Egyptian Informatics Journal 22 (2021) 225–237

Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

HMATC: Hierarchical multi-label Arabic text classification model using machine learning

Nawal Aljedani *, Reem Alotaibi, Mounira Taileb

Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia

ARTICLE INFO

Article history:
Received 21 May 2020
Revised 4 August 2020
Accepted 29 August 2020
Available online 22 September 2020

Keywords:
Text classification
Multi-label classification
Hierarchical classification
Machine learning
Arabic natural language processing

ABSTRACT

Multi-label classification assigns multiple labels to each document concurrently. Many real-world classification problems tend to employ high-dimensional label spaces, which can be naturally structured in a hierarchy. In this type of problem, each instance may belong to multiple labels and labels are organized in a hierarchical structure. It presents a more complex problem than flat classification, given that the classification algorithm has to take into account hierarchical relationships between labels and be able to predict multiple labels for the same instance. Few studies have investigated multi-label text classification for the Arabic language. Most of these studies have focused mainly on flat classification and have neglected the hierarchical structure. Therefore, this paper explores the hierarchical multi-label classification in the context of the Arabic language. It proposes a hierarchical multi-label Arabic text classification (HMATC) model with a machine learning approach. The impact of feature selection methods and feature set dimensions on classification performance are also investigated. In addition, the Hierarchy Of Multilabel ClassifiER (HOMER) algorithm is optimized via examination of different sets of multi-label classifiers, clustering algorithms and different numbers of clusters to improve the hierarchical classification. Moreover, this study contributes to existing research by introducing a hierarchical multi-label Arabic dataset in an appropriate format for hierarchical classification and making it publicly available. The results reveal that the proposed model outperforms all models considered in the experiments in terms of the computational cost, which consumed less cost (2 h) compared with other evaluated models. In addition, it shows a significant improvement compared with the state-of-the-art model (Fatwa model) in terms of Hamming loss (0.004), hierarchical loss (1.723), multi-label accuracy (0.758), subset accuracy (0.292), micro-averaged precision (0.879), micro-averaged recall (0.828), and micro-averaged F-measure (0.853).

1. Introduction

The rapid growth of web pages, online storage providers, and social media networks has led to a significant increase in the number of available electronic text documents. According to the International Data Corporation, "the digital data on the internet will grow to 40,000 exabytes in 2020 from 130 exabytes in 2005" [1]. Effective management and classification of such data requires an accurate automatic text classification model. Thus, text classification or categorization (TC) remains an important field of research which attracts a significant attention. In general, TC can be formally defined as a supervised machine learning technique that automatically assigns a given instance to predefined labels based on its content [2]. A classification model is trained using training data, which includes a collection of instances and their corresponding labels (categories) [3].

Prior research has used two approaches for classification. On one hand, single-label classification, is the traditional classification, which assigns only one predefined label to each instance. Single-label classification can be either binary classification or multi-class classification. On the other hand, multi-label classification (MLC) assigns a set of predefined labels for each instance simultaneously [4]. Usually, it is inadequate to classify each instance under just one single label, because several labels could describe its content concurrently [5]. For example, a news article that is assigned to an 'education' label could also be assigned to several other labels simultaneously, such as 'social' and 'technology'.

* Corresponding author.
E-mail addresses: naljedani0026@stu.kau.edu.sa (N. Aljedani), ralotibi@kau.edu.sa (R. Alotaibi), mtaileb@kau.edu.sa (M. Taileb).

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

https://doi.org/10.1016/j.eij.2020.08.004

MLC is divided into two classification types: flat and hierarchical. In flat classification, a set of predefined labels are classified without considering the hierarchy of the relationship between the labels [6]. In contrast, in hierarchical multi-label classification (HMC) a single instance may have multiple labels concurrently, and these labels are structured in a hierarchy [7]. This type of classification, presents a more complex classification problem than flat classification, given that the classification algorithm has to take into account hierarchical relationships between labels and be able to predict multiple labels for the same instance.

HMC tasks have become important in many application, such as web pages classification, digital libraries, electronic books, patents, and newspaper articles. The use of machine learning to conduct HMC is a good way to facilitate document classification and retrieve, search, and request information easily and efficiently [8].

Several studies have conducted MLC for the English language. However, few studies have been conducted for the Arabic language, and research in this field has tended to focus on flat multi-label classification and has not considered HMC in depth. Arabic is the native language of 380 million people [9] and is one of the six official languages used by the United Nations [10]. It contains twenty-eight alphabet letters, with twenty-five consonants and three long vowels. It also has a vast vocabulary and complex morphology [11]. The significant use of Arabic language on electronic websites and social media networks, especially in recent years, has led to the requirement to develop an automatic text classification technique that can organize and categorize a large amount of electronic Arabic text documents efficiently.

Therefore, this paper addresses the problem of HMC in the context of the Arabic language by proposing a hierarchical multi-label Arabic text classification (HMATC) model that can deal with the hierarchical structure of Arabic text. The proposed model optimized the Hierarchy Of Multilabel classifiER (HOMER) algorithm to improve the hierarchical classification task.

In particular, the primary contributions of this paper can be summarized as follows:

- Proposes the HMATC model based on HOMER algorithm.
- Optimizes the essential parameters of the HOMER algorithm using different sets of multi-label classifiers, clustering algorithms and different numbers of clusters.
- Provides an insight into the impact of feature selection methods and feature set dimensions on the proposed model.
- Introduces multi-label Arabic dataset in an appropriate format for a hierarchical classification and making it publicly available online.
- For reproducibility purposes, the source code used in the HMATC implementation is accessible online on the GitHub page.[1]

The rest of this paper is organized as follows: Section 2 summarizes different MLC methods and discusses the relevant research studies that have been conducted on MLC for Arabic text. Section 3 introduces the HMATC model and describes the steps required for its development. Then, Section 4 discusses the experiment by describing the dataset statistics, the evaluated methods, and the experimental setting. Section 5 discusses the results. Finally, Section 6 presents the conclusion and future research directions.

2. Related work

An extensive review of multi-label text classification is presented in the following sections to give insight into the existing MLC techniques and the relevant research studies that have focused on Arabic text.

2.1. Multi-label classification methods

MLC can be divided into flat and hierarchical classification. Flat classification can be conducted using a problem transformation (PT) or algorithm adaptation technique. A PT technique simply aims to transform MLC problem into single-label problems, after which a traditional single-label classification algorithm is used to perform the classification task. An algorithm adaptation technique is concerned with adapting single-label classification algorithms to deal with the MLC problem directly. Examples of such techniques include a multi-label lazy learning (ML-kNN) algorithm [12] and a multi-label decision tree (ML-DT) algorithm [13].

The PT technique includes two general methods. The first of which involves transforming the multi-label problem into a set of binary classification problems. Examples of this method include binary relevance (BR) [14], classifier chains (CC) [15], and ranking by pairwise comparison (RPC) [16]. The second method converts the multi-label problem into a multi-class classification problem. Examples of this method include label powersets (LP) [17], pruned sets (PS) [18], and random *k*-labelsets (RA*k*EL) [19].

On the other hand, HMC is considered an extension or variant of MLC in which a hierarchical structure is considered on the multi-labels. The output of the classification algorithm for a given multi-label problem is a set of labels structured in a hierarchy [7]. The hierarchical multi-label problems are classified typically as a tree-hierarchy or a directed acyclic graph (DAG) [8]. Generally, the classification algorithms proposed for HMC are more capable of dealing with large sets of labels than those proposed for flat classification.

Several algorithms have been proposed for HMC, such as: hierarchical decision trees [20], the hierarchical *k*-nearest neighbors algorithm [6], incremental algorithms for hierarchical classification [21], hierarchical support vector machines [22], HMC using fully associative ensemble learning [23], the HOMER algorithm [24], basic categorization algorithm (BCA), and percentage difference categorization (PDC) algorithm [25].

2.2. Multi-label classification for Arabic text

Generally, MLC methods applied for English can be applied to Arabic language as well, but the difference basically is in the preprocessing phase. The following sections present a review of existing research studies that have been conducted regarding MLC in Arabic texts.

Ahmed et al. [26] conducted a study on binary and multi-class classification transformation methods using several multi-label classifiers. They transformed the MLC of Arabic data into single-label classification using MEKA[2] tool to implement LP, BR, and ranking and threshold-based (RT) approaches. The standard single-label machine learning algorithms applied as base classifiers were: a support vector machine (SVM), a *k*-nearest neighbor (*k*-NN) algorithm, a Naive Bayes (NB) method, and a decision tree (DT). The collected dataset on which the evaluation was performed consisted of 10,000 news articles classified into five labels (Sports, Arts, Economy, Politics, and Science). The results of the evaluation showed that using SVM as a base classifier with the LP method achieved the best ML-accuracy with 71%.

Taha and Tiun [5] developed a new MLC model using a BR method. The main objective of the study was to solve the MLC problem for Arabic datasets by employing a BR method based on

[1] https://github.com/NawalJed/HMATC.

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

different sets of single-label machine learning classifiers including SVM, NB, and k-NN. The evaluative experiments were performed using the same dataset collected in [26]. The experiment results showed that using a BR method consisting of various sets of single-label classifiers (SVM, NB, and k-NN) achieved the best results.

Shehab et al. [27] conducted a study that focused on Arabic news articles. Three multi-label classifiers were adapted to deal with MLC problems: random forest (RF), DT, and k-NN with k = 5 (5-NN). The researchers conducted experiments on a collected dataset of 10,997 news articles classified with multiple labels, such as Economics, Sports, World, Middle East, Science & Technology, and Miscellaneous. The evaluation results showed that the DT classifier achieved a better performance than the RF and 5-NN classifiers.

Hmeidi et al. [28] proposed a lexicon-based multi-label Arabic text classification model. They collected 4720 Arabic articles with thirty-five labels from the BBC news website. To classify the multi-label Arabic dataset, they employed a combination of lexicons of each label in a multi-label problem. Label prediction was achieved by matching the terms of each label stored in the lexicons with the term vector of a given instance and classifying them according to term frequency. Then the first five labels with the greatest count values were predicted. Finally, several experiments were conducted, and the results indicated that the performance of the lexicon-based model outperformed the corpus-based approach in terms of multi-label accuracy (ML-accuracy).

Al-Salemi et al. [1] conducted a study that sought to investigate MLC problems by conducting an in-depth comparison of the most common MLC algorithms used in the PT method, such as BR, CC, LP, and calibrated ranking by pairwise comparison (CRPC) [29]. These methods were trained using three base classifiers (SVM, kNN, and RF). Four algorithm adaptation techniques were also evaluated. These were: ML-kNN, RFBoost [30], binary relevance kNN (BRkNN) [31], and instance-based learning by multi-label logistic regression (IBLRML) [32]. The algorithms were evaluated using the RTAnews dataset[3] which is a multi-label Arabic dataset of 23,837 Arabic news articles distributed over forty categories. A comparison was performed to investigate the effectiveness of the introduced dataset (RTAnews) in MLC tasks. The experiment results showed that both RFBoost and LP with SVM outperformed the other MLC algorithms. Moreover, the algorithm adaptation methods performed faster than the other PT algorithms except for the LP method.

Elnagar et al. [33] conducted a study that introduces two new Arabic datasets for both single-label and multi-label text classification tasks. The datasets called SANAD (Single-label Arabic News Articles Dataset) and NADiA (multi-label News Articles Dataset in Arabic). Both datasets were collected from news sources and available online on Mendely.[4] Further, they conducted an extensive comparison of several deep learning models, to investigate the effectiveness of the introduced datasets on the Arabic text classification tasks. The results showed that all models achieved good results when evaluated using SANAD dataset, CGRU (convolutional gated recurrent unit) achieved the lowest accuracy of 91.18%, whereas HANGRU (hierarchical attention network-gated recurrent unit) achieved the highest performance of 96.94%. Concerning NADiA dataset, HANGRU has the best overall accuracy of 88.68%.

Zayed et al. [34] constructed a hierarchical multi-label classification model to address HMC problems in the Arabic language which used the HOMER algorithm to classify received Islamic requests (Fatwa) into the most appropriate hierarchical categories. They trained the HOMER algorithm using the default classifiers (BR and NB classifiers) and conducted experiments on the collected hierarchical multi-label Arabic dataset. The dataset before pre-processing contains about 100,000 text instances labelled with 830 labels. It was processed with standard pre-processing methods, including text cleaning, stop-word removal, and stemming. The words were stemmed using a Light10 stemmer, and the features were selected using a BR and Chi-square feature selection method. After processing the dataset, removing the empty instances (instances with no features or labels), and removing the labels that have less than 20 instances, the final version of the dataset included about 15,539 text instances assigned to 310 multiple labels organized as a tree-structured hierarchy. The authors focused on comparing the HOMER classifier with a flat BR-NB multi-label classifier. The results of the study showed that using the HOMER and its variations in the hierarchical classification of Fatwa requests achieved more effective predictive performance compared to the BR-NB classifier, which simply classified each label independently.

Table 1 presents a summary of the relevant research studies that have applied MLC methods to Arabic text using machine learning. It shows the type of MLC methods that have been investigated in each study. It also identifies the dataset size and the dataset source for each study.

The main challenges identified by the authors of the previous studies, which caused the lack of research into MLC in the Arabic context were: the dearth of large and publicly-available multi-label Arabic datasets and the vast vocabulary and complex morphology of the Arabic language.

Notably, most previous research has focused mainly on flat MLC methods, and to the best of our knowledge, only one study has investigated dealing with HMC problems in Arabic texts using machine learning. This study [34] employed a HOMER algorithm with its default classifiers (BR and NB classifiers). The study did not investigate empirically the impact of feature selection methods and feature set dimensions on the model. In addition, the dataset used in that research has not been published online.

Therefore, in our work, we focus on addressing HMC problems in the context of the Arabic language by proposing the HMATC model. We optimize the essential parameters of HOMER algorithm using different sets of multi-label classifiers, clustering algorithms and different numbers of clusters to improve the hierarchical classification. We further, provide an insight into the impact of feature selection methods and feature set dimensions on the proposed model. Moreover, we introduce a hierarchical multi-label Arabic dataset in an appropriate format for hierarchical classification and making it available online to the research community. The model proposed in [34], which is called the 'Fatwa model', is included in the evaluation comparison conducted in this paper as it was applied in the same domain.

3. Hierarchical Multi-label Arabic text classification (HMATC) model

The overall architecture of the proposed HMATC model is presented in Fig. 1, which illustrates the phases of the model's development.

The main purpose of this model is to classify Arabic text (Islamic Fatwa requests) automatically into multiple labels organized in an appropriate hierarchical structure. The model considers label dependencies by exploiting label correlations found in the training data. It incorporates the pre-processing technique and feature selection method and optimizes the essential parameters of the HOMER algorithm using different multi-label classifiers and clustering algorithms in order to obtain a competitive hierarchical multi-label classification model for the Arabic language.

[3] https://data.mendeley.com/datasets/322pzsdxwy/1.
[4] https://doi.org/10.17632/57zpx667y9.1;https://doi.org/10.17632/hhrb7phdyx.1.

Table 1
Summary of the relevant studies that applied MLC methods on the Arabic text using machine learning.

Reference	Year	MLC		Dataset size	Dataset source
		Flat	HMC		
[34]	2015		√	15,539	Provided by the Egyptian Dar al-Ifta.
[26]	2015	√		10,000	Collected from BBC news website.
[5]	2016	√		10,000	Taken from the study conducted in [26].
[27]	2016	√		10,997	Collected from the CNN Arabic news website.
[1]	2019	√		23,837	Obtained from "Russia Today Arabic news portal" website.

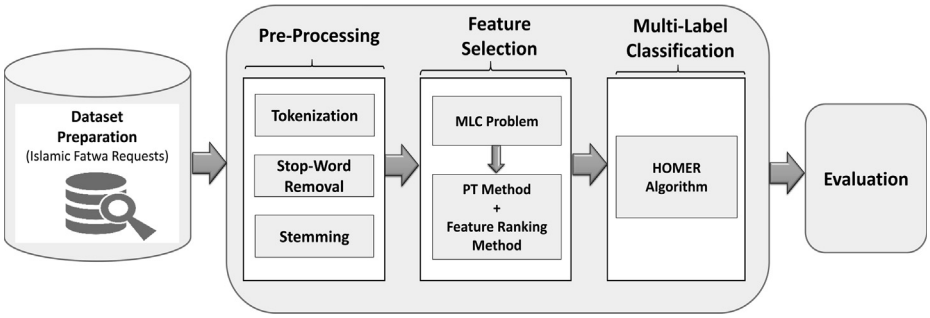

Fig. 1. Architecture of HMATC model.

The HMATC model was trained in this study using a hierarchical multi-label Arabic dataset. Then it was evaluated using evaluation metrics dedicated to multi-label classification. The following subsections describe the main phases of the model's development.

3.1. Dataset preparation phase

Since the objective of this study is to address HMC problems in the Arabic language, the experiments were conducted on a multi-label Arabic dataset, in which the labels are organized into a tree-structured hierarchy.

There is a lack of publicly-available multi-labelled Arabic datasets, especially those with hierarchical multi-labels. For this reason, the dataset used in this study was a raw hierarchical multi-label Arabic dataset taken from a study performed by Zayed et al. [34], which was applied in the same domain.

The dataset was a raw dataset related to the Islamic field and written in Modern Standard Arabic (MSA). It was stored in three main database tables. The first one was the Fatwa table, which contained about 100,000 text instances (Islamic Fatwa requests), as presented in Table 2. The second table was the Categories table, which contained 830 labels associated with their parent-ID that defined their hierarchical structure, as presented in Table 3. The third table was the Fatwa-Categories table, which contained a set of instances Ides' and labels Ides' and defined the associated labels for each instance. The third database table was used to assign each instance to its set of labels and the second table was used to trace the parent-ID of each label to define the labels' hierarchical structure.

One of the main challenges faced in this study was preparing the dataset by assigning each instance its own set of labels based on the third aforementioned database table (Fatwa-Categories table), and ensuring that the labels were in an appropriate format for hierarchical classification (they satisfied the hierarchical constraints). After removing the empty instances (instances with no features or labels), and assigning each text instance to its hierarchical multi-labels, the dataset was reduced to 26,484 text instances as presented in Table 4. The table shows that e.g., instance number one was labelled with two sets of hierarchical multi-labels. These were similar in terms of parent labels (General Category, Jurisprudence, Worship, Zakat – التصنيف العام، الفقه، العبادات، الزكاة) but differed in terms of leaf(ves) labels (Zakat Conditions – شروط الزكاة) and (The Rule of Zakat – حكم الزكاة).

After the labelling process, the hierarchical multi-label problems were identified based on their occurrences using a Boolean bag of words ({0,1} representation). To satisfy the hierarchical constraints, whenever the instance labelled by a label at a particular node, all parent labels of this node were represented by {1}, including the root node, otherwise, they were represented by {0}. The total number of labels identified was 830, which was a large number that would have increased computational cost. As a result, the label dimensionality was reduced by removing the labels that were rarely used in the dataset (e.g., labels associated with two instances or less), such that when the rare labels were the only labels related to the text instances, we deleted the associated text instances with these labels. In other cases, when there are other labels associated with text instances than these rare labels, we kept the text instances. The removed rare labels usually were the leaf(ves) labels, leaving 578 remaining labels that were used in the experiments. Fig. 2 presents an example of the tree-structured hierarchy of the labels in the Islamic Fatwa requests dataset.

3.2. Text pre-processing phase

To classify any text document using a machine learning algorithm, a raw text should be pre-processed. Pre-processing is a challenging task in the Arabic language but has the greatest impact on the classification performance of the model used due to the richness of the Arabic language, which contains more complex morphology than other languages [35–38]. In this phase, the raw text was prepared and transformed into a representation suitable for the application of a classification algorithm [11]. The vector space model is the most well-known approach used for document representation. According to this approach, each text instance is represented by a vector x that contains a list of distinct features (words) [39]. A term frequency-inverse document frequency (tf-idf) weighting scheme [40,41] was applied in this study to identify the features of the model, whereby each instance was represented using a vector of the terms' weights.

The following procedures were applied to the dataset in this study in the pre-processing phase [42,43]:

Table 2
Examples of Fatwa requests (Questions) from database table of the Islamic Fatwa requests dataset, where Instance # represents number of instance and ID represents the primary key of this table in the database.

Instance #	ID	Date	Question
1	4	07/30/1930	اطلعنا على الطلب المقدم من السيد معجب.... هل في هذا المبلغ المدخرزكاة أم لا؟ ⟨p⟩⟨/p⟩⟨/p⟩ We read the question from Mr.Muajab....is there a zakat for this saved amount? ⟨p⟩⟨/p⟩⟨/p⟩
2	5	07/30/1930	...يطلب السائل الإفادة عما إذا كانت تصوم هذه الأيام رغم تأديتها الفدية أم لا؟ ⟨p⟩ ...The questioner is asking if she has to fast,despite giving the ransom? ⟨p⟩
3	6	06/18/1932	السائل يقول سائق اشترى سيارة بالتقسيط ... ويطلب السائل الإفادة عن حكم شراء السيارة؟ ⟨p⟩ The questioner says a driver bought a car in installments.....he asks about the ruling of buying a car? ⟨p⟩
...			
100,000	26910	02/23/1953	سأل عبدالله عبدالحميد قال: في سنة ١٩٥٢ توفي المرحوم محمد ... Abdullah Abdul Hamid asked: In 1952 Muhammad died...

Table 3
Example of categories (Labels) database table of the Islamic Fatwa requests dataset, where Label # represents number of label and ID represents the primary key of this table in the database.

Label #	ID	Name	ParentID	Category_order
1	3	التصنيف العام (General Category)	0	0
2	191	العقائد (Doctrines)	3	0
3	192	الإلهيات وما يتعلق بها (Divinities and related matters)	191	0
4	193	النبوات وما يتعلق بها (Prophecies and related matters)	191	1
5	194	السمعيات وما يتعلق بها (Audiovisual and related matters)	191	2
6	195	الفرق والملل والنحل وما يتعلق بها (Sects and creeds and related matters)	191	3
7	274	أصول الفقه وما يتعلق به (Fundamentals of Islamic Jurisprudence)	3	1
8	324	الفقه (Jurisprudence)	3	2
9	325	العبادات (Worship)	324	0
...				
830				

Table 4
Examples of text instances from the Islamic Fatwa requests dataset, labelled with their hierarchical multi-labels.

Instance #	Text instances	Hierarchical multi-label				
1	اطلعنا على الطلب المقدم من السيد معجب... هل في هذا المبلغ المدخرزكاة أم لا؟ (We read the question from Mr.Muajab...is there a zakat for this saved amount of money or not?)	التصنيف العام General Category	الفقه Jurisprudence	العبادات Worship	الزكاة Zakat	حكم الزكاة The Rule of Zakat
1	اطلعنا على الطلب المقدم من السيد معجب... هل في هذا المبلغ المدخرزكاة أم لا؟ (We read the question from Mr.Muajab...is there a zakat for this saved amount of money or not?)	التصنيف العام General Category	الفقه Jurisprudence	العبادات Worship	الزكاة Zakat	شروط الزكاة Zakat Conditions
2	...يطلب السائل الإفادة عما إذا كانت تصوم هذه الأيام رغم تأديتها الفدية أم لا؟ (...The questioner is asking if she has to fast,despite giving the ransom?)	التصنيف العام General Category	الفقه Jurisprudence	العبادات Worship	الصيام Fast	صحة الصيام Fasting Validity
2	...يطلب السائل الإفادة عما إذا كانت تصوم هذه الأيام رغم تأديتها الفدية أم لا؟ (...The questioner is asking if she has to fast,despite giving the ransom?)	التصنيف العام General Category	الفقه Jurisprudence	العبادات Worship	الصيام Fast	القضاء Fasting
3	السائل يقول سائق اشترى سيارة بالتقسيط ... ويطلب السائل الإفادة عن حكم شراء السيارة؟ (The questioner says a driver bought a car in installments.....he asks about the ruling of buying a car?)	التصنيف العام General Category	الفقه Jurisprudence	المعاملات Transactions	البيع Sale	
....						
26484						

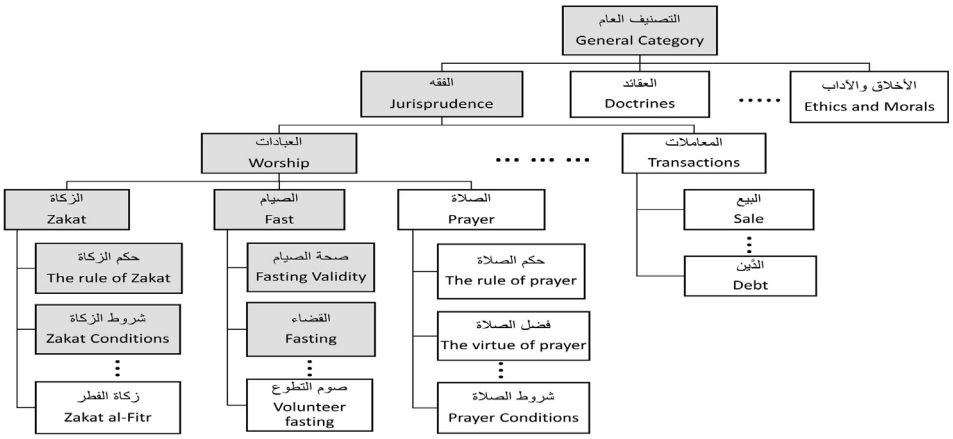

Fig. 2. Example of a hierarchical structure found in the labels of the Islamic Fatwa requests dataset.

- **Tokenization.** Also called text segmentation, tokenization aims to split a text into a set of features, namely tokens, based on a specific delimiter such as commas, white space, periods, etc.,
- **Stop-word removal.** Stop-word-removal aims to remove insignificant words. Stop-words are words which are frequently encountered in the text and carry no useful information and therefore can affect classification performance. These include prepositions, conjunctions, pronouns, and others.
- **Stemming.** Stemming aims to stem or return the derived features to their roots or stems. Transforming semantically similar features to their root form reduces the feature space, decreases the morphological variance of words, and improves the classification performance of the model [44,45].

In the current study, the researchers pre-processed the dataset and cleaned the raw text instances by removing numbers, non-Arabic letters, punctuation, and special characters ($, %, @, &). After text cleaning, the three steps described above were applied. First, tokenization was applied using space as the text delimiter. Then, a stop-words list was prepared and the identified words were removed from all instances. These included preposition (from, to, about, in – من، إلى، عن، في) and conjunctions (but, or, even, then – لكن، أو، حتى، ثم), although stop-words with a significant meaning in the Islamic context, such as Ramadan, during, before, and after were retained. In addition, Arabic human names (such as Khaled, Ahmed, Sara, Amal – خالد، أحمد، سارة، أمل) were also removed, since they were repeated frequently in the dataset and carried no useful information, which may have affected classification performance. Finally, stemming was carried out using a Snowball stemmer. Snowball is a small text processing language used to create the stemming algorithm, it supports many languages including the Arabic language. It aims to return words to their stems by stripping off common prefixes (e.g., سا، ب، ك، ل، بال، وال، فال، لل، ال) and suffixes (e.g., هم، هن، و، ي، هما، كما، ها، نا)

The number of features that remained after the pre-processing phase was 11,000 features. After completion of the dataset preparation, implementation of the pre-processing phase, and removal of unlabeled instances and rare labels, the final version of the dataset contained 26,470 text instances labelled with 578 hierarchical multi-labels in total.

Table 5 presents the dataset as a vector space model, whereby each instance is represented as a vector x containing a list of distinct features represented using a tf-idf weighting scheme.

The dataset was imbalanced, which means that the number of labels associated with the instances was not equally distributed. In general, the high-level labels in the root and parent nodes were usually associated with a high number of instances, whereas the low-level labels in the leaf(ves) nodes were associated with a low number of instances.

3.3. Feature selection phase

The pre-processing phase produced a set of stemmed features. However, the total number of features and the number of unnecessary features remained high, which would have affected classification performance by increasing computational complexity. Therefore, an additional feature selection method was needed to reduce the features number by selecting a subset of the most relevant and highest-ranking features. Using such a method effectively reduces the dataset dimensionality by removing irrelevant or redundant features without decreasing the classification performance. Hence, it can improve the learning process and reduce computational complexity [43].

However, unlike single-label data, which can directly use traditional feature ranking methods (such as Chi-square (χ^2) [46], information gain (IG) [47], gain ratio (GR), and relieF (RF)) to score features according to their relevance to the labels and then select the most relevant features, the standard method of feature selection in multi-label data, as illustrated by Spolaor et al. [48], is based on applying a PT classifier (e.g., LP, BR, CC) to transform MLC problems into single-label problems, after which a traditional feature ranking method can be employed to perform the feature selection.

One of the most common methods of multi-label text classification is using a BR method to determine the discriminative power of each feature concerning each label independently of the rest of the labels. After this, the computed scores are aggregated to obtain overall features' ranking [49].

In the MLC context, each combination of PT classifier and feature ranking method is considered a separate feature selection method. Thus, this study used an abbreviation of the PT classifier followed by a feature ranking method abbreviation separated by a hyphen symbol to denote the feature selection method in the multi-label classification context. For example; LP-IG referred to an independent feature selection method consisting of an LP classifier and IG feature ranking method, respectively.

The feature selection phase was examined by investigating different sets of feature selection methods and examining different sets of the high-ranking features, to obtain the best method and feature set dimensions for optimizing the model performance. The details of this phase are presented in the results and discussion section (see Section 5.1 and Section 5.3).

3.4. Classification phase

This paper investigates applying the HOMER algorithm in a domain with a large set of labels such as the Islamic context. The HOMER algorithm is an effective hierarchical multi-label classifier that uses a divide-and-conquer approach [24]. It can efficiently handle MLC problems with a large number of labels by constructing a tree-shaped hierarchy of simpler MLC problems. First, HOMER automatically organizes the large set of labels into a tree-shaped hierarchy. This is accomplished by applying a clustering algorithm which repetitively partitions the labelset into a number of nodes (clusters). Then, it employs a multi-label classifier (e.g., BR) for each node, where each classifier can handle a small number of labels instead of handling a large labelset.

There are three variants of the HOMER algorithm based on the clustering algorithm applied in the label distribution task. The first variant is HOMER-K, which uses a k-means clustering algorithm to distribute the labels into k number of clusters based on label similarity, without any constraint on the size of the cluster. The motivation here is investigating the benefits of even label distribution

Table 5
An example of a vector space model of the processed dataset.

Instance #	مبلغ Amount	مدخر Saved	زكا Zakat	فدية Ransom	صوم Fast	...	التصنيف العام General Category	الفقه Jurisprudence	العبادات Worship	الزكاة Zakat	حكم الزكاة The Rule of Zakat	شروط الزكاة Zakat Conditions	الصيام Fast	صحة الصيام Fasting Validity	القضاء Fasting	...	Label # 578
1	1.615	0.122	0.137	0.169	0		1	1	1	1	1	1	0	0	0		
2	0.101	0.105	0	0.925	1.017		1	1	1	0	0	0	1	1	1		
. . .																	
26470	0	1.615	0.122	0.103	1.004		1	1	1	1	1	0	0	0	0		

on top of the clustering. However, the authors of [24] proposed a new clustering algorithm known as a balanced *k*-means clustering algorithm, where the labels are evenly distributed into *k* balanced clusters. This defines the second and the default variant of HOMER called HOMER-B, which includes the additional constraint of constructing clusters with equal size. The third HOMER variant known as HOMER-R evenly but randomly clusters the labels into *k* clusters. The motivation of HOMER-R is investigating the benefits of clustering on top of the even label distribution.

Fig. 3 illustrates the tree hierarchy in the HOMER algorithm used for the classification of a simple MLC problem with nine labels (General Category, Jurisprudence, Worship, Zakat, Fast, Fasting Validity, Fasting, Zakat Conditions, The Rule of Zakat – التصنيف العام، الفقه، العبادات، الزكاة، الصيام، صحة الصيام، القضاء، شروط الزكاة، حكم الزكاة). These labels represent a small part of the hierarchical multi-label of the Islamic Fatwa requests dataset, as highlighted in Fig. 2. Each internal node includes the union of the meta-labels μ of its children, and the root node includes the labels of all nodes in the tree. A multi-label classifier *S* was employed for each node to predict the meta-labels of its children. The default structure of the HOMER algorithm was implemented by employing a BR-NB multi-label classifier and balanced *k*-means clustering algorithm. Where the abbreviation BR-NB refers to the BR multi-label classifier implemented using NB base classifier.

To predict the labels of an unseen instance *e* (let's suppose it is related to Zakat – الزكاة), HOMER starts from the root node and then follows a recursive approach to forward *e* to the most relevant multi-label classifier. In the example illustrated in Fig. 3, the multi-label classifier *S*1 at the root node will forward the instance *e* to the multi-label classifier *S*2 only if μ_2 is among the predicted labels of the *S*1 classifier. By following a recursive process, the final prediction will be a union of all predicted labels just higher than the corresponding leaf(ves) nodes. Regarding the instance *e*, the predicted labels of this instance will be General Category, Jurisprudence, Worship, Zakat, Zakat Conditions, The Rule of Zakat – التصنيف العام، الفقه، العبادات، الزكاة، شروط الزكاة، حكم الزكاة. This means that when the instance *e* is assigned to the labels at a particular node (e.g., Zakat Conditions, the Rule of Zakat – شروط الزكاة، حكم الزكاة), it should satisfy the hierarchical constraint and should therefore be assigned to all parent labels of these nodes including the root node (e.g., General Category, Jurisprudence, Worship, Zakat – التصنيف العام، الفقه، العبادات، الزكاة). If no label is predicted, the algorithm will return an empty set of labels.

Tsoumakas et al. [24] evaluated the HOMER algorithm with its variations by employing a BR-NB multi-label classifier. The results of the study showed that the HOMER which relied on similarity-based distribution and employed a BR-NB classifier reduced computational complexity and improved predictive performance.

The benefit of similarity-based distribution is that it keeps the labels belonging to each node as similar as possible. This means that only relevant meta-labels are predicted, and the rest of the sub-tree is inactivated during the testing phase, which reduces the total computational cost. Another advantage of even label distribution is that it avoids the class-imbalance problem by attempting to distribute the labels into a set of balanced clusters; thus, each multi-label classifier deals with a more balanced distribution of positive instances at each node.

One of the main contributions achieved by the HMATC model is the optimization of the essential parameters of the HOMER algorithm (multi-label classifier, clustering algorithm and numbers of clusters) using a different set of multi-label classifiers, and clustering algorithms along with different numbers of clusters to improve hierarchical classification outcomes (see Section 5.2 and Section 5.4).

3.5. Evaluation metrics

MLC models are evaluated using evaluation metrics that are commonly used in the MLC field [49]. Several multi-label evaluation metrics were proposed which divided into two main approaches: example-based (instance-based) and label-based metrics [3]. The first approach is measured for each test instance and then averaged over all test instances. Whereas the second approach is measured for each label and then averaged over all labels.

3.5.1. 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 $1 \leqslant i \leqslant m$), $L = \{\lambda_j : j = 1 \ldots q\}$ is the set of labels, where q is the total number of labels, Z_i and Y_i refer 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 as shown in Eq. (1). Where the factor $\frac{1}{(q)}$ is used to get the normalized value in [0,1] and $\triangle$ defines the symmetric difference between predicted and actual labels.

$$Hamming\ loss = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{q}|Z_i \triangle Y_i| \tag{1}$$

- **Hierarchical loss**. It takes the hierarchical structure of labels into consideration [21]. It follows a top-down approach to examine the predicted labels based on the existing label hierarchy. Thus, whenever the label is predicted wrong, the subtree rooted at that node is not considered in the loss calculation [49] as shown in Eq. (2). Where anc(λ) refers to the all ancestor nodes (subtree) of label λ.

$$H - Loss = \frac{1}{m}\sum_{i=1}^{m}|\{\lambda : \lambda \in Y_i \Delta Z_i, anc(\lambda) \cap (Y_i \Delta Z_i) = \phi\}| \tag{2}$$

- **ML-accuracy.** It is known as multi-label accuracy or a Jaccard index. It computes the ratio of the labels predicted correctly over the total number of labels, as shown in Eq. (3).

$$ML - accuracy = \frac{1}{m}\sum_{i=1}^{m}\frac{|Zi \cap Yi|}{|Zi \cup Yi|} \tag{3}$$

- **Subset accuracy**. It also called exact match ratio or classification accuracy [50]. 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. (4). Where *I (true)* = 1 and *I(false)* = 0.

$$Subset\ accuracy = \frac{1}{m}\sum_{i=1}^{m}I(Zi = Yi) \tag{4}$$

- **Precision.** This metric giving us the ratio of labels correctly classified of the predicted labels as shown in Eq. (5).

$$Precision = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i|} \tag{5}$$

- **Recall.** This metric is computed as shown in Eq. (6), computes the ratio of correctly predicted labels of the actual labels.

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

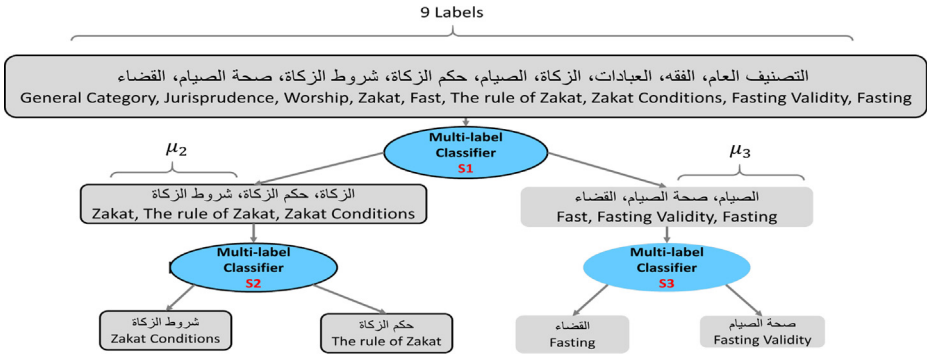

Fig. 3. Example of a classification task of MLC problem with nine labels of the used dataset using HOMER algorithm.

- **F-Measure.** It represents the harmonic mean between precision and recall. It is computed as shown in Eq. (7).

$$F-measure = \frac{1}{m}\sum_{i=1}^{m}\frac{2|Z_i \cap Y_i|}{|Z_i| + |Y_i|} \tag{7}$$

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

3.5.2. 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 $B(tp, tn, fp, fn)$ a binary evaluation measure of the label i computed based on the number of true positives (tp), true negatives (tn), false positives (fp), and false negatives (fn). The expressions of macro-averaged and micro-averaged metrics for $B(tp, tn, fp, fn)$ are illustrated in Eq. (8) and Eq. (9), respectively.

$$B_{macro} = \frac{1}{q}\sum_{i=1}^{q} B(tp_i, fp_i, tn_i, fn_i) \tag{8}$$

$$B_{micro} = B\left(\sum_{i=1}^{q} tp_i, \sum_{i=1}^{q} fp_i, \sum_{i=1}^{q} tn_i, \sum_{i=1}^{q} fn_i\right) \tag{9}$$

4. Experiment set-up

The following sub-sections describe the dataset statistics, methods, and experimental setting used in this study.

4.1. Dataset statistics

The size of the final version of the dataset was 26,470 labelled text instances, which included 11,000 features. The total number of labels was 578, and the total number of labelsets was 6107. Labelsets are the active set of labels associated with each instance and represented by {1} in a vector space model. There are other basic measures of multi-label data, which are label cardinality (Card) and density (Dens). Label cardinality represents the average number of labelsets (active labels) per instance, and it is computed as shown in Eq. (10). Where i is any instance in the dataset ($1 \leqslant i \leqslant m$), q and Y refer to the number of labels and labelsets, respectively [49].

$$Card = \frac{1}{m}\sum_{i=1}^{m}|Y_i| \tag{10}$$

Dividing the label cardinality by the number of labels (q) obtains the normalized version of label cardinality known as density [49], and it is computed as shown in Eq. (11).

$$Dens = \frac{1}{m}\sum_{i=1}^{m}\frac{|Y_i|}{q} \tag{11}$$

4.2. Methods

The primary aim of this work was to incorporate the preprocessing techniques, feature selection methods, and HOMER algorithm to obtain a competitive HMC model for the Arabic language. Thus, several experiments were conducted, which primarily focused on investigating the impact of the feature selection method and the dimension of the selected feature set on the classification performance of the model. In addition, the effect of the HOMER parameters (multi-label classifier, clustering algorithm, and the number of clusters) on the model performance were also examined.

Finally, after the HMATC model was constructed using the methods that obtained the best performance in each phase, it was then compared against four other models, two of which are baseline models and the two others are the state-of-the-art models — the BR, LP, CC, and Fatwa model [34], respectively. The results of these experiments are illustrated and discussed in Section 5.

4.3. Experimental setting

Different experiments were performed on the large dataset, which was associated with a large set of labels. Thus, the experiments required a large memory and greater computational resources. As a result, all experiments were carried out using the 'Aziz' High Performance Computing Center (HPCC)).[5] Aziz supports parallel processing and comprises of 496 computing nodes. There are 4 login servers in addition to several servers that offer different system services. The compute nodes are grouped into four types of queues depending on the physical characteristics of these computing nodes. All nodes run Unix operating system (CentOS 6.4). The experiments were conducted on the "Fat Queue" which contains 112 compute nodes, each node has 256 GB memory and an Intel Xeon processor with 2.40 GHz. These are vital properties for parallelizing experiments and reduce the total computational cost of the evaluation process.5-fold cross-validation sets were used to evaluate the

[5] https://www.hpcc-kau.com/aziz-super-computer.

predictive performance of the models. All the MLC methods were implemented using the MULAN multi-label learning tool [51], which is an open-source Java library that supports multi-label learning.

The dataset was prepared in ARFF file format, along with the XML file which are compatible with the MULAN tool. As the purpose of this study is to address the lack of available multi-label Arabic datasets, the processed version of the dataset used in the experiments has been made publicly available online.[6] It consists of 26,470 text instances distributed over 578 hierarchical multi-labels. It also contains different sets of high ranking features (1000,. . .,8000) in the ARFF file format, along with the XML file that defines the hierarchical structure of the labels.

5. Results and discussion

All the conducted experiments were evaluated according to the four example-based metrics (Hamming loss, H-loss, ML-accuracy, subset accuracy) and three label-based metrics (micro-averaged recall, micro-averaged precision, micro-averaged F-measure) dedicated to multi-label learning described in Section 3.5. The evaluated methods were ranked based on the seven metrics used, and the average ranks [52] were calculated in order to compare the effectiveness of the methods, whereby the method that obtained the best result was ranked with a 1. Several experiments were performed to explore the model's predictive performance from different perspectives; thus, they are discussed separately in the following sub-sections.

5.1. Feature selection methods

The first experiment investigated the impact of feature selection method on the predictive performance of the model. The standard multi-label feature selection approach described previously in Section 3.3 was applied. Two of the most common PT methods — LP and BR — were used with four feature ranking methods (χ^2, GR, RF, IG); thus, the methods evaluated were: BR-χ^2, BR-GR, BR-IG, BR-RF, LP-χ^2, LP-GR, LP-IG, and LP-RF.

Each feature selection method was used to select a subset of high-ranking features from the total number of features (11,000). Accordingly, the classification model was evaluated using the 2000 features selected by each feature selection method. Also, the HOMER was implemented using its default multi-label classifier (BR-NB) and balanced *k*-means algorithm with four clusters.

Table 6 presents the results of all evaluated feature selection methods. It also reports the average ranks for each method across all evaluation metrics. As the table shows, the BR-χ^2 method obtained the best results for Hamming loss, H-loss, ML-accuracy, micro-averaged precision, micro-averaged recall, and micro-averaged F-measure. Meanwhile, BR-GR performed better regarding subset accuracy (0.0394). In terms of average ranks, BR-χ^2 performed best followed by BR-IG, (1.29 and 2.29, respectively). This indicates that using a BR method, which transforms MLC problem into several single-label problems to determine the discriminative features for each label independently of other labels, obtains better results than LP method, which transforms MLC problem into a multi-class classification problem. The following are examples of the top ten high-ranking features over all labels (خطء، طالق، دي، نصيب، وارث، قتل، طلاق، صلا، يجوز، تراويح – mistake, divorced, blood money, share, heir, homicide, divorce, prayer, permissible, taraweeh).

[6] https://data.mendeley.com/datasets/rxhpvwwmbz/1.

5.2. Multi-label classifiers

The second experiment sought to optimize the HOMER algorithm by investigating different sets of multi-label classifiers in order to examine their effect on the predictive performance of the model. Three PT methods were employed as multi-label classifiers since they are among the most widely-used methods of the PT technique. These were: BR, CC, and LP classifiers. The PT methods were employed with three base classifiers (NB, SVM and J48). Accordingly, HOMER was run with each different multi-label classifier set (BR-NB, BR-J48, BR-SVM, CC-NB, CC-J48, CC-SVM, LP-NB, LP-J48 and LP-SVM).

The experiment was conducted using the BR-χ^2 feature selection method since it obtained the best results in the first experiment. It was employed to select 2000 high-ranking features. HOMER was implemented using balanced *k*-means clustering algorithm with four clusters. The results of all evaluated multi-label classifiers are presented in Table 7. As the table shows, the best results were obtained when HOMER was run using the LP-SVM multi-label classifier across all evaluation metrics with an average rank of 1, followed by BR-SVM and CC-SVM, which achieved average ranks of 2 and 3, respectively. It was observed that the SVM base classifier used with the PT methods (LP, CC, BR), achieved a better predictive performance than the other base classifiers (NB and J48). This indicates that SVM is a more effective classification algorithm and may improve classification performance [24].

5.3. Feature set dimensions

The third experiment was conducted using the feature selection method and the multi-label classifier that yielded the best results in the first two experiments. The HOMER algorithm was run using the LP-SVM multi-label classifier and balanced *k*-means clustering algorithm with four clusters. This experiment mainly sought to investigate the effect of the dimensions of the feature set on model performance. The BR-χ^2 feature selection method was used to select different sets of high-ranking features involves 1000, 2000,. . ., 8000 features, from the total number of features (11,000).

The results of the experiment are presented in Table 8, which shows that outcomes obtained for feature sets ranging from 1000 to 8000 features were approximately the same, with small differences in favor of the 4000-feature set, which obtained the best results in terms of Hamming loss, H-loss, ML-accuracy and micro-averaged F-measure. In the average ranks, the 4000 and 5000 features obtained the highest value, which was 2.29. Consequently, the 4000-feature set was employed with the evaluated models in order to avoid model overfitting and improve predictive performance.

5.4. Clustering algorithms

The fourth experiment sought to investigate the effect of three applied clustering algorithms on the model. These were: balanced *k*-means, *k*-means, and random clustering, which represent the three HOMER variations (HOMER-B, HOMER-K and HOMER-R), respectively. The model was evaluated using the methods and the feature set dimensions that achieved the best results in the previous experiments. The BR-χ^2 feature selection method with 4000 selected features was employed in the feature selection phase, and HOMER was implemented with the LP-SVM multi-label classifier in the classification phase.

Table 9 presents the results of the label distribution performed using the three clustering algorithms with different numbers of clusters k = 2, 4, 6, and 8. It was observed that when the number of clusters increased, model performance improved for all three

Table 6
Effect of feature selection methods on the predictive performance of the model.

Feature selection method	Hamming loss	H-loss	ML-accuracy	Subset accuracy	Micro-averaged Precision	Micro-averaged recall	Micro-averaged F-measure	Average Ranks
LP-χ^2	0.0163 (**6**)	3.9715 (**5**)	0.4128 (**6**)	0.0053 (**6**)	0.4906 (**6**)	0.6438 (**6**)	0.5567 (**6**)	5.86
LP-GR	0.0163 (**6**)	3.9715 (**5**)	0.4128 (**6**)	0.0053 (**6**)	0.4906 (**6**)	0.6438 (**6**)	0.5567 (**6**)	5.86
LP-IG	0.0163 (**6**)	3.9715 (**5**)	0.4128 (**6**)	0.0053 (**6**)	0.4906 (**6**)	0.6438 (**6**)	0.5567 (**6**)	5.86
LP-RF	0.0139 (**4**)	3.8185 (**4**)	0.4925 (**4**)	0.034 (**5**)	0.5481 (**4**)	0.7249 (**3**)	0.6242 (**3**)	3.86
BR-χ^2	**0.0123** (**1**)	**3.5999** (**1**)	**0.5191** (**1**)	0.0366 (**3**)	**0.5906** (**1**)	**0.7366** (**1**)	**0.6555** (**1**)	1.29
BR-GR	0.0146 (**5**)	4.1826 (**6**)	0.4805 (**5**)	**0.0394** (**1**)	0.5304 (**5**)	0.7141 (**5**)	0.6084 (**5**)	4.57
BR-IG	0.0135 (**2**)	3.7173 (**2**)	0.5002 (**2**)	0.0361 (**4**)	0.5566 (**2**)	0.729 (**2**)	0.6312 (**2**)	2.29
BR-RF	0.0138 (**3**)	3.7297 (**3**)	0.4936 (**3**)	0.0367 (**2**)	0.55 (**3**)	0.7213 (**4**)	0.6241 (**4**)	3.14

Table 7
Effect of the multi-label classifiers on the predictive performance of the model.

Multi-label classifier	Hamming loss	H-loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure	Average Ranks
BR-NB	0.0131 (**8**)	3.7131 (**9**)	0.513 (**9**)	0.04 (**8**)	0.5688 (**9**)	0.741 (**8**)	0.6435 (**9**)	8.57
BR-J48	0.0052 (**4**)	1.9367 (**4**)	0.7256 (**4**)	0.2462 (**5**)	0.8491 (**4**)	0.818 (**4**)	0.8332 (**4**)	4.14
BR-SVM	0.0048 (**2**)	1.8715 (**2**)	0.7436 (**2**)	0.2522 (**2**)	0.867 (**2**)	0.8248 (**2**)	0.8454 (**2**)	2
CC-NB	0.0129 (**8**)	3.6792 (**8**)	0.5149 (**8**)	0.0396 (**9**)	0.5727 (**8**)	0.7407 (**9**)	0.6459 (**8**)	8.29
CC-J48	0.0054 (**5**)	1.9947 (**5**)	0.7200 (**5**)	0.2456 (**6**)	0.8433 (**5**)	0.8148 (**5**)	0.8288 (**5**)	5.14
CC-SVM	0.0049 (**3**)	1.8945 (**3**)	0.7402 (**3**)	0.2516 (**3**)	0.8638 (**3**)	0.8224 (**3**)	0.8426 (**3**)	3
LP-NB	0.0102 (**7**)	3.1981 (**7**)	0.5771 (**7**)	0.0513 (**7**)	0.6520 (**7**)	0.7654 (**7**)	0.7042 (**7**)	7
LP-J48	0.0055 (**6**)	2.0550 (**6**)	0.7133 (**6**)	0.2478 (**4**)	0.8376 (**6**)	0.8085 (**6**)	0.8228 (**6**)	5.71
LP-SVM	**0.0047** (**1**)	**1.8280** (**1**)	**0.7483** (**1**)	**0.2720** (**1**)	**0.8705** (**1**)	**0.8261** (**1**)	**0.8477** (**1**)	1

Table 8
Effect of the dimensions of the features set on the predictive performance of the model.

Feature dimensions	Hamming loss	H-loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure	Average Ranks
1000	0.0049 (**3**)	1.8915 (**8**)	0.7392 (**8**)	0.2577 (**7**)	0.8721 (**1**)	0.8143 (**7**)	0.8422 (**7**)	5.86
2000	0.0047 (**1**)	1.828 (**7**)	0.7483 (**7**)	0.272 (**6**)	0.8705 (**2**)	0.8261 (**6**)	0.8477 (**3**)	4.57
3000	0.0047 (**1**)	1.8216 (**5**)	0.7494 (**3**)	0.2733 (**5**)	0.8682 (**3**)	0.8284 (**5**)	0.8478 (**2**)	3.43
4000	**0.0047** (**1**)	**1.8136** (**1**)	**0.7501** (**1**)	0.2755 (**4**)	0.8672 (**4**)	0.8301 (**4**)	**0.8482** (**1**)	2.29
5000	0.0047 (**1**)	1.8157 (**2**)	0.7498 (**2**)	0.2766 (**1**)	0.8661 (**5**)	0.8303 (**3**)	0.8478 (**2**)	2.29
6000	0.0048 (**2**)	1.8171 (**3**)	0.7493 (**4**)	0.2762 (**2**)	0.8654 (**6**)	0.8303 (**3**)	0.8475 (**4**)	3.43
7000	0.0048 (**2**)	1.8202 (**4**)	0.749 (**6**)	0.276 (**3**)	0.8645 (**7**)	0.8306 (**2**)	0.8472 (**6**)	4.29
8000	0.0048 (**2**)	1.8242 (**6**)	0.7492 (**5**)	0.276 (**3**)	0.8639 (**8**)	0.8313 (**1**)	0.8473 (**5**)	4.29

Table 9
Effect of clustering algorithm along with a set of number of clusters k = {2,4,6,8} on the predictive performance of the model.

Clustering algorithm	k	Hamming loss	H-loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure	Average ranks
Balanced *k*-means	k = 2	0.0049 (**5**)	1.9134 (**11**)	0.7414 (**11**)	0.256 (**11**)	0.8583 (**10**)	0.8262 (**8**)	0.8419 (**9**)	9.28
	k = 4	0.0047 (**3**)	1.8136 (**8**)	0.7501 (**8**)	0.2755 (**8**)	0.8672 (**8**)	0.8301 (**5**)	0.8482 (**6**)	6.57
	k = 6	0.0046 (**2**)	1.7499 (**5**)	0.7552 (**5**)	0.2845 (**5**)	0.8770 (**4**)	0.8272 (**7**)	0.8514 (**3**)	4.42
	k = 8	**0.0045** (**1**)	1.7232 (**2**)	**0.7581** (**1**)	0.2921 (**3**)	0.8795 (**3**)	0.8287 (**6**)	0.8533 (**2**)	2.57
***k*-means**	k = 2	0.0049 (**5**)	1.8555 (**10**)	0.7460 (**10**)	0.2655 (**10**)	0.8568 (**11**)	0.8325 (**2**)	0.8445 (**8**)	8
	k = 4	0.0048 (**4**)	1.7922 (**7**)	0.7518 (**7**)	0.2841 (**6**)	0.8622 (**9**)	0.8337 (**1**)	0.8477 (**7**)	5.87
	k = 6	0.0046 (**2**)	1.7497 (**4**)	0.7569 (**3**)	0.2953 (**2**)	0.8711 (**7**)	0.8318 (**3**)	0.8510 (**4**)	3.57
	k = 8	0.0046 (**2**)	1.7292 (**3**)	0.7561 (**4**)	**0.3020** (**1**)	0.8716 (**6**)	0.8306 (**4**)	0.8506 (**5**)	3.57
Random clustering	k = 2	0.0050 (**6**)	1.9366 (**12**)	0.7397 (**12**)	0.2543 (**12**)	0.8551 (**12**)	0.8260 (**9**)	0.8403 (**10**)	10.42
	k = 4	0.0047 (**3**)	1.8243 (**9**)	0.7492 (**9**)	0.2691 (**9**)	0.8722 (**5**)	0.8246 (**10**)	0.8477 (**7**)	7.42
	k = 6	0.0046 (**2**)	1.7718 (**6**)	0.7551 (**6**)	0.2805 (**7**)	0.8808 (**2**)	0.8233 (**11**)	0.851 (**4**)	5.42
	k = 8	**0.0045** (**1**)	**1.7149** (**1**)	0.7579 (**2**)	0.2917 (**4**)	**0.8832** (**1**)	0.8260 (**9**)	**0.8536** (**1**)	2.71

HOMER variations. Then, the three clustering algorithms were compared with a high number of clusters (k = 8 clusters). Label distribution performed using the balanced k-means algorithm obtained the best results for Hamming loss (0.0045) and ML-accuracy (0.7581), random clustering performed best regarding Hamming loss (0.0045), H-loss (1.7149), micro-averaged precision (0.8832), and micro-averaged F-measure (0.8536), and the k-means algorithm obtained the best result for subset accuracy (0.302).

In the average ranks, the balanced k-means algorithm scored the highest (2.57), followed by random clustering and the k-means algorithm with 2.71 and 3.57, respectively. This indicates that performing even label distribution using a balancing clustering algorithm (balanced k-means) is more effective in this domain than performing label distribution using a simple clustering algorithm (k-means). As stated in [24], this may be because balanced clustering is more useful in domains with a large number of labels,

Table 10
Compare HMATC model against state-of-the-art (Fatwa, CC) and baseline (BR, LP) models.

Model	Hamming loss	H-loss	ML-accuracy	Subset accuracy	Micro-averaged precision	Micro-averaged recall	Micro-averaged F-measure	Average Ranks
HMATC	0.0045 **(2)**	1.7232 **(4)**	0.7581 **(3)**	0.2921 **(3)**	0.8795 **(3)**	0.8287 **(2)**	0.8533 **(3)**	2.85
Fatwa	0.0191 **(5)**	4.7739 **(5)**	0.4622 **(5)**	0.0309 **(5)**	0.4428 **(5)**	0.7654 **(5)**	0.5610 **(5)**	5
CC	0.0046 **(3)**	1.6168 **(2)**	**0.7635 (1)**	0.3335 **(2)**	0.8721 **(4)**	**0.8364 (1)**	0.8539 **(2)**	2.14
BR	**0.0043 (1)**	1.6579 **(3)**	0.7583 **(2)**	0.2703 **(4)**	**0.9047 (1)**	0.8155 **(3)**	**0.8577 (1)**	2.14
LP	0.0048 **(4)**	**1.61 (1)**	0.7433 **(4)**	**0.3562 (1)**	0.8888 **(2)**	0.7949 **(4)**	0.8392 **(4)**	2.85

and it also helps to avoid class imbalance. Hence, it is possible to conclude that implementing HOMER using a balanced *k*-means clustering algorithm with a large number of clusters will obtain the best results.

5.5. Model comparisons

Finally, the HMATC was compared with the state-of-the-art models (Fatwa, CC) and the baseline models (LP, BR). This experiment was conducted using the BR-χ^2 feature selection method and 4000 selected features. The HMATC model was implemented using the HOMER algorithm with LP-SVM multi-label classifier and balanced *k*-means clustering algorithm with eight clusters. In addition, the BR, CC, and LP models were implemented using SVM as a base classifier. Furthermore, the Fatwa model was implemented, in accordance with [34], by running the HOMER algorithm using the BR-NB multi-label classifier.

As presented in Table 10, BR obtained the best results in terms of Hamming loss, micro-averaged precision, and micro-averaged F-measure, whereas, CC yielded the best result for ML-accuracy and micro-averaged recall, and LP outperformed the other models in terms of H-loss and subset accuracy.

The average ranks presented in Table 10 show that the BR and CC models performed better than other models (2.14), followed by the HMATC and LP models with 2.85, and finally the Fatwa model with 5. A Friedman test was conducted based on the average ranks for all evaluated models [53]. This test ranked each metric separately over the five evaluated models; thereby, the best model obtained the rank of 1, the second best the rank of 2, and so on. After this, it calculated the test statistic on the average ranks to analyze the significant differences in the models' performance.

By performing the Friedman test and obtaining the distribution with a $k - 1$ degree of freedom, the *p*-values were measured at a significance level of 5%. Since the Friedman test gave a significant difference of 5% for all metrics, the null-hypothesis supposing that all the models would be equivalent was rejected. A Wilcoxon signed-rank test [53] at a 5% significance level was performed. The *p*-value of the comparison between the HMATC model and the Fatwa model = 0.016. Thus, the null-hypothesis supposing that both models would be equally effective was rejected and the alternative accepted, indicating that the HMATC model outperformed the Fatwa model since it achieved a higher average score. In contrast, the HMATC model was compared with the CC, BR, and LP models, which obtained the *p*-values 0.160, 0.160, and 1.000, respectively. Thus, the null hypothesis indicating that both models would be equally effective was also accepted.

Overall, it is possible to conclude that the HMATC, CC, BR, and LP models are equally effective. Although BR, CC, and LP outperformed the HMATC model with a small difference in terms of some evaluation metrics, they suffered from some drawbacks such as a lack of scalable efficiency with a large number of labels. Due to the large number of labels in the dataset, the models implemented using BR, CC, LP classifiers suffered from class imbalance, high computational cost, and the inefficiency of applications requiring fast response times. For instance, BR needed to train a set of base classifiers equal to the number of labels, so it employed 578 base classifiers to equal the number of labels in the dataset.

Moreover, CC suffered from random order chain. The results also indicated that the chaining property in CC may cause error propagation at classification time. Furthermore, LP experienced an exponential increase in computational complexity in relation to the number of labels. Additionally, it could only predict labelsets observed in the training data and it also suffered from model overfitting.

In contrast, the HMATC, which was built based on the HOMER algorithm, efficiently dealt with a large number of labels by constructing a tree-shaped hierarchy of simpler MLC problems, in which the root node was constructed to contain all labels in the tree. Then, the balanced *k*-means algorithm was employed to distribute large labelsets into balanced clusters that represented new nodes. For each cluster, the LP-SVM multi-label classifier was employed (since it achieved the best results) to handle labels in that cluster only. If the predicted label was in the meta-labels of the child node, only the multi-label classifier of that node was activated.

One of the advantages of the balanced clustering algorithm was that the related labels belonged to the same cluster, which meant that only the multi-label classifier of that node will be invoked and the computational cost was reduced. Also, since each node dealt with fewer training instances, predictive performance was improved.

On the other hand, the Fatwa model was also built based on the HOMER algorithm, but the HOMER algorithm in the Fatwa model was implemented using the BR-NB classifier, which led to lower predictive performance compared to the other evaluated models.

5.6. Computational cost

Since computational cost is a vital factor in performance evaluation, it was essential to measure the computational cost consumed by all evaluated models (BR, LP, CC, Fatwa, HMATC). Fig. 4 presents the computational cost of each model (in hours). The figure shows that the evaluated models can be ranked based on the lowest cost achieved in the following order: HMATC, Fatwa, BR, CC, and LP model. It is important to note that all models were

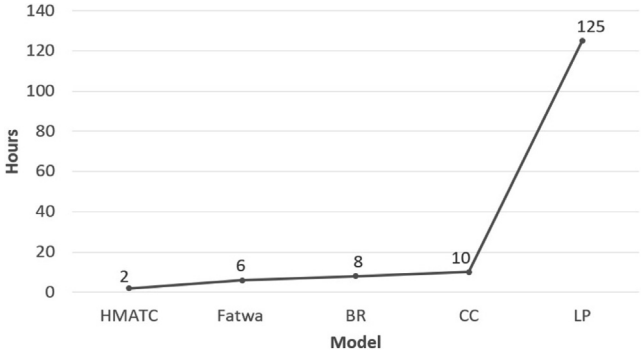

Fig. 4. Computational cost required for each evaluated models.

evaluated using the same experimental setting, as described previously in Section 4.3. Overall, the HMATC model consumed less computational cost (2 h) than the other evaluated models. This may be because the HMATC model was implemented using the HOMER algorithm, which is an effective and computationally efficient hierarchical multi-label algorithm that requires a linear training and logarithmic testing complexities with respect to the number of labels.

Furthermore, the effect of each different set of methods was investigated in each phase. Then, the HMATC model was implemented using the methods which obtained the best performance. This led to improved model performance and reduced total computational cost.

6. Conclusion

Hierarchical classification is an important research field and it has been increasingly required by many applications in various domains, including text classification. This paper proposed a model for hierarchical multi-label text classification of the Arabic language. The HMATC model was implemented based on the HOMER algorithm, which was optimized by employing LP-SVM classifier and balanced *k*-means algorithm with eight clusters in order to improve hierarchical classification outcomes. The researchers prepared a hierarchical multi-label Arabic dataset in an appropriate format for hierarchical multi-label classification. This dataset was published online to address the lack of publicly available multi-labelled Arabic datasets, especially those which are well annotated with hierarchical multi-labels, and also to encourage future researchers to investigate hierarchical multi-label classification methods for Arabic text.

Furthermore, the impact of feature selection method and feature set dimensions on the predictive performance of the model was investigated. The results showed that the HMATC model outperformed all the evaluated models in terms of computational cost. In addition, it was equally effective with the BR, CC, and LP models and performed significantly better than the state-of-the-art model (Fatwa model) over a wide range of evaluation metrics.

The researchers intend to extend this work in the future by investigating the HMATC model with other multi-label classifiers, such as RA*k*EL, RPC, and PS. They also aim to apply different structured methods for selecting the number of clusters in the clustering algorithm (*k*). Regarding the feature selection phase, the authors intend to examine a principal component analysis (PCA) method instead of transforming the multi-label problem using a PT method. Finally, they plan to give more attention to the preprocessing phase by employing word embedding techniques like word2vec and FastText as text representation methods, applying different stemming algorithms, and preparing a list of Arabic human names so they can be easily removed during the stop word removal phase.

Acknowledgment

This work was supported by the Deanship of Scientific Research (DSR), King Abdulaziz University, Jeddah, Saudi Arabia, under Grant No. (DG 29-612-1440). The authors, therefore, gratefully acknowledge the DSR technical and financial support.

References

[1] Al-Salemi B, Ayob M, Kendall G, Noah SAM. Multi-label arabic text categorization: a benchmark and baseline comparison of multi-label learning algorithms. Inf Process Manage 2019;56(1):212–27.
[2] Al-Salemi B, Noah SAM, Ab Aziz MJ. Rfboost: an improved multi-label boosting algorithm and its application to text categorisation. Knowl-Based Syst 2016;103:104–17.
[3] Gibaja E, Ventura S. A tutorial on multi tutorial on multilabel learningilabel learning, ACM Comput Surv 47(3):2015; 52:1–52:38. [Online]. Available: http://doi.acm.org/10.1145/2716262..
[4] Tsoumakas G, Katakis I. Multi-label classification: an overview. Int J Data Warehousing Min (IJDWM) 2007;3(3):1–13.
[5] Taha AY, Tiun S. Binary relevance (br) method classifier of multi-label classification for arabic text. J Theor Appl Inf Technol 84(3):2016..
[6] Duwairi RM, Al-Zubaidi R. A hierarchical k-NN classifier for textual data. Int Arab J Inf Technol 2011;8(3):251–9.
[7] Brucker F, Benites F, Sapozhnikova E. Multi-label classification and extracting predicted class hierarchies. Pattern Recogn 2011;44(3):724–38.
[8] Silla CN, Freitas AA. A survey of hierarchical classification across different application domains. Data Min Knowl Discovery 2011;22(1–2):31–72.
[9] Mubarak H, Darwish K. Using twitter to collect a multi-dialectal corpus of arabic. In: Proceedings of the EMNLP 2014 workshop on arabic natural language processing (ANLP). p. 1–7.
[10] Eldos T. Arabic text data mining: a root-based hierarchical indexing model. Int J Model Simul 2003;23(3):158–66.
[11] Ahmed Y, Xiang J, Zhao D, Al-qaness MAA, Elsayed abd el aziz M, Abdelghani D. A study of the effects of stemming strategies on arabic document classification. IEEE Access PP:2019;1–1..
[12] Zhang M-L, Zhou Z-H. Ml-knn: a lazy learning approach to multi-label learning. Pattern Recogn 2007;40(7):2038–48.
[13] Clare A, King RD. Knowledge discovery in multi-label phenotype data. In: European conference on principles of data mining and knowledge discovery. Springer; 2001. p. 42–53..
[14] Boutell MR, Luo J, Shen X, Brown CM. Learning multi-label scene classification. Pattern Recogn 2004;37(9):1757–71.
[15] Read J, Pfahringer B, Holmes G, Frank E. Classifier chains for multi-label classification. Mach Learn 2011;85(3):333.
[16] Hüllermeier E, Fürnkranz J, Cheng W, Brinker K. Label ranking by learning pairwise preferences. Artif Intell 2008;172(16–17):1897–916.
[17] Tsoumakas G, Vlahavas I. Random k-labelsets: an ensemble method for multilabel classification. In: European conference on machine learning. Springer; 2007. p. 406–417..
[18] Read J, Pfahringer B, Holmes G. Multi-label classification using ensembles of pruned sets. In: Data mining, 2008. ICDM'08. Eighth IEEE international conference on. IEEE; 2008. p. 995–1000..
[19] Tsoumakas G, Katakis I, Vlahavas I. Random k-labelsets for multilabel classification. IEEE Trans Knowl Data Eng 2011;23(7):1079–89.
[20] Vens C, Struyf J, Schietgat L, Džeroski S, Blockeel H. Decision trees for hierarchical multi-label classification. Mach Learn 2008;73(2):185.
[21] Cesa-Bianchi N, Gentile C, Zaniboni L. Incremental algorithms for hierarchical classification. J Mach Learn Res 7(Jan);2006:31–54..
[22] Chen Y, Crawford MM, Ghosh J. 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. p. 949–952..
[23] Zhang L, Shah S, Kakadiaris I. Hierarchical multi-label classification using fully associative ensemble learning. Pattern Recogn 2017;70:89–103.
[24] Tsoumakas G, Katakis I, Vlahavas I. 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..
[25] Yahya A, Salhi A. Arabic text categorization based on arabic wikipedia. ACM Trans Asian Lang Inf Process (TALIP) 2014;13(1):4.
[26] Ahmed NA, Shehab MA, Al-Ayyoub M, Hmeidi I. Scalable multi-label arabic text classification. In: Information and communication systems (ICICS), 2015 6th international conference on. IEEE; 2015. p. 212–217..
[27] Shehab MA, Badarneh O, Al-Ayyoub M, Jararweh Y. 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. p. 1–6..
[28] Hmeidi I, Al-Ayyoub M, Mahyoub NA, Shehab MA. A lexicon based approach for classifying arabic multi-labeled text. Int J Web Inf Syst 2016;12(4):504–32.
[29] Fürnkranz J, Hüllermeier E, Mencía EL, Brinker K. Multilabel classification via calibrated label ranking. Mach Learn 2008;73(2):133–53.
[30] Schapire RE, Singer Y. Improved boosting algorithms using confidence-rated predictions. Mach Learn 37(3);1999:297–336. [Online]. Available: doi: 10.1023/A:1007614523901..
[31] Spyromitros E, Tsoumakas G, Vlahavas I. An empirical study of lazy multilabel classification algorithms. In: Darzentas, J., Vouros, G.A., Vosinakis, S., Arnellos, A. (Eds.), Artificial intelligence: theories, models and applications, Berlin, Heidelberg: Springer, Berlin Heidelberg; 2008. p. 401–406..
[32] Cheng, W, Hüllermeier, E. Combining instance-based learning and logistic regression for multilabel classification. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (Eds.), Machine learning and knowledge discovery in databases. Berlin, Heidelberg: Springer, Berlin Heidelberg; 2009. p. 6–6..
[33] Elnagar A, Al-Debsi R, Einea O. Arabic text classification using deep learning models. Inf Process Manage 2020;57(1). 102121.
[34] Zayed, RA, Hady MFA, Hefny H. Islamic fatwa request routing via hierarchical multi-label arabic text categorization. In: Arabic computational linguistics (ACLing), 2015 first international conference on. IEEE; 2015. p. 145–151..
[35] Ababneh J, Almomani O, Hadi W, El-Omari NKT, Al-Ibrahim A. Vector space models to classify arabic text. Int J Comput Trends Technol (IJCTT) 2014;7 (4):219–23.

[36] Mustafa SH. Word stemming for arabic information retrieval: the case for simple light stemming. Abhath Al-Yarmouk Sci Eng Ser 2012;21(1):2012.
[37] Froud H, Lachkar A, Ouatik SA. A comparative study of root-based and stem-based approaches for measuring the similarity between arabic words for arabic text mining applications, arXiv preprint arXiv:1212.3634, 2012..
[38] Habib MB. An intelligent system for automated arabic text categorization, Master's thesis, University of Twente; 2008..
[39] Joachims T. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Carnegie-mellon univ pittsburgh pa dept of computer science, Tech. Rep.; 1996..
[40] Karisani P, Rahgozar M, Oroumchian F. A query term re-weighting approach using document similarity. Inf Process Manage 2016;52(3):478–89.
[41] Wu H, Gu X, Gu Y. Balancing between over-weighting and under-weighting in supervised term weighting. Inf Process Manage 2017;53(2):547–57.
[42] Uysal AK, Gunal S. The impact of preprocessing on text classification. Inf Process Manage 2014;50(1):104–12.
[43] Syiam M, Fayed ZT, Habib MB. An intelligent system for arabic text categorization. Int J Intell Comput Inf Sci 2006;6(1):1–19.
[44] Ayedh A, Tan G. Building and benchmarking novel arabic stemmer for document classification. J Comput Theor Nanosci 2016;13(3):1527–35.
[45] Larkey LS, Ballesteros L, Connell ME. Improving stemming for arabic information retrieval: light stemming and co-occurrence analysis. In: Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval. p. 275–82.
[46] Chen Y-T, Chen MC. Using chi-square statistics to measure similarities for text categorization. Expert Syst Appl 2011;38(4):3085–90.
[47] Lee C, Lee GG. Information gain and divergence-based feature selection for machine learning-based text categorization. Inf Process Manage 2006;42 (1):155–65.
[48] SpolaôR N, Cherman EA, Monard MC, Lee HD. A comparison of multi-label feature selection methods using the problem transformation approach. Electron Notes Theor Comput Sci 2013;292:135–51.
[49] Tsoumakas G, Katakis I, Vlahavas I. Mining multi-label data. In: Data mining and knowledge discovery handbook. Springer; 2009. p. 667–685..
[50] Zhu S, Ji X, Xu W, Gong Y. 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. p. 274–281.
[51] Tsoumakas G, Spyromitros-Xioufis E, Vilcek J, Vlahavas I. Mulan: a java library for multi-label learning. J Mach Learn Res 2011;12:2411–4.
[52] Brazdil PB, Soares C. A comparison of ranking methods for classification algorithm selection. In: European conference on machine learning. Springer; 2000. p. 63–75..
[53] Demšar J. Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7(Jan);2006:1–30.

Nawal Aljedani received the B.S. and M.S. degrees in information technology from King Abdulaziz University, Jeddah, Saudi Arabia. She is currently a Collaborative Teaching Assistant with the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests include machine learning, data mining, natural language processing, and multi-label classification.

Reem Alotaibi received the Ph.D. degree in computer science from the University of Bristol, Bristol, U.K., in 2017. She is currently an Assistant Professor with the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests include machine learning, data mining, and multi-label classification.

Mounira Taileb received the Ph.D. degree in computer science from the University of Paris-Sud, Paris, France, in 2008. She is currently an Associate Professor at the Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests include image retrieval, image annotation, machine learning, data mining, and text classification.