like classification of digital libraries, electronic emails, electronic books, patents, and newspaper articles. Many studies have addressed the MLC problem in English language. However, regarding the Arabic language only limited and insufficient studies have been conducted in the MLC field [5].

Arabic language is “the native language of 380 million speakers" [6], and considered from the “six official languages of the United Nations" [7]. It has vast vocabulary and complex morphology [8]. Moreover, since online data in Arabic are increasing rapidly especially in the recent days. As a result, there is a need to develop an automatic text classification technique can organize and categorize such amount of electronic Arabic text documents efficiently. Arabic text classification task are well studied using traditional single label classification algorithms e.g., Naive Bayes (NB) [9], k-Nearest Neighbor (k-NN) [10], and Support Vector Machine (SVM) [11].

However, multi-label Arabic text classification is not well addressed. According to our performed review and the study conducted in [5], there are few researches have been conducted in this field on a small and non-publicly available datasets. Consequentially, MLC in the context of Arabic language is becoming a significant topic in the recent years and attracting attention of many researchers.

Therefore, the main contribution of this paper is to conduct an extensive review to get knowledge of the existing methods and techniques that can be applied to deal with MLC problems. It also contributes to organizing the sparse state-of-the-art MLC methods into a structured presentation. Besides, a set of common multi-label evaluation metrics used to evaluate MLC models have been presented. Furthermore, it focuses on the Arabic language by covering the existing studies that are applied in the Arabic context and identifies the main challenges faced by these studies. An experimental comparisons of several state-of-the-art MLC methods in the Arabic context are also provided.

A structured representation of different state-of-the-art MLC methods is provided in the text classification taxonomy shown in Fig. 1. According to this taxonomy, two approaches are used to solve MLC problems: lexicon approach which is based on creating a dictionary for each label that contains the list of all related words, or machine learning approach that relied on labelled instances. The two approaches are discussed in the following sections (see Section II and Section III).

The rest of the paper is organized as follows. Section II discusses MLC using lexicon approach. After that, Section III discusses MLC using machine learning approach. It also reports an extensive review of MLC methods by providing illustrative examples and discussing their advantages and limitations. The most common multi-label evaluation metrics are presented in Section IV. Then, Section V focuses on the Arabic language and presents a set of applications for multi-label Arabic text classification. It also identifies MLC techniques which have been applied in each application and briefly discusses some important challenges and remarks for future studies. An experimental comparisons of the state-of-the-art MLC methods for the Arabic text are conducted in

Section VI. Finally, Section VII concludes the paper.

II. MLC using Lexicon Approach

Lexicon-based classification as presented in [12] means that each instance is assigned to a label based on a classification rule which considers the count of words from lexicons of each label. Obviously, the basic rule is to predict the label that most of words in the instance are associated with its lexicon. Generally, lexicon-based approach has several advantages such as: it is intuitive, easy to use, simple, and makes the classification faster compared with labelled instances.

On the other hand, from machine learning perspective, lexicon-based classification suffers from some drawbacks. One of the most drawbacks is the lack of theoretical justification and it is not clear what conditions are required for it to work. Moreover, it assigns a similar weight for each word and this is not reasonable because some words are strongly predictive compared to others. In addition, the lexicon-based approach ignores multi-word phenomena, for example negation (e.g., not so good), and the lexicons might be incomplete.

MLC using machine learning approach which are trained on labelled instances, seem to outperform lexicon-based classification even without considering the multi-word phenomena. Lexicon-based classification is widely used in opinion mining and sentiment classification [12]. However, in some applications, it is used as a MLC approach [13].

III. MLC using Machine Learning Approach

Solving multi-label problem using machine learning approach can be divided as shown in Fig. 1 into flat and hierarchical classifications.

A. Flat Classification

In the flat classification, the set of predefined labels are treated independently and classified in one level where they are not organized in a structure that defines the relationship among these labels [14] as shown in Fig. 2. The flat classification in multi-label problem divided into two techniques, which are problem transformation (PT) and algorithm adaptation technique.

PT can be defined as transformation of the multi-label problem into a set of single-label problems which are classified using classical (traditional) single-label classification algorithms. Whereas, algorithm adaptation are those methods that adapt single-label classification algorithms to deal directly with multi-label problem. It is worth mentioning that algorithm adaptation methods are algorithm-dependent and PT methods are algorithm-independent [2].

1) Strategies based on PT Technique: PT technique is considered as the simplest way to classify multi-label data because it simply transforms the data into single-label problems, then classical single-label classification algorithms are applied to perform the classification task. Basically, there are two straightforward PT techniques, either based on transformation to binary classification or transformation to multi-class classification.

The classification task in both techniques is based on single-label classifiers. But, concerning the multi-class classification, the similar set of labels are combined as a distinct class. Thus, every similar set of labels that found in the training dataset is considered a new class of a multi-class classification task [15]. The most common PT methods which are based on transformation to binary classification are described in the following.

Binary Relevance (BR). As presented in [2], a simple and straightforward method used to handle MLC task is BR method. Simply, it transforms a MLC problem into several single-label classification problems and predicts the instance relevance for each single-label independently by training a binary classifier one per label.

The instances of the original multi-label data are included in each single-label data and they are predicted with positive label if they have the existing label, otherwise, they are predicted with negative label. The classification of a new (unseen) multi-label problem using BR is performed by transforming it into n single-label problems (where n refers to number of labels). Consequently, the new instance will be labelled with a union of the positive labels predicted by the n-binary classifiers [15], as shown in Fig. 3.

BR presents several obvious advantages such as it is relatively fast and conceptually a simple method, it has a linear complexity related to the number of labels, so it has a low computational complexity. In addition, since it does not consider labels correlation, therefore it is possible to add or remove labels without having an effect on the rest of the BR model which makes it appropriate for a dynamic or evolving scenario and it can be easily parallelized.

However, BR has some drawbacks; it does not consider any label correlations and might reduce the predictive performance if such label dependency is present. Moreover, increasing the label dimensionality causes an increase in the number of trained binary classifiers. In addition, it suffers from sample imbalance problem that may occur when number of negative instances outperform the positive ones [2], [16].

Classifier Chain (CC). There are several methods that have been proposed to minimize the BR drawbacks. The most popular one is the CC method proposed in [17]. CC method is like BR method with small difference which is that, it resolves BR drawback by considering label correlations. Obviously, it transforms the MLC problem into a chain of binary classification problems where each subsequent binary classifier in the chain is extended with 0, 1 label predictions of all preceding classifiers, and these labels predictions are considered as future attributes for the next classifier. Chaining method passes the label information among the labels making the CC considering correlation in the label space. An example of CC method is presented in Fig. 4.

The main CC advantages are that it considers label dependency to obtain high predictive performance while maintaining a reasonable computational complexity of BR. It is straightforward to predict label from this chain by obtaining the label of one classifier and propagating it along the chain [17].

Furthermore, it has some drawbacks; since CC has chaining property it loses the opportunity of parallel implementation

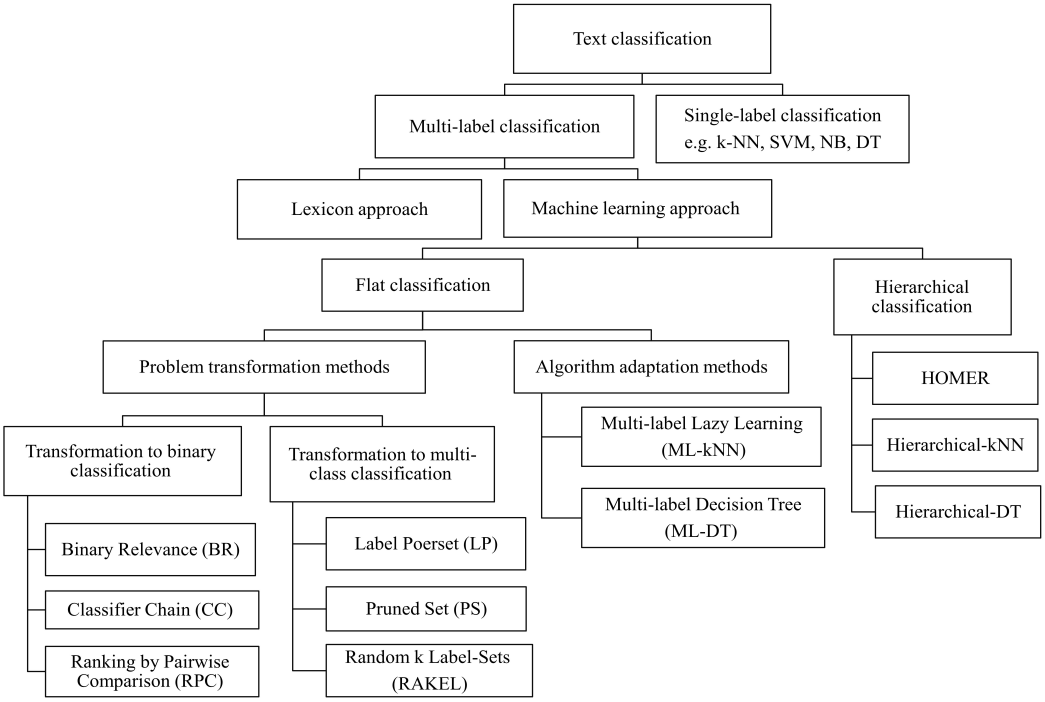

Fig. 1. Text Classification Taxonomy

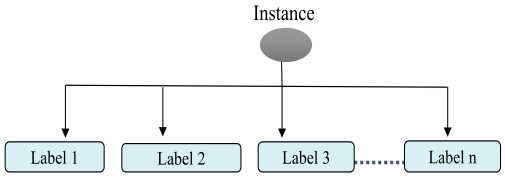

Fig. 2. Example of an Instance with Multi-Labels in the Flat Classification

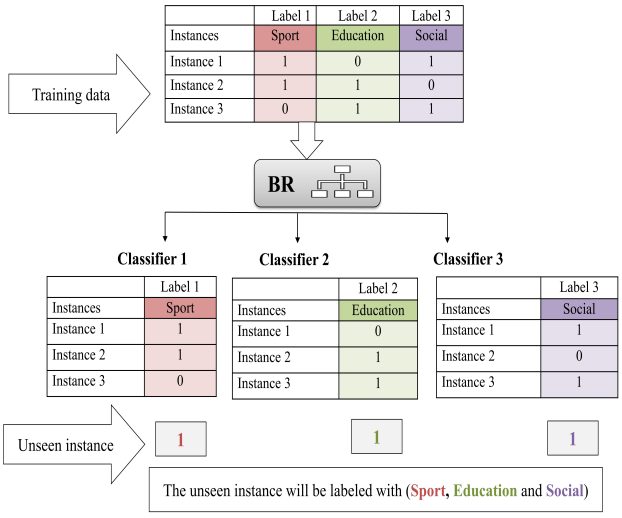

Fig. 3. Binary Relevance Method

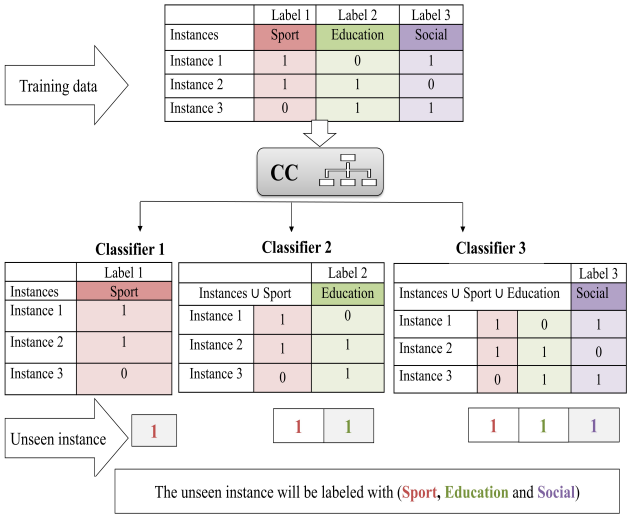

Fig. 4. Classifier Chain Method

[18]. In addition, the chaining property in CC also cause an error propagation at the classification time in case of a poor prediction from one or more of the preceding classifiers. This might affect the prediction of the following labels. Moreover, the poorly ordered of the chain can affect the prediction accuracy. Thus, an ensembles of CC method has been proposed to overcome this issue by training the CC classifiers with random order of chains [17].

Ranking by Pairwise Comparison (RPC). RPC proposed in [19] is another type of PT methods, it is considered as an extension of the original Pairwise Comparison (PC) method.

RPC is a sequence of two processes; in the first one, PC are trained with a given data. This means that the multi-label dataset with n labels is transformed into $n(n\text{-}1)/2$ single-label binary classification problems, where for each pair of labels a binary classifier is used by covering all pairs of labels.

Each single binary classification problem involves the instances in the original multi-label problem, which are assigned to just one of the two labels but not both, as shown in Fig. 5. First, in the label prediction task of each classifier, the classifier performs a comparison between each pair of labels. For example, if it compares between label 1 and label 2, thus, the resulted predicted label, let us call it y, will be (0 or 1) according to Eq. (1).

$$y = \begin{cases} 1, & Label1 \succ Label2 \\ 0, & Label2 \succ Label1 \end{cases} \quad (1)$$

Then, in the second step, the predicted labels are ranked using a ranking procedure such as a generalization of voting strategy where all labels are ranked based on the evaluated sum of the weighted votes. To classify a new instance using RPC method, each binary classifier is invoked and voted to one of the two labels. After prediction of labels by all classifiers, the labels' ranking is obtained according to the sum of votes of each label.

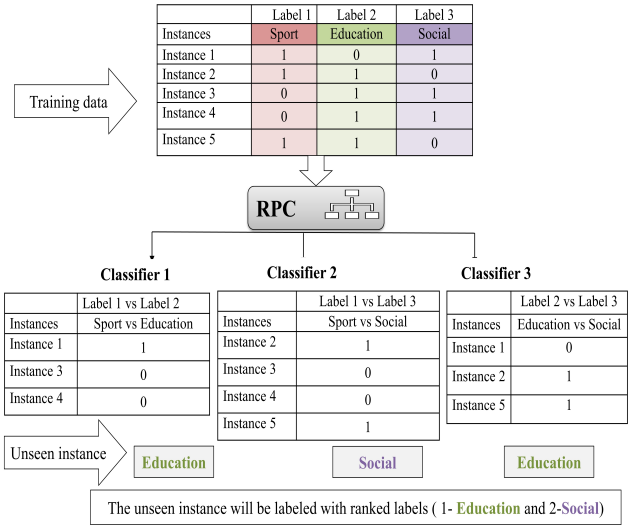

Fig. 5. Ranking by Pairwise Comparison Method

The main disadvantage of the RPC is the need to query all the generated classifiers at the classification time [2]. In addition, a quadratic complexity (n^2) of RPC makes it very sensitive to the large number of labels and usually it is difficult to deal with large problems. However, more experiments showed that RPC is a competitive method in terms of accuracy and prediction quality. In addition, the extended version of the RPC uses the ensemble of pairwise classifiers to adapt to different loss functions on label ranking strategy without retraining the pairwise classifiers [19].

A summary of pros and cons of the above mentioned PT methods based on transformation to binary classification is presented in Table I. In the following we present the most common PT methods based on transformation to multi-class classification.

Label Powerset (LP). It is the most simple and standard method under this classification technique. It transforms the MLC problem into a multi-class classification problem, and considers each distinct label-set in the training data as a new class of a multi-class classification task [20]. To classify a new multi-label instance using LP, a multi-class classifier can be used to assign the most probable class from many new classes to this instance, which can then be reversible to the corresponding set of the initial labels. In the example shown in Fig. 6, the classifier will randomly predict a class for an unseen instance. Since there are two classes with 50% chance to be assigned to the unseen instance.

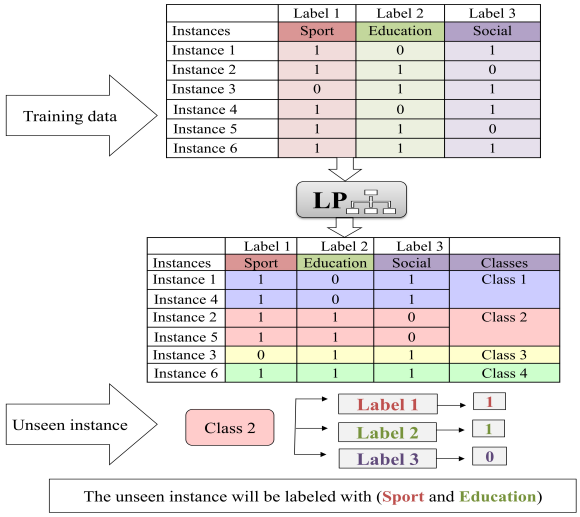

Fig. 6. Label Powerset Method

Although, LP has many advantages such as its simplicity, effectiveness, and consideration of correlations among labels in the training data. In contrast, it suffers from some drawbacks. The first problem is, it has a computational complexity increases exponentially with the number of labels and that is why LP quickly deteriorates for larger label-sets and makes the work of the classifier harder [15]. Consequently, LP is usually recommended just for datasets that have small number of distinct classes. On the other hand, since LP method can consider only the label-sets present in the training data, so it can not classify any unseen label-set. In addition, for many new classes, it is potential to have limited training instances due to the infrequent combinations of label-sets for those classes, so this lead to the class imbalance problem [2].

Two variants of LP method have been proposed, the first, is Pruned Set method proposed in [21], and the second proposed method is Random k-Labelsets [22].

Pruned Set (PS). The main idea of PS method is similar to LP, it extends the same paradigm of LP whereas trying to resolve the limitations related to LP complexity [21]. This is achieved by pruning away the label-sets with a frequency of occurrence less than a threshold defined by the user. Hence, it reduces the LP complexity by keeping just the label-set that are occurring more frequently by comparing it to the threshold.

TABLE I. PROS AND CONS OF PT METHODS BASED ON TRANSFORMATION TO BINARY CLASSIFICATION

PT method	Pros	Cons
BR	• Relatively fast. • Conceptually simple. • Low computational complexity. • Appropriate for a dynamic scenario. • Can be easily parallelized. • Scale up linearly with the number of labels.	• Ignores labels correlations. • May reduce predictive performance. • An increase in the number of labels causes the increase in the number of binary classifiers.
CC	• Considers labels dependencies. • Obtains high predictive performance. • Maintains acceptable computational complexity. • Straightforward to predict label from chain.	• Loses the opportunity of parallel implementation. • Might cause an error propagation at the classification time. • Poorly ordered chain that might affect the accuracy prediction.
RPC	• Provides high accuracy and prediction quality. • Does not require to retrain the pairwise classifiers if the label ranking method change.	• All generated binary classifiers are queried at the run-time. • Quadratic complexity. • Very sensitive to large number of labels.

An example is illustrated in Fig. 7 using PS method, it keeps only class 1 and class 2 because they have more frequent label-sets in the training data compared to the other classes and prunes away both of class 3 and class 4 represented in Fig. 6, because they occurred infrequently. However, it is similar to LP when taking into account only the distinct label-sets present in the training data. In addition, it suffers from class imbalance problem.

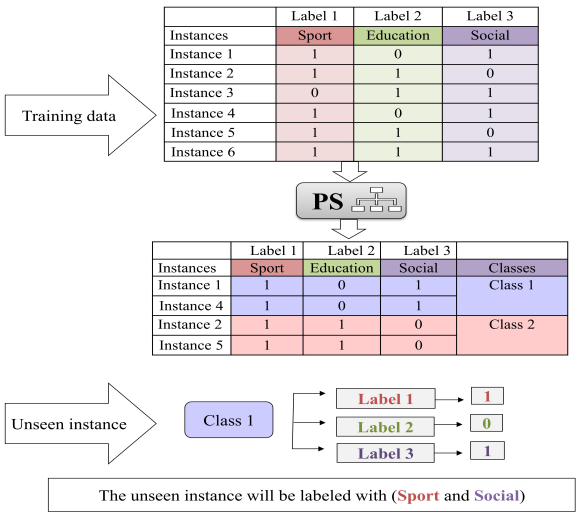

Fig. 7. Pruned Set Method

The classification of an unseen instance using PS method is illustrated with an example in Fig.7. It is similar to LP; a multi-class classifier is applied to randomly assign a probable class to this instance, which can be reversible to the corresponding set of the initial labels. Since there are two classes with 50% chance to be assigned to the unseen instance.

Random *k*-Labelsets (RAKEL). RAKEL method also has resolved LP problems by generating an ensembles of LP method by breaking the multi-label problem into r models or subsets [22]. Each subset is assigned with a random k label-sets where k is a random number used to determine the size of each model. The label-sets in each model can be either disjoint or overlapping according to the strategy used to construct them.

To classify a new instance, RAKEL queries all models and obtains their average decision for each label. In addition, it makes the final prediction based on the value of the given threshold, so the final label prediction is positive when the average decision is greater than the threshold and negative otherwise.

Furthermore, it provides some advantages since it considers label correlation and overcomes LP limitation by increasing number of distinct label-sets and thus it provides more accurate label prediction and competitive performance. Nevertheless, it suffers from the increasing number of classifiers generated according to a k random number used to determine the size of each model. The pros and cons of the above mentioned PT methods based on transformation to multi-class classification are summarized in Table II.

2) Strategies based on Algorithm Adaptation Technique: Overall, classical single-label classification algorithms can not deal directly with MLC problem. Consequently, algorithm adaptation technique has been developed to tackle MLC problem by extending and adapting single-label classification algorithm to be able to directly deal with MLC problem [23]. Almost most of classical single-label classifiers have been adapted to handle MLC problems directly [2].

For example, SVM algorithm has been adapted to Rank-SVM in [24], Neural Network algorithm has been customized as the baseline algorithm for a Back-Propagation Multi-Label Learning (BP-MLL) algorithm in [25]. Besides, a Multi-label Lazy Learning algorithm called ML-kNN proposed in [26] extended from the classical k-NN algorithm. In addition, Multi-Label Decision Tree algorithm (ML-DT) has been developed in [27], [28] by adapting the C4.5 decision tree algorithm. The most common algorithms in the algorithm adaptation context are detailed in the following paragraphs.

Multi-Label Lazy Learning (ML-kNN). MLC task consists of training a classification model using a specific algorithm to predict several labels for each unseen instance by analyzing labelled training instances. ML-kNN was proposed

TABLE II. PROS AND CONS OF PT METHODS BASED ON TRANSFORMATION TO MULTI-CLASS CLASSIFICATION

PT method	Pros	Cons
LP	• Simple and effective. • Considers labels correlations. • Works well only when using datasets with a small number of classes.	• Exponential increase of the computational complexity if the number of labels increases. • Class imbalance problem. • Can not classify any unseen label-set in the training data.
PS	• Considers labels correlations. • Reduces LP complexity.	• Class imbalance problem. • Can not classify any unseen label-set in the training data.
RAKEL	• Considers labels correlations. • Reduces LP complexity. • Provides more accurate label prediction. • Considers any unseen label-set even if it is not present in the training data.	• Increasing number of generated classifiers.

in [26], which is a MLC algorithm derived from the classical k-NN algorithm.

To predict the label-sets for a new instance using ML-kNN, the algorithm follows the same approach of the classical k-NN algorithm. First: the *k* nearest neighbors from the training instances are determined. Usually, “Euclidean distance" is used to compute the distance as well as similarity between instances to identify the *k* neighboring instances. The second step, which is the additional step in the ML-kNN algorithm focuses on label-sets aggregation of the *k* neighboring instances. It utilizes “Maximum A Posteriori (MAP) principle" to determine the more probable label-sets for the new instance by relying on prior and posterior probabilities for the frequency of each label in the *k* nearest neighbors. ML-kNN as well as can output an ordered list of ranking labels [3].

Finally, to validate the approach, the authors conducted some experiments on the three real world problems. The result showed that ML-kNN achieved better performance than some other MLC algorithms, e.g., Rank-SVM, Adtboost.Mh, and Boostexter.

Multi-Label Decision Tree (ML-DT). Regarding the DT algorithm, the C4.5 algorithm has been adapted in [27] to ML-DT algorithm to deal with MLC problem directly. The adaptation is accomplished by modifying the original entropy formula of C4.5 algorithm for solving MLC problems, where entropy is a metric of the amount of uncertainty in the dataset. The entropy formula was modified for multiple labels to compute the weighted sum of all entropies for each individual label in each subset as shown in Eq. (2).

$$Entropy = -\sum_{i=1}^{n} ((p(L_i) log\, p(L_i)) + (q(L_i) log\, q(L_i))) \quad (2)$$

Where; n = number of labels, $p(L_i)$= membership probability (relative frequency) of label L_i in a data subset, $q(L_i) = 1 - p(L_i)$ = non-membership probability of label L_i in a data subset. Consequently, allowing ML-DT to handle multi-label problem means it should allow multiple labels for the leaf nodes in the tree. Besides, in case of generating rules in leaf nodes, these rules could output a set of labels.

The authors in [28], developed a multi-label decision tree algorithm (LaCovaC) based on C4.5 that learns the labels relations and exploits them to improve the predictive performance.

B. Hierarchical Multi-Label Classification

Hierarchical multi-label classification (HMC) is considered as an extension or variant of MLC where a hierarchy structure is used on the multi-label. In HMC, an instance is assigned with multiple labels concurrently and those labels are structured in a hierarchy [29]. Moreover, an instance should satisfy the hierarchy constraint, which means that if it belongs to some label *L* it automatically belongs to all super-labels of *L* [30].

There are many classification tasks in the real world that can be considered as HMC problems, where the predicted labels are classified as a hierarchy, typically as a tree-shaped or a directed a cyclic graph (DAG) [31]. However, according to a survey conducted in [32], most of the current researches are only concerned with the tree structured hierarchy as shown in Fig. 8. In general, HMC is considered as a more challenging task by nature compared to the flat classification [31]. A comparison between flat and hierarchical classification is illustrated in Table III.

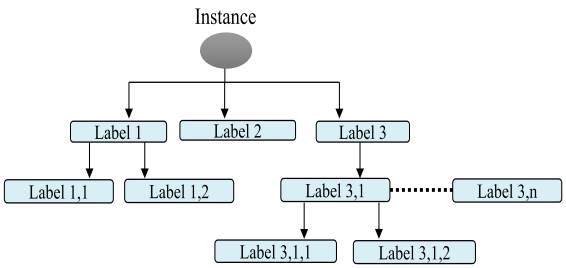

Fig. 8. Example of an Instance with Multi-Labels Classified using Hierarchical Multi-Label Classification; Labels are Structured in a Tree-Shaped Hierarchy

1) Hierarchical Multi-Label Classification Algorithms: Several algorithms have been proposed to deal with HMC problem such as: Hierarchy Of Multilabel classifiERs algorithm (HOMER) [36], Hierarchical Decision Tree algorithm [37], Hierarchical k-NN algorithm [14], Incremental algorithm for hierarchical classification [38], Hierarchical-SVM [39], and

TABLE III. Comparison between flat and hierarchical classification.

	Flat classification	Hierarchical classification
Basic concept	Classify labels in MLC problem as one level, where they are not organized as a structure that defines the relationship among these labels.	Imposing a hierarchy structure on MLC problem, where an instance belongs to many labels and the labels are structured in a hierarchy.
Properties	• Conceptually simple. • Ignoring the label hierarchy. • Difficult to handle a large number of labels. • Handle MLC task of unstructured documents. • Divided into PT and algorithm adaptation methods.	• More challenging approach. • Considering the label hierarchy. • Capable to handle a large number of labels. • Handle MLC task of structured documents. • Structured as tree or DAG hierarchy.
Relevant studies conducted on Arabic text using machine learning algorithms	[33], [5], [34], [4].	[35].

HMC using Fully Associative Ensemble Learning [40]. We discuss some of them in the following paragraphs.

Hierarchy Of Multilabel classifiERs (HOMER). HOMER algorithm is an effective hierarchical multi-label classifier, relies on divide-and-conquer approach [36]. This algorithm can efficiently handle a MLC problem with a large number of labels by constructing a tree-shaped hierarchy of simpler MLC problems. Then, each classifier can handle MLC problem with a small number of labels rather than handling a large label-set.

Labels distribution task in this algorithm is done using the top-down approach. That means the large label-sets should be distributed into *k* disjoint label-sets based on similarity, starting from parent to child nodes. This task is usually done using balanced clustering algorithm. However, in [36] the authors proposed a clustering algorithm for labels distribution called balanced *k*-means clustering algorithm.

Consequently, each node in the tree involves the union of meta-labels μ of its children and the root node contains labels of all nodes in the tree. Besides, each internal node also has a multi-label classifier *S* to predict one or more meta-labels of its children. Figure 9 illustrates the tree hierarchy in HOMER for the classification task of multi-label problem with 9-labels $\{L1, ..., L9\}$.

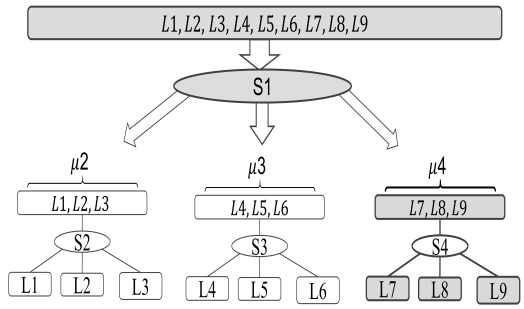

Fig. 9. Example of a Classification Task of Multi-Label Problem with 9 Labels using HOMER Algorithm

For multi-label classification task of an unseen instance *e*, HOMER starts from the root node then follows a top-down approach to forwards *e* to the most relevant multi-label classifier *S*. In the example of HOMER illustrated in Fig. 9, the multi-label classifier *S1* at the root node will forward the instance *e* to the multi-label classifier *S4* only if μ_4 is among the predictions of *S1* classifier. Finally, by following a recursive process, the result of the final prediction labels will be a union of all predicted labels by the multi-label classifier that higher than the corresponding leaf(ves) nodes. Otherwise, if no labels are predicted, the algorithm returns an empty set.

The authors have conducted some experiments using HOMER algorithm by employing the BR as a multi-label classifier and NB as a base classifier. The results showed that, since HOMER relies on similarity-based distribution and employed BR and NB classifiers, this reduces the computational cost in both training and test phases and it improves the predictive performance as well.

Hierarchical Multi-Label Classification using k-NN Algorithm. The authors in [14] have proposed a new framework by extending the modified version of k-NN algorithm and proposed a new similarity measure to handle HMC task. The classical k-NN algorithm was modified to work with label representative instead of training instances. Where each label is represented by one instance constructed from all instances in the training data related to this label. This process is done by identifying the most important features of that label and combining the training instances accordingly. By using label representative, the classification time have been reduced because a new instance will be compared with just a portion of instances rather than all the training instances.

Several methods can be used to measure the instance similarity such as, cosine similarity, Jaccard, and dice. These measures calculate the similarity based on instance vector, where each vector consists of instance features along with their frequencies in the training instances. However, the proposed framework relies on a new proposed similarity measure named New Expected Information value I_{new}. It is calculated according to only the shared features between label representative and a test instance, this resulted to a smaller vector compared to the other similarity measures mentioned previously.

The authors have tested and evaluated the proposed framework by conducting an experiment using a test dataset. The results revealed that applying the proposed classifier with I_{new} similarity measure has reduced the classification time compared to the other similarity measures. Besides, precision and recall evaluation metrics showed that using I_{new} measure performed results close to the three similarity measures (cosine

similarity, Jaccard, and dice).

Decision Tree for Hierarchical Multi-Label Classification. The authors in [37] have investigated the suitability of the classical DT algorithm to HMC task. They have instantiated three decision tree algorithms: single-label classification (SC) algorithm, hierarchical single-label classification (HSC) algorithm, and hierarchical multi-label classification (HMC) algorithm. The three proposed algorithms are based on "predictive clustering trees (PCTs) framework" which represents the DT in a hierarchy of clusters.

SC approach tends to transform HMC task into a set of binary classification problems and applies DT algorithm for each label in the hierarchy. This algorithm is not efficient because it should run the DT algorithm *n* times based on the number of labels. In addition, it predicts each label separately and thus it is not automatically ensuring the hierarchy constraint. HSC overcomes the last SC drawback by imposing the hierarchy constraint. It adopts SC method and run DT algorithm for each edge in the hierarchy. In contrast, the third approach which is HMC aims to learn the single classification model by employing one DT algorithm to predict all hierarchical labels at once.

The authors have considered the label hierarchy as a tree then they have extended it toward a DAG structure. The three proposed algorithms have been evaluated by using 24 functional genomics datasets. The empirical evaluation showed the superiority of HMC in both of tree and DAG hierarchy structures. It has achieved better predictive performance that outperforms HSC and SC algorithms.

IV. Multi-Label Evaluation Metrics

MLC models are evaluated in terms of performance and accuracy using metrics that are commonly used in the MLC field [41]. The evaluation of MLC models is unlike the evaluation of single-label classification models. Thus, according to [2] several multi-label evaluation metrics have been proposed which are classified into two main approaches: example-based (instance-based) and label-based metrics. The first approach is computed for each test instance and then averaged over all test instances. Whereas, the second approach is computed for each label and then averaged over all labels.

A. Example-based Metrics

The most common example-based metrics that are used to evaluate MLC models are described in the following. Suppose that: *m* refers to the total number of instances in the test dataset, *i* indicates an instance in the test dataset (where $1 \leq i \leq m$), *n* is the total number of labels, Z_i and Y_i refers to the predicted and actual labels, respectively.

- **Hamming loss** It calculates the average number of errors found in the instance-label pairs, averaged over all instances. The expression of this metric is shown in Eq. (3). Where the factor $\frac{1}{(n)}$ is used to obtain the normalized value in [0,1] and $\triangle$ defines the symmetric difference between predicted and actual labels.

$$Hamming\ loss = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n}\left|Z_i \triangle Y_i\right| \tag{3}$$

- **ML-accuracy.** It is known as multi-label accuracy or example-based accuracy. It computes the ratio of the labels predicted correctly over the total number of labels, as shown in Eq. (4).

$$ML - accuracy = \frac{1}{m}\sum_{i=1}^{m}\frac{|Zi \cap Yi|}{|Zi \cup Yi|} \tag{4}$$

- **Subset accuracy** [42]. It also called exact match ratio or classification accuracy. It is a very strict metric used to measure the ratio of predicted labels which exactly match their corresponding actual set of labels, as shown in Eq. (5). Where *I(true)* = 1 and *I(false)* = 0.

$$Subset\ accuracy = \frac{1}{m}\sum_{i=1}^{m} I(Zi = Yi) \tag{5}$$

- **Precision.** This metric giving us the ratio of labels correctly classified of the predicted labels. The expression of this metric is shown in Eq. (6).

$$Precision = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i|} \tag{6}$$

- **Recall.** This metric is computed as shown in Eq. (7), computes the ratio of correctly predicted labels of the actual labels.

$$Recall = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Y_i|} \tag{7}$$

- **F-Measure.** It is the harmonic mean between precision and recall. It is computed as shown in Eq. (8).

$$F - measure = \frac{1}{m}\sum_{i=1}^{m}\frac{2\,|Z_i \cap Y_i|}{|Z_i| + |Y_i|} \tag{8}$$

All example-based metrics described in this section indicate that the metric with the highest value has better performance, except Hamming loss metric, the lower value of Hamming loss indicates the better performance.

B. Label-based Metrics

The binary evaluation metrics (e.g., recall, precision, and F-measure) can be calculated for all labels based on two approaches of computing the average; either macro or micro averaged approaches. These metrics are widely used to measure the average for recall, precision, and F-measure. Let *B* a binary evaluation measure used to calculate these metrics, which computed based on the number of true positives (tp), false positives (fp), true negatives (tn), and false negatives (fn). The expressions of macro-averaged and micro-averaged metrics for *B*(tp, tn, fp, fn) are illustrated in Eq. (9) and Eq. (10), respectively.

$$B_{macro} = \frac{1}{n}\sum_{i=1}^{n} B(tp_i, fp_i, tn_i, fn_i) \tag{9}$$

$$B_{micro} = B\left(\sum_{i=1}^{n} tp_i, \sum_{i=1}^{n} fp_i, \sum_{i=1}^{n} tn_i, \sum_{i=1}^{n} fn_i\right) \tag{10}$$

V. Applications of MLC Methods in Arabic Text Classification

In this section we focus on Arabic language by presenting the relevant research studies which have been found in the literature that apply MLC methods in Arabic text classification.

A. Lexicon Approach

The study conducted in [13] has proposed an Arabic multi-label text classification model based on lexicons. The authors of this study aimed to classify multi-label Arabic dataset using a combination of lexicons. They have collected 4,720 Arabic articles with 35 labels from the BBC news website. The collected datasets were divided into training and test datasets using 70/30 split. Then, the training data were exploited to construct the lexicons for each label. They implemented the term frequency (TF) method that automatically counts the term frequency of each label.

After that, they have built 35 lexicons for the 35 labels, there are some common terms found in the lexicons of several labels and this support the MLC and return multi-labels for those terms. Then, the lexicons are used by the classifier to predict the labels for a new instance. The label prediction is done by matching the terms of each label stored in the lexicons with the term vector of a given instance and classifying them according to the term frequency. Then, just the first five labels with the greatest count values are predicted.

Finally, they have evaluated their classification model using Hamming loss, ML-accuracy, subset accuracy, and the execution time. Several experiments have been performed and the results showed that the lexicon-based model achieved better performance in term of ML-accuracy compared with the corpus-based approach using MEKA tool[1].

B. PT Methods based on Binary Classification

Regarding transformation to binary classification method, a new model for MLC has been developed in [4] based on BR method. The main contribution of this study was to solve MLC problem for Arabic dataset by designing BR method based on different set of single label classifiers including SVM, NB, and k-NN. These base classifiers have been employed with BR, and evaluated using four approaches which are: the set of SVM classifiers, NB classifiers, k-NN classifiers, and various set of classifiers.

Besides, three feature selection methods have been investigated namely Chi-square, odd ratio, and mutual information to improve Arabic MLC performance. Obviously, this study aimed to incorporate feature selection method and classification algorithm efficiently to obtain a more accurate MLC task.

The developed model was trained using the standard corpus which is collected in [33], it contains 10,000 Arabic articles written in “modern standard Arabic language (MSA)", where those articles are assigned to five labels: Sports, Economy, Arts, Science, and Politics. In addition, three multi-label evaluation metrics were used to evaluate the given Arabic datasets which are average recall, average precision, and average F-measure.

[1] http://waikato.github.io/meka/

After the evaluation process, the results showed that using BR which consists of various sets of machine learning classifiers (SVM, NB, k-NN) obtained the best results compared with the other evaluated methods. Moreover, the results showed the important effect of the feature selection method on the classification task. On the other hand, the main challenges faced by this study were the complex morphology of Arabic language, and the lack of a well-annotated publicly available Arabic datasets that covers more labels.

C. PT Methods based on Multi-Class Classification

The study conducted in [33] aimed to handle the MLC problem of Arabic language based on transformation to multi-class classification approach using a set of single-label machine learning classifiers. The authors aimed to transform the MLC problem of Arabic data into several single-label classification problems by using MEKA tool to perform PT methods which are: LP, BR, and Ranking and Threshold-based method (RT). The main idea of RT is that, after transforming MLC problem into a set of single-label classification problems, it assigns each label to a copy of the instance. Then, the label that has a value greater than a particular threshold remains with this instance, otherwise it is discarded.

The standard single-label machine algorithms have been applied as a base classifier to predict each resulted single-label data are: SVM, k-NN, NB, and DT. Then, they train the problem transformation methods (LP, BR, and RT) using their own multi-label Arabic dataset. They faced the same problem of the previous studies which have been conducted on MLC for Arabic language, which is the lack of publicly available multi-label Arabic datasets. Therefore, they have built their standard corpus from BBC news website, which contains about 10,000 Arabic articles, where those articles assigned to five labels: Sports, Arts, Economy, Politics, and Science. In addition, they run several experiments on five versions of their datasets with different sizes ranging from 3,000 to 7,000 articles, to study the effect of the scaling up of the datasets on the considered methods.

Three multi-label metrics were used during the evaluation phase which are: Hamming loss, ML-accuracy, and subset accuracy. Consequently, after performing the evaluation process on the given datasets, the results showed that using SVM as a base classifier with LP method achieved the best result with 71% ML-accuracy. However, the ML-accuracy of some classifiers changed when scaling up the datasets, more experiments were needed to justify this abnormal behavior in this study.

D. Algorithm Adaptation Technique

The study conducted in [34] focused on using algorithm adaptation technique to address MLC task on the Arabic news articles. Three multi-label classifiers were considered which are, Random Forest (RF) classifier, DT classifier, and the k-NN classifier (where $k = 5$ (5NN)). They chose these classifiers due to their ability to support MLC in a natural way.

On the other hand, one of the main challenges faced by this study, as stated in [33], is the lack of publicly available and well-annotated multi-label Arabic dataset. Therefore, they

have built their own crawler to collect the dataset which is used to train the considered multi-label classifiers.

The dataset consists of 10,997 articles obtained from the CNN Arabic news website and written in MSA. The articles have multiple labels e.g., Economics, Sports, Middle East, World, Science & Technology, and Miscellaneous. The considered classifiers were evaluated using several MLC metrics (Hamming loss, accuracy, micro-average F-measure, micro-average recall, and micro-average precision). After conducting some experiments, the evaluation results revealed that the DT classifier achieved better performance compared to the RF and 5NN classifiers.

Another recent study has been conducted in [5] addresses MLC in the Arabic language. The study investigated the MLC problem by conducting a vast evaluation comparison on the most common MLC algorithms including PT methods. It includes transformation to binary classification approaches such as BR, CC, and Calibrated Ranking by Pairwise Comparison (CRPC) [43]. Also, it covers PT methods which are based on transformation to multi-class classification such as LP. They trained these techniques using three base classifiers (SVM, kNN, and RF). Besides, four algorithms based on adaptation technique are also evaluated which are: ML-kNN, RFBoost [44], Instance-Based Learning by Logistic Regression Multi-label (IBLRML) [45], and Binary Relevance kNN (BRkNN) [46]. The algorithms were evaluated using the introduced multi-label Arabic dataset named “RTAnews", which is a new benchmark dataset consists of 23,837 Arabic news articles distributed over 40 multiple labels.

The comparison has been done to study the effectiveness of the new dataset on the MLC task. The experiments were evaluated using MLC evaluation metrics (macro-F-measure, macro-precision, macro-recall, micro-F-measure, micro-precision, micro-recall). The results showed that both RFBoost, and LP (with SVM base classifier) outperformed other MLC algorithms. Also, the performance of algorithm adaptation algorithms is faster than PT algorithms except for the LP algorithm.

E. Hierarchical Multi-Label Classification

To address HMC problem in Arabic language, a hierarchical classification system has been constructed in [35] based on HOMER algorithm, to classify the received Islamic requests (fatwa) and send them to the most appropriate Muslim scholar. Since, each fatwa request has multi-labels organized in a tree hierarchy. Thus, for each incoming fatwa request, the proposed system can automatically route them to one or more appropriate labels associated with the relevant Muslim scholar, as shown in Fig. 10.

Before any classification task, the Arabic text (Islamic fatwa dataset) should go through several pre-processing and feature-selection phases to make the text appropriate for a classification task. Several methods can be used to do these tasks, for example, the authors found that using light stemmer (light 10) in the pre-processing phase and Chi-square method in the feature-selection phase are suitable for their data. They conducted the experiments and trained the HOMER classifier using the processed dataset that includes about 15,539 text instances where the instance is assigned to multiple labels

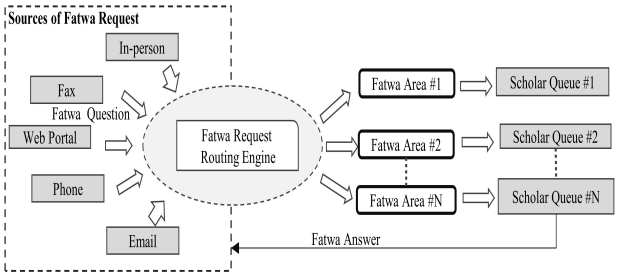

Fig. 10. Architecture of Fatwa Request System

organized in a tree-structured hierarchy consisting of 310 nodes (labels) in total.

They focused on comparing HOMER classifier against BR classifier (using NB base classifier). Several evaluation metrics were considered to evaluate the classification performance of the hierarchical multi-label classifier, which are Hamming loss, micro-averaged precision, micro-averaged F-measure, and micro-averaged recall. Finally, the results showed that using HOMER algorithm in the hierarchical classification of fatwa requests achieved more effective and efficient predictive performance compared to the BR classifier which relies on classifying each label independently.

Overall, we found that there are few researches conducted to tackle MLC problem in Arabic language compared to English [5]. The few research studies that addressed MLC problem for Arabic text in the literature are the aforementioned studies in this section [33], [5], [13], [34], [4], [35]. According to these studies we observed that feature selection method [4], dataset size [33], and pre-processing phase could have an important effect on the predictive performance of MLC model.

Table IV presents a summary of the relevant studies that applied MLC methods on the Arabic language. It shows the type of MLC methods and approaches that have been investigated in each application. Besides, it identifies the dataset size and the dataset source for each study.

The studies in this table are organized increasingly according to the publishing year. The main challenge has been mentioned by the authors of those studies which cause the limitation of the conducted studies on the MLC field in the Arabic context, is the lack of large enough and publicly available multi-label Arabic datasets. Besides, the vast vocabulary and complex morphology of the Arabic language. As a result, the study conducted in [5] addressed the lack of datasets by introducing a new benchmark dataset called “RTAnews". RTAnews dataset is a multi-label Arabic dataset which publicly available online on different formats including the format compatible with MEKA and MULAN multi-label learning tools. The RTAnews dataset obtained from “Russia Today Arabic news portal", and consists of 23,837 Arabic news articles distributed over 40 labels. Thus, this study led to promote evaluating of MLC methods and other supervised learning algorithms, by using the RTAnews dataset which is publicly available on its web page [2].

[2] https://data.mendeley.com/datasets/322pzsdxwy/1

TABLE IV. SUMMARY OF THE RELEVANT STUDIES THAT APPLIED MLC METHODS ON THE ARABIC LANGUAGE

Reference	Year	Multi-label classification			Dataset size	Dataset source
		Lexicon approach	Machine learning approach			
			Flat classification	Hierarchical classification		
[35]	2015			√	15,539	Provided by the Egyptian Dar al-Ifta.
[33]	2015		√		10,000	Collected from BBC news website.
[4]	2016		√		10,000	Taken from the study conducted in [33].
[13]	2016	√			4,720	Collected from BBC news website.
[34]	2016		√		10,997	Obtained from the CNN Arabic news website.
[5]	2019		√		23,837	Collected from Russia Today Arabic news portal website.

Furthermore, according to Table IV it is noticeable that there are about four studies conducted on the flat multi-label classification using machine learning algorithms, and only one study conducted to address the hierarchical classification problem in Arabic. However, since hierarchical classification is becoming a very significant topic in the recent days and applied in different domains including text classification. Thus, there is a need to be well-investigated by the future studies.

VI. EXPERIMENTAL COMPARISON OF MLC METHODS FOR ARABIC TEXT

In the previous sections, we presented an extensive review of the existing MLC methods and applications on Arabic text classification. It is interesting to evaluate the predictive performance of the state-of-the-art MLC methods introduced in this paper. The most common MLC methods have been evaluated involve four PT methods (BR, CC, LP, PS), one method based on algorithm adaptation technique (ML-kNN), and one hierarchical classification algorithm (HOMER).

The experiments performed using “RTAnews" dataset which was introduced in [5], and retrieved from its online web page. RTAnews is an imbalance dataset consists of 23,837 text instances distributed over 40 labels. It is available with a different set of features, 2000 feature is the selected dimension of the feature set used in the experiments. The summarization of the multi-label statistics of RTAnews dataset is illustrated in Table V.

TABLE V. RTANEWS DATASET STATISTICS

Number of instances	23,837
Total number of labels	40
Number of features	2000
Number of attributes	2040
Number of label-sets	442

MULAN[3] is an open-source java library for multi-label learning used to conduct the experiments. The classification performance of MLC methods are measured concerning the three example-based metrics (Hamming loss, ML-accuracy, subset accuracy) and three label-based metrics (micro-averaged precision, micro-averaged recall, micro-averaged F-measure) which presented previously in Section IV.

To make the evaluation more reliable 10-folds cross-validation were applied. Two experiments have been performed. The first experiment aims to investigate the performance of MLC methods (BR, CC, LP, PS, ML-kNN, and HOMER) on the predictive performance of the model. Noting that, all PT methods (BR, CC, LP, PS) evaluated using NB as a base classifier. HOMER algorithm was run using its default classifiers (BR and NB) and ML-kNN was evaluated by setting the number of k to 10. The second experiment aims to investigate the effect of three single-label base classifiers (NB, SVM, J48) on the performance of PT methods (BR, CC, LP, PS).

The results of the first experiment are presented in Table VI. We notice that the classification performance of all MLC methods in terms of Hamming loss, subset accuracy, micro-averaged precision, micro-averaged F-Measure, can be ranked according to the best performance achieved in the following order: ML-kNN, LP, PS, HOMER, CC, and BR. Whereas, concerning ML-accuracy, LP yielded the best results with 0.6219. Meanwhile, for micro-averaged recall, the best result was 0.8552 which obtained by the BR method.

Table VII presents the results of the second experiment. It is noticeable that the classification performance of PT methods might be affected by the single-label base classifier. Whenever the single-label base classifier changes, the performance of PT methods (BR, CC, LP, and PS) also change. Accordingly, we observe that, the SVM base classifier yielded the best results with all PT methods in terms of all evaluation metrics except micro-averaged recall. Whereas, concerning the micro-averaged recall, NB classifier with BR and CC methods obtained the best results. Note that in Table VII we used the abbreviation (e.g., BR-NB) to denote the multi-label classifier and the implemented base classifier.

Also, it is worth mentioning that the base classifier J48 obtained the second-best results with most of PT methods. Meanwhile, NB obtained the lowest results. Overall, we conclude that there are several factors affect the classification performance of MLC methods. For instance, PT methods might be affected by the base classifier. In contrast, algorithm adaptation techniques such as ML-kNN might be affected by the used parameters e.g., k. Whereas, HOMER algorithm performance depends on the multi-label classifier, the base classifier, and the clustering algorithm [36].

VII. CONCLUSION

Multi-label classification is an important research field and it has been increasingly required by many applications in various domains including text classification. In this paper, an

[3] http://mulan.sourceforge.net/

TABLE VI. EXPERIMENTAL COMPARISON AMONG MLC METHODS

MLC method	Hamming loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure
BR	0.0549±0.0008	0.4293±0.0065	0.1948±0.0085	0.3288±0.0055	**0.8552±0.0089**	0.4750±0.0065
CC	0.0546±0.0008	0.4295±0.0065	0.1948±0.0085	0.3300±0.0055	0.8551±0.0088	0.4762±0.0064
LP	0.0211±0.0006	**0.6219±0.0113**	0.5283±0.0119	0.6344±0.0133	0.6476±0.0088	0.6409±0.0103
PS	0.0213±0.0007	0.6187±0.0110	0.5237±0.0119	0.6288±0.0139	0.6475±0.0090	0.6380±0.0105
ML-kNN	**0.0156±0.0004**	0.6123±0.0081	**0.5480±0.0091**	**0.8076±0.0052**	0.6075±0.0100	**0.6933±0.0074**
HOMER	0.0365±0.0008	0.5103±0.0075	0.3107±0.0089	0.4294±0.0078	0.7799±0.0059	0.5538±0.0062

TABLE VII. EXPERIMENTAL COMPARISON AMONG PT METHODS WITH DIFFERENT SINGLE-LABEL BASE CLASSIFIERS

PT method	Hamming loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure
BR method based on different base classifiers						
BR-NB	0.0549±0.0008	0.4293±0.0065	0.1948±0.0085	0.3288±0.0055	**0.8552±0.0089**	0.4750±0.0065
BR-J48	0.0178±0.0003	0.6076±0.0070	0.5078±0.0068	0.7125±0.0048	0.6502±0.0110	0.6799±0.0063
BR-SVM	**0.0163±0.0005**	**0.6632±0.0066**	**0.5561±0.0109**	**0.7269±0.0095**	0.7048±0.0058	**0.7157±0.0063**
CC method based on different base classifiers						
CC-NB	0.0546±0.0008	0.4295±0.0065	0.1948±0.0085	0.3300±0.0055	**0.8551±0.0088**	0.4762±0.0064
CC-J48	0.0177±0.0003	0.6260±0.0060	0.5334±0.0068	0.7117±0.0057	0.6549±0.0092	0.6820±0.0053
CC-SVM	**0.0164±0.0005**	**0.6841±0.0089**	**0.5845±0.0124**	**0.7212±0.0091**	0.7096±0.0091	**0.7153±0.0078**
LP method based on different base classifiers						
LP-NB	0.0211±0.0006	0.6219±0.0113	0.5283±0.0119	0.6344±0.0133	0.6476±0.0088	0.6409±0.0103
LP-J48	0.0199±0.0006	0.6429±0.0084	0.5730±0.0091	0.6648±0.0107	0.6361±0.0092	0.6501±0.0093
LP-SVM	**0.0141±0.0004**	**0.7380±0.0075**	**0.6615±0.0117**	**0.7715±0.0052**	**0.7295±0.0092**	**0.7499±0.0068**
PS method based on different base classifiers						
PS-NB	0.0213±0.0007	0.6187±0.0110	0.5237±0.0119	0.6288±0.0139	0.6475±0.0090	0.6380±0.0105
PS-J48	0.0198±0.0006	0.6427±0.0092	0.5725±0.0121	0.6665±0.0086	0.6355±0.0087	0.6506±0.0084
PS-SVM	**0.0142±0.0005**	**0.7369±0.0083**	**0.6605±0.0104**	**0.7695±0.0064**	**0.7276±0.0107**	**0.7480±0.0080**

extensive review has been conducted on the MLC methods by illustrating the concept of each method and discussing their advantages and limitations. In addition, it organized the sparse state-of-the-art MLC methods by providing a taxonomy which summarized different MLC methods and techniques that can deal with MLC problem. Besides, a set of common multi-label evaluation metrics used to evaluate MLC models have been presented. Furthermore, we focused on the Arabic language by discussing the existing applications of MLC methods in the Arabic context. It is noticeable that only a few research studies addressed the problem of MLC for the Arabic language which mainly focused on flat classification and neglected the hierarchical structure. We found that the main challenges faced by the those studies are the lack of large and publicly available multi-label Arabic datasets, and also the vast vocabulary and complex morphology of the Arabic language. Finally, the survey provided an experimental comparisons of different MLC methods in the Arabic context and points out some baseline results. However, further research is still needed to improve the multi-label classification task for the Arabic language, especially the hierarchical classification task. Additionally, there is a need to give more attention for preparation multi-label Arabic datasets in an appropriate representation for a multi-label classification task.

REFERENCES

[1] M. Syiam, M. Tolba, Z. Fayed, M. Abdel-Wahab, S. Ghoniemy, and M. Habib, "An intelligent system for arabic text categorization," *International journal of cooperative information systems*, vol. 6, no. 1, pp. 1–19, 1 2006.

[2] E. Gibaja and S. Ventura, "A tutorial on multilabel learning," *ACM Comput. Surv.*, vol. 47, no. 3, pp. 52:1–52:38, Apr. 2015. [Online]. Available: http://doi.acm.org/10.1145/2716262

[3] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining (IJDWM)*, vol. 3, no. 3, pp. 1–13, 2007.

[4] A. Y. Taha and S. Tiun, "Binary relevance (br) method classifier of multi-label classification for arabic text." *Journal of Theoretical & Applied Information Technology*, vol. 84, no. 3, 2016.

[5] B. Al-Salemi, M. Ayob, G. Kendall, and S. A. M. Noah, "Multi-label arabic text categorization: A benchmark and baseline comparison of multi-label learning algorithms," *Information Processing & Management*, vol. 56, no. 1, pp. 212–227, 2019.

[6] H. Mubarak and K. Darwish, "Using twitter to collect a multi-dialectal corpus of arabic," in *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, 2014, pp. 1–7.

[7] T. Eldos, "Arabic text data mining: a root-based hierarchical indexing model," *International Journal of Modelling and Simulation*, vol. 23, no. 3, pp. 158–166, 2003.

[8] Y. Ahmed, J. Xiang, D. Zhao, M. A. A. Al-qaness, M. Elsayed abd el aziz, and D. Abdelghani, "A study of the effects of stemming strategies on arabic document classification," *IEEE Access*, vol. PP, pp. 1–1, 03 2019.

[9] M. El Kourdi, A. Bensaid, and T.-e. Rachidi, "Automatic arabic document categorization based on the naïve bayes algorithm," in *proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages*. Association for Computational Linguistics, 2004, pp. 51–58.

[10] R. Al-Shalabi and R. Obeidat, "Improving knn arabic text classification with n-grams based document indexing," in *Proceedings of the Sixth*

International Conference on Informatics and Systems, Cairo, Egypt, 2008, pp. 108–112.

[11] G. Hassan, "Categorization arabic text using svm and knn algorithms," *International Journal of Engineering and Technology*, pp. 906–909, 01 2018.

[12] J. Eisenstein, "Unsupervised learning for lexicon-based classification," *CoRR*, vol. abs/1611.06933, 2016. [Online]. Available: http://arxiv.org/abs/1611.06933

[13] I. Hmeidi, M. Al-Ayyoub, N. A. Mahyoub, and M. A. Shehab, "A lexicon based approach for classifying arabic multi-labeled text," *International Journal of Web Information Systems*, vol. 12, no. 4, pp. 504–532, 2016.

[14] R. M. Duwairi and R. Al-Zubaidi, "A hierarchical k-nn classifier for textual data." *Int. Arab J. Inf. Technol.*, vol. 8, no. 3, pp. 251–259, 2011.

[15] A. Aldrees and A. Chikh, "Comparative evaluation of four multi-label classification algorithms in classifying learning objects," *Computer Applications in Engineering Education*, vol. 24, no. 4, pp. 651–660, 2016.

[16] O. Luaces, J. Díez, J. Barranquero, J. J. del Coz, and A. Bahamonde, "Binary relevance efficacy for multilabel classification," *Progress in Artificial Intelligence*, vol. 1, no. 4, pp. 303–313, 2012.

[17] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine learning*, vol. 85, no. 3, p. 333, 2011.

[18] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE transactions on knowledge and data engineering*, vol. 26, no. 8, pp. 1819–1837, 2014.

[19] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, no. 16-17, pp. 1897–1916, 2008.

[20] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *European conference on machine learning*. Springer, 2007, pp. 406–417.

[21] J. Read, B. Pfahringer, and G. Holmes, "Multi-label classification using ensembles of pruned sets," in *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 995–1000.

[22] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1079–1089, 2010.

[23] D. Stojanova, M. Ceci, D. Malerba, and S. Džeroski, "Learning hierarchical multi-label classification trees from network data," in *International Conference on Discovery Science*. Springer, 2013, pp. 233–248.

[24] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, ser. NIPS'01. Cambridge, MA, USA: MIT Press, 2001, pp. 681–687. [Online]. Available: http://dl.acm.org/citation.cfm?id=2980539.2980628

[25] M.-L. Zhang and Z.-H. Zhou, "Multilabel neural networks with applications to functional genomics and text categorization," *IEEE transactions on Knowledge and Data Engineering*, vol. 18, no. 10, pp. 1338–1351, 2006.

[26] ——, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[27] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 2001, pp. 42–53.

[28] R. Al-Otaibi, M. Kull, and P. Flach, "Declaratively capturing local label correlations with multi-label trees," in *Proceedings of the 22nd Biennial European Conference on Artificial Intelligence (ECAI2016), Including Prestigious Applications of Intelligent Systems (PAIS-2016).*, ser. Frontiers in Artificial Intelligence and Applications, G. A. Kaminka, M. Fox, P. Bouquet, E. Hüllermeier, V. Dignum, F. Dignum, and F. van Harmelen, Eds., vol. 285. IOS press, 29 August- 2 September 2016, pp. 1467 – 1475. [Online]. Available: http://ebooks.iospress.com/volumearticle/44904

[29] F. Brucker, F. Benites, and E. Sapozhnikova, "Multi-label classification and extracting predicted class hierarchies," *Pattern Recognition*, vol. 44, no. 3, pp. 724–738, 2011.

[30] G. Madjarov, V. Vidulin, I. Dimitrovski, and D. Kocev, "Web genre classification via hierarchical multi-label classification," in *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 2015, pp. 9–17.

[31] C. N. Silla and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining and Knowledge Discovery*, vol. 22, no. 1-2, pp. 31–72, 2011.

[32] W. Bi and J. T. Kwok, "Multi-label classification on tree-and dag-structured hierarchies," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 17–24.

[33] N. A. Ahmed, M. A. Shehab, M. Al-Ayyoub, and I. Hmeidi, "Scalable multi-label arabic text classification," in *Information and Communication Systems (ICICS), 2015 6th International Conference on*. IEEE, 2015, pp. 212–217.

[34] M. A. Shehab, O. Badarneh, M. Al-Ayyoub, and Y. Jararweh, "A supervised approach for multi-label classification of arabic news articles," in *Computer Science and Information Technology (CSIT), 2016 7th International Conference on*. IEEE, 2016, pp. 1–6.

[35] R. A. Zayed, M. F. A. Hady, and H. Hefny, "Islamic fatwa request routing via hierarchical multi-label arabic text categorization," in *Arabic Computational Linguistics (ACLing), 2015 First International Conference on*. IEEE, 2015, pp. 145–151.

[36] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Effective and efficient multilabel classification in domains with large number of labels," in *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*, vol. 21. sn, 2008, pp. 53–59.

[37] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Machine learning*, vol. 73, no. 2, p. 185, 2008.

[38] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni, "Incremental algorithms for hierarchical classification," *J. Mach. Learn. Res.*, vol. 7, pp. 31–54, Dec. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248549

[39] Y. Chen, M. M. Crawford, and J. Ghosh, "Integrating support vector machines in a hierarchical output space decomposition framework," in *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, vol. 2. IEEE, 2004, pp. 949–952.

[40] L. Zhang, S. Shah, and I. Kakadiaris, "Hierarchical multi-label classification using fully associative ensemble learning," *Pattern Recognition*, vol. 70, pp. 89–103, 2017.

[41] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data mining and knowledge discovery handbook*. Springer, 2009, pp. 667–685.

[42] S. Zhu, X. Ji, W. Xu, and Y. Gong, "Multi-labelled classification using maximum entropy method," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2005, pp. 274–281.

[43] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine learning*, vol. 73, no. 2, pp. 133–153, 2008.

[44] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, vol. 37, no. 3, pp. 297–336, Dec 1999.

[45] W. Cheng and E. Hüllermeier, "Combining instance-based learning and logistic regression for multilabel classification," *Machine Learning*, vol. 76, no. 2, pp. 211–225, Sep 2009. [Online]. Available: https://doi.org/10.1007/s10994-009-5127-5

[46] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Artificial Intelligence: Theories, Models and Applications*, J. Darzentas, G. A. Vouros, S. Vosinakis, and A. Arnellos, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 401–406.