Information Sciences Letters

Volume 12
Issue 3 *Mar. 2023*

Article 41

2023

Performance Analysis of Machine Learning Approaches in Automatic Classification of Arabic Language

Fahd S. Alharithi
Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia, f.alshalawi@tu.edu.sa

Inf. Sci. Lett. **12**, No. 3, 1563- 1578 (2023)

Information Sciences Letters
An International Journal

http://dx.doi.org/10.18576/isl/120342

Performance Analysis of Machine Learning Approaches in Automatic Classification of Arabic Language

Fahd S. Alharithi

Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

Received: 9 Aug. 2022, Revised: 21 Dec. 2022, Accepted: 27 Dec .2022.
Published online:1 Mar. 2023.

Abstract: Text classification (TC) is a crucial subject. The number of digital files available on the internet is enormous. The goal of TC is to categorize texts into a series of predetermined groups. The number of studies conducted on the English database is significantly higher than the number of studies conducted on the Arabic database. Therefore, this research analyzes the performance of automatic TC of the Arabic language using Machine Learning (ML) approaches. Further, Single-label Arabic News Articles Datasets (SANAD) are introduced, which contain three different datasets, namely Akhbarona, Khaleej, and Arabiya. Initially, the collected texts are pre-processed in which tokenization and stemming occur. In this research, three kinds of stemming are employed, namely light stemming, Khoja stemming, and no-stemming, to evaluate the effect of the pre-processing technique on Arabic TC performance. Moreover, feature extraction and feature weighting are performed; in feature weighting, the term weighting process is completed by the term frequency-inverse document frequency (tf-idf) method. In addition, this research selects C4.5, Support Vector Machine (SVM), and Naïve Bayes (NB) as a classification algorithm. The results indicated that the SVM and NB methods had attained higher accuracy than the C4.5 method. NB achieved the maximum accuracy with a performance of 99.9%.

Keywords: Text Classification, SVM, NB, C4.5, SANAD, Machine Learning, Stemming, Arabic Text.

1 Introduction

A considerable number of repositories now exist online due to the widespread use of Web 2.0 and the Internet, necessitating the development of automatic classification systems. Although about 80% of data remains textual and unorganized, it is nonetheless regarded as a valuable and rich data resource. Machine Learning (ML) techniques have proven to reduce the time it takes to retrieve insights and analyze large amounts of data. Because of the particular character of the Arabic language, developing an automatic text classification system for Arabic articles/documents has been a complex task. Arabic is a 28-letter language written from right to left. Its orthography and morphology concepts are unique. Text classification was a crucial assignment in Natural Language Processing (NLP). It's utilized to give textual information labels predicated on its contexts. Automating the procedure streamlines document classification, aids platform interoperability, and makes searching for particular data simple and achievable. Efficient automatic text categorization is required for optimal data categorization and management. As a result, text categorization or classification continues to be an essential subject of research that receives much attention. In brief, TC seems to be a supervised ML technique that dynamically applies predefined categories (labels) to a given case depending on its contents [1]. A classification method is taught utilizing training data consisting of examples and associated types [2].

Moreover, classification means determining, distinguishing, and comprehending thoughts and entities. The term "classification" refers to grouping items into distinct groups, typically for a particular reason. A category investigates the connection between words, knowledge tasks, and words and classifications. Short and long texts are included in the text categorization. Text categorization (TC) emerges as a critical tool for dealing with and structuring vast amounts of data. Text classification (TC) would be a crucial job employed in data management technologies to assign a given content to one or more predetermined classes. Automatic TC has been considered an ML technique. This method aims to see if a document corresponds to a specific classification by examining the terms or words in that category. This TC object

*Corresponding author e-mail: f.alshalawi@tu.edu.sa

research seeks to improve text summarization quality and produce high-quality classifiers [3].

Moreover, TC assists users in maintaining their domains of focus by automatically classifying texts as per their topics, allowing them to quickly filter out texts unrelated to their focus. As a result, these groupings can be used to improve certain activities, such as obtaining search results and enhancing user expertise in exploring the crucial text dataset [4]. Over the last several decades, TC challenges have been researched extensively and handled in different real-world applications [5, 6]. Many scholars are presently interested in designing technologies that use TC approaches, specifically in light of recent NLP and text mining advancements Extraction of features, dimension minimization, selection of classifier, and evaluations are the four procedures that many TC and document classification systems go through. Generally, there are four primary grades of scope, which can be implemented to the TC system is scribed as follows [7]:

1. Paragraph level: The method obtains the applicable classifications of a solitary paragraph at the paragraph levels (a document's portion).

2. Sub-sentence level: The method extracts the appropriate classifications of sub-expressions within such a paragraph at the sub-sentence phase (sentence's portion).

3. Document level: The strategy gathers the relevant classifications of a whole document somewhere at the textual level.

4. Sentence level: Determine the applicable classifications of a solitary sentence (a section of a paragraph) somewhere at the lexical level.

In addition, the four processes in TC and document classification system are described below [7]:

a. Feature extraction: Documents and texts were unstructured information collections in principle. When retaining mathematical modeling as a classifier component, these unstructured text patterns must be interpreted into an organized feature set. The files must be cleaned to remove any extraneous words or characters to instigate. Technical feature extraction techniques could be used once the information was cleaned.

b. Dimensionality reduction: Data pre-processing activities can take quite a long time and demand much memory since file or text information compilations typically contain many unique words. The use of low-cost methods is a typical response to this problem. However, these low-cost techniques in specific databases do not perform as well as expected. Several researchers use dimensionality minimizations to reduce their procedures' memory complexity and duration to avoid performance loss. It's possible that pre-processing using dimensionality lowering is more successful than building low-cost learners.

c. Classification techniques: The selection of the optimal classifier seems to be the most crucial phase in the TC process. It is impossible to select the most effective method for a TC implementation without an inclusive intangible grasp of each technique.

d. Evaluation: Evaluation was the last step in the TC process. The usage and advancement of TC approaches require knowledge of how a system works. There are several approaches for calculating ML algorithms. The modest way of assessment is accuracy computation, but it does not function for imbalanced databases [8].

The problem of TC can be described as automatically categorizing a batch of documents into one or more predetermined classes based on their topics [9]. As a result, the primary goal of TC would be to develop algorithms for classifying NLP documents [10]. A group of training files $D = \{d_1....d_n\}$ with predefined classes $C = \{c_1,...,c_q\}$ and a unique document q that is commonly referred to as query would forecast the query's class that will correspond to one or both of the class in C [11]. TC approaches are used for various activities, including finding related documents, classifying subjects by documents from relevant documentation, and creating documents in different subjects. As a result, the categorization's goal is to dynamically assign the appropriate classification to every document, which has to be classified. TC may also be employed in a variety of NLP tasks. Conventional TC approaches rely on knowledge rules, dictionaries, and separate tree kernels, which are all human-designed constructs [12].

TC is subdivided into several sub-problems, including sentiment classification, functional classification, subject classification, and other types of classification [13]. Nonetheless, the emphasis of this research is on text categorization or text classification into different classifications. Automatic TC was a machine learning issue in which a corpus of classified texts was used to train the algorithm. This method often assigns topics to one or more known class labels. Various strategies were used to classify these subjects into separate fields.

A supervised classification technique has been used to employ a group of predetermined class texts generated as a learning approach. TC plays a vital role in various applications, including word sense disambiguation, information retrieval, web page classification, and other areas that require text arrangement [14]. TC can be utilized to find documents that are related

to each other. However, one of the most challenging aspects of using this sort of region is the difficult classification and limiting of materials for the same subject [15].

The six main phases of the TC process are the data document collection, text pre-processing, extraction of features, dimensionality reduction, various classification approaches, and performance assessment. Moreover, the TC process is represented in Fig. 1.

1. Data collection

The first phase was data gathering, which comprises the collection of various databases in formats like PDF, HTML, and DOC, among others.

2. Text pre-processing

Stop word removal, tokenization, stemming, and vector space modeling are the four phases of text preparation. Tokenization seems to be a method of removing special characters and white space from a document. Stop words were general terms employed to delete informational data with minimal meaning; they provide a grammatical function but don't reveal subject matters. The fact that a collection of operational English words (e.g., "a," "the," "and," and "that") has been ineffective as weighted terms were well established among information retrieval specialists. Because they occur in each English text file [16], these terms have a relatively poor discrimination value. The list of words obtained by word separation was examined to ensure that no terms appeared in the stop lists.

Stemming seems to be a method that reduces altered terms to their phrase stems to eliminate prefixes and suffixes from keywords. As a result, when the various forms of keywords were generated as unique, the process was used to reduce the keywords numeral in the keyword field and improve the effectiveness assessment for TC. Furthermore, it is frequently used in knowledge retrieval assignments to improve recall [16] and obtain the most relevant answers, such as sky- > ski. Vector Space method: Each word in the training sets could be specified as vectors in the form (x, td), here, x ∈ Rn represents a dimensions vector, n represents the total of terms, and d represents the class tag. A solitary keyword has calculated the appearance frequency for each keyword in that file in the feature space, which is why the TC, IR-based vector space paradigm is frequently used in the data presentation of documents.

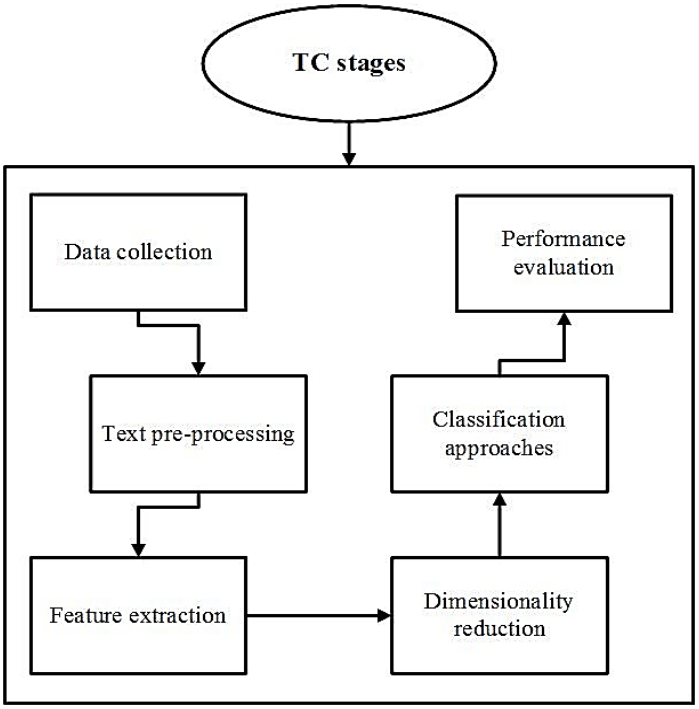


Figure.1: TC stages

3. Feature extraction (FE)

The primary goal of extracting features is to renovate a text from every setting into a keyword pattern that ML will effortlessly process. FE also gives data about the texts, such as the maximum phrase recurrence for each book. They select relevant keywords and determine the mechanism for encoding them in an ML model. Those keywords could significantly affect the capacity of classification systems to obtain the best sequence.

4. Feature weighting

Supervised Term Weighting (STW) seems to be a word weighting approach specifically explored for information retrieval domains connected to ML, like document cleansing and TC. Debole and Sebastiani [17] argued that supervised word sorts of effects training information by weighting a term, rendering how varied its circulation could be in the positively and negatively learning instances. The term's weighting methods are represented in table. 1.

Table 1. Terms weighting methods

Schemes (Methods)	**Scheme description**
tf	Term_Frequency
bool	Boolean method
wc	Word counts of output
idf	Inverse-Document-Frequency
tfidf_norm_minFreq5	Perform normalization with the minimum possible frequency
wc_minFreq3	3 for word count > minimum frequency
wc_norm	Perform normalization for word count
tfidf_norm_minFreq3	Perform normalization with the minimum possible frequency
wc_norm_minFreq5	Perform normalization with the minimum possible frequency
wc_minFreq5	5 for word count > minimum frequency
wc_norm_minFreq3	Perform normalization with the minimum possible frequency
tfidf	Term_Frequency based Inverse_Document

5. Dimensionality Reduction (DR)

TC, on the other hand, has been coping with the massive dataset. As a result, Dimension Reduction (DR) was critical in TC since it reduces the computation budget while providing the most excellent dimensional document labeled. A deeper photographic and visual study of the attentiveness data may give the users DR. There are three main classes of DR approaches [18]. The strategies are structured to benefit from class information data when calculating in the smallest possible space. Some systems use a feature selection process that reduces the DR by picking a subset of the relevant keywords.

The first category of approaches is used to generate new keywords by classifying the term [19]. These DR approaches aim to reduce the information cost associated with the original information while preserving the connection space established in the corpus. The techniques of computation are predicated on statistical assessment. The second category of DR approaches embraces Multi-Dimensional Scaling (MDS), Principal Component Analysis (PCA), and latent semantic indexing. This category can be used to represent linear relationships between dimensions.

6. Classification techniques

The classification methodology is a time-saving way of classifying patterns from a data set based on revenue. This method asks an ML algorithm to establish a routine that identifies the relationship between the keywords grouping and the revenue data's category tag. This ML approach must match the revenue data and predict the category tags of previously unappreciated registrants. Several techniques, including Support Vector Machine (SVM), k-nearest neighbour (KNN), and Naïve Bayes, have been utilized to categorize the texts into one or more regions predicated on their contents. In this research, Naïve Bayes, SVM, and C4.5 algorithms have been employed for TC. Moreover, after TC, the performance of the methods is evaluated.

The critical contribution of the present research is described as follows:

- Initially, it introduced new massive, well-annotated datasets of Arabic text gathered from a variety of news. All databases on Arabic computational linguistics were open to the scientific community.
- Hereafter, ML approaches have been employed for automatic Arabic text categorization.
- In addition, text pre-processing, feature selection, and feature weighting were also performed.

- Finally, the selected two different databases performance is computed; moreover, the evaluation chosen metrics are precision, F1-score, recall, and accuracy.

2 Related works

Some of the recent literature related to text classification or categorization is described as follows:

For TC, the bag of words paradigm has often been employed. The fundamental flaw in this approach is the enormous number of features included, which impacts classification task effectiveness. A feature selection strategy has been required to solve this challenge. Feature selection was good for minimizing the problem's complexity, minimizing computing time, and improving the classification task's efficacy. Therefore, Mouhoub Belazzoug et al. [20] have introduced a new enhanced technique for selecting features based on the initial SCA, allowing more significant search space expansion. The evaluation findings showed that the SCA technique outperformed the competition, making it a valuable tool for TC. However, it cannot be used to solve complex optimization issues.

In the fields of NLP and ML, classifying Arabic text content is an important topic. The number of Arabic documents has rapidly grown as news stories, new web pages, and social media information are posted daily. As a result, many individuals and organizations place great value on sorting such papers into specified categories. Convolutional Neural Networks (CNNs) have been a kind of DL, which has been proven beneficial for various NLP applications, notably TC and text translation in English. Word embedding seems to be a text format representing both semantic and syntactic aspects of text via real-valued variables in vector space. Conventional ML methods were already employed in Arabic TC with positive results. However, M. Alhawarat and Ahmad O. Aseeri et al. [21] presented SATCDM, a Multi-Kernel CNN prototype for categorizing Arabic news texts accompanied by n-gram word embedding. In contrast to current work in Arabic TC employing 15 openly accessible databases, the anticipated approach obtained high accuracy. As a result, this strategy outperforms similar research on classifying Arabic documents. Furthermore, the computing time is extended.

The technique of autonomously tagging text contents with the most appropriate categories or labels is often known as TC. The task became single-label TC whenever the titles were limited to one. On the other hand, the multi-label variant is complex. The lack of vast and open Arabic rich and logical databases makes these tasks far more challenging for the Arabic language. As a result, Ashraf Elnagar et al. [22] have developed new comprehensive and unbiased databases for both multi-label (NADiA) and single-label (SANAD) Arabic TC applications. In addition, a detailed assessment of many DL algorithms for Arabic TC was performed to assess their efficiency on NADiA and SANAD. The outcomes of the experiments revealed that all methods performed admirably on the SANAD dataset, with better accuracy. However, the NADiA database has attained less accuracy.

The difficulty of categorizing text using a multi-label learning method is called multi-label TC. Asian languages-based TC, like Chinese languages, differs from research for those other languages that employ gaps to isolated words, like English. Before categorizing text, a word segmentation procedure must be performed to turn a consistent language into a collection of discrete words, which must then be converted into vectors of a specific dimension. In principle, there are two types of multi-label learning methods: adapted techniques and problem transformation techniques. As a result, Jiahui He et al. [23] utilized customer feedback about certain hotels as a training database, which includes labels for hotel evaluation in all aspects to evaluate and compare several multi-label learning methods performances on Chinese TC. The experimental outcomes indicated that the presented SVM had attained improved performance than Random Forest (RF), CNN, and KNN. However, the RF model is not applicable for classification in this research.

With the advent of multi-label databases, multi-label learning procedures have sparked attention and are now used in various sectors. While single-label learning, examples in such learning procedures have several class labels simultaneously. Furthermore, multi-label education struggles from the dimensions constraint, making feature selection problematic. As a result, Mohsen Paniri et al. [24] have introduced MLACO; a new multi-label significance features selecting strategy predicated on Ant colony optimization (ACO). The standardized cosine resemblance between characteristics and category labels was employed as the starting pheromones of every ant to allow faster the algorithm's completion. The suggested mode is a filter-based approach because it does not use any learning techniques. The experimental outcomes on various commonly used databases show that the MLACO outperforms the competition regarding multi-label assessment criteria and execution time. However, the efficiency of MLACO is less that needs improvement.

The Marathi language is one of India's most widely spoken languages. Maharashtra residents primarily talk about it. Language usage on digital sites has expanded dramatically during the last decade. Yet, there hasn't been much focus on NLP techniques for Marathi texts. As a result, Atharva Kulkarni et al. [25] set out to present a complete overview of the resources and methods accessible for Marathi TC. In addition, two publicly accessible Marathi TC databases were used to compare LSTM, CNN, BERT, and ULMFiT predicated models. The results showed that the ULMFiT algorithm

outperforms basic LSTM- and CNN-based methods. The runtime, on the other hand, is quite long.

TC is becoming increasingly important since the volume of information on the internet is developing rapidly. Because this material needs to be more organized, it will require to be transformed. Because these papers are digital, they must be classified by automatically sending a group of them to specified labels depending on their contents. Keyword discovery techniques predicated on largely supervised classification algorithms have been presented to reduce the growing influence of news TC. However, in practice, the available data is mainly unlabeled, requiring a semi-supervised learning approach. As a result, D Naga Sudha and Y Madhavee Latha [26] tested the efficacy of a semi-supervised technique for Telugu news stories. The proposed approach analyses various ML approaches for classifying Telugu media articles and finds that SVM performs well with a greater classification rate. Moreover, the classification rate is lesser than in other models.

In recent years, NLP and, in particular, natural language-based text analysis have made significant progress. The use of DL in text analysis has changed text processing approaches and produced astonishing outcomes. Many DL approaches, such as LSTM, CNN, and the more modern Transformer, were employed on NLP applications to obtain state-of-the-art effects. As a result, Ramchandra Joshi et al. [27] examined various DL structures for TC applications. The work focuses on the categorization of Hindi texts in particular. Owing to the unavailability of a huge labeled corpus, the investigation into the categorization of semantically low and rich resources in Hindi expressed in Devanagari script was already restricted. In addition, systems predicated on LSTM, CNN, and Attention have been assessed using translated copies of English databases. Compared to BERT, the LASER multidisciplinary approach retrieved more detailed sentence descriptions. Overall, specially trained methods on particular datasets outperformed lightweight systems that used phrase encodings explicitly. However, the accuracies of the employed DL models are less than other approaches.

Several digital Hindi text papers are produced daily at news portals, Government sites, and public and commercial industries in the current world. They must be successfully categorized into numerous independently incompatible pre-defined classifications. As a result, many Hindi text-based computation processes exist in the implementation realms of IR, text summarization, keyword extraction, machine translation, simplification, and other connected linguistic and parsing viewpoints. Still, there is much room to categorize the retrieved text of Hindi files into predetermined classes using classifiers. As a result, Shalini Puri and Satya Prakash Singh [28] presented a Hindi TC (HTC) framework that takes a set of existing Hindi texts, pre-processes those at the sentence, text, and word stage, excerpts characteristics, and learns SVM classifiers that further categorize a group of unknown Hindi text documents. The studies were carried out on a sequence of four Hindi texts divided into two categories, each of which was categorized with 100% reliability using SVM. Moreover, this method is only applicable to smaller documents.

Multiple labels are assigned to every document simultaneously in multi-label categorization. In several real-world categorization tasks, high-dimensional based label spaces are used, which could be organically arranged in a hierarchy. Every occurrence in this issue may correspond to numerous labels structured in a hierarchical system. Because the categorization algorithm must allow for hierarchical connections between categories and be possible to forecast multiple tags for similar instances, it is more complex than flat categorization. Only a few researchers have looked into multi-label TC in Arabic. Most of this research has concentrated on flat categorization, ignoring the hierarchy organization. As a result, Nawal Aljedani et al. [29] investigated hierarchical multi-label categorization in the Arabic language. With an ML method, it offers a Hierarchical Multi-label Arabic TC (HMATC) framework. The influence of feature assortment techniques and feature set characteristics on classifier performance is also examined. The outcomes showed that the suggested model beats all other methods included in the trials in terms of computing cost, consuming less than the other methods. Moreover, this method applies to smaller databases. Further, the overview of reviewed literature is shown in table.2.

Table 2. Overview of the reviewed literature

Authors	Reference	Year	Article title	Proposed approach	Algorithm employed	Learning type	Label type	The language used for TC	Limitation
Mouhoub Belazzoug et	[20]	2020	An improved SCA for selecting	ISCA[1]	Naïve Bayes, GA[2], ACO[3],	ML	Multi-label	English	This method can't apply to solving

[1] Improved Sine-Cosine Algorithm
[2] Genetic Algorithm
[3] Ant-Colony Optimization

al.			features for TC		MFO[4]				complex optimization issues.
M. Alhawarat and Ahmad O. Aseeri et al.	[21]	2020	Superior Arabic TC deep model	SATCDM[5]	n-gram, multi-kernel CNN	ML	Multi-label	Arabic	The computational time is high because of employing higher databases.
Ashraf Elnagar et al.	[22]	2020	Arabic TC using DL techniques	CNN+RNN	LSTM[6], RNN[7], HAN[8], CNN, GRU[9], BiRNN[10]	DL	Multi- and Single-label	Arabic	The accuracy of the NADiA database is less
Jiahui He et al.	[23]	2019	Chinese comments classification: Multi-label identification methods comparison	SVM, RF, KNN	Logistic regression, RFBoost	ML	Multi-label	Chinese	The RF method does not apply to the classification.
Mohsen Paniri et.al	[24]	2020	MLACO: Multi-label-based feature selection approach predicated on ACO	MLACO	ML-KNN	ML	Multi-label	English	The presented methods efficiency is less
Atharva Kulkarni et al.	[25]	2022	Experimental assessment of DL approaches for Marathi TC	LSTM, BERT[11], ULMFiT[12], and CNN	NLP	DL	N/A	Marathi	Computational time is high
Naga Sudha and Madhavee Latha	[26]	2021	Semi-supervised based Multi-TC for Telugu texts	Logistic regression, SVM, and Naïve Bayes	ANN[13], n-gram	ML	Single-label	Telugu	The classification rate is less compared to other ML methods.
Ramchandra Joshi et al.	[27]	2019	DL for Hindi TC: A comparison	LSTM, LASER, BERT	CNN, LSTM, Bi-LSTM+CNN	DL	Single-label	Hindi	The accuracy of the employed model is less
Shalini and Satya Prakash	[28]	2019	An effective Hindi TC method by SVM	HTC-SVM	SVM	ML	N/A	Hindi	The developed model applies to smaller documents.
Nawal Aljedani et al.	[29]	2021	HMATC: using ML	HMATC	*k*-means random clustering	ML	Multi-label	Arabic	Only applicable to smaller databases

3 Dataset description

This study gathered three separate databases from well-known news websites (alkhaleej.ae, akhbarona.com, and

[4] Moth-Flame Optimizer
[5] Superior Arabic TC Deep-Model
[6] Long Short-Term Memory
[7] Recurrent Neural Network
[8] Hierarchical Attention Network
[9] Gated-Recurrent Unit
[10] Bi-directional RNN
[11] Bi-directional Encoder-Representations from Transformers
[12] Universal Language-Model Fine-Tuning
[13] Artificial Neural Network

alarabiya.net). Except for alarabiya.net, which lacks (Religion or culture) classifications, all databases have the sorts (Finance, Culture, Medical, Religion, Politics, Sports, and Technology). There have been no dialects in such databases because they were acquired from news websites, and the articles were written in MSA. We aggregated all databases into a single corpus termed SANAD because they were all annotated with single labels. The datasets are divided into testing and training sets, represented in table.3, which describes the number of categories and articles in each. To maintain a fair distribution of classes, the testing and training databases employed in this study are SANAD database subsets. The reports were pre-processed to give a pure form after eliminating the punctuation marks and Latin alphabet; spelling errors weren't corrected. In this research, 80% of data are employed for training, and the rest, 20%, are used for testing.

Table 3. SANAD categories and articles balanced subset's count per database

Source	Category name	Categories	Total	Testing	Training	Per category
alkhaleej.ae	Finance	7	6486	1297	5189	6500
	Culture		6488	1298	5190	
	Medical		6486	1297	5189	
	Sports		6484	1297	5187	
	Religion		6486	1297	5189	
	Technology		6484	1297	5187	
	Politics		6486	1297	5189	
Total			45400	9080	36320	
alarabiya.net	Finance	5	3900	780	3120	3800
	Medical		3900	780	3120	
	Sports		3900	780	3120	
	Technology		3900	780	3120	
	Politics		3900	780	3120	
Total			19500	3900	15600	
akhbarona.com	Finance	7	6555	1311	5244	6600
	Culture		6555	1311	5244	
	Medical		6561	1312	5249	
	Sports		6561	1312	5249	
	Religion		6555	1311	5244	
	Technology		6555	1311	5244	
	Politics		6558	1312	5246	
Total			45900	9180	36720	

3.1 alkhaleej.ae

From 2008 to 2020, they accumulated roughly 1.2 million (4GB) articles by reviewing every piece on their website. On the other hand, the site's classification could have been more comprehensive and precise. As a result, we had to individually classify a decent number of articles in every one of the seven categories stated above, a total of less than 46k articles. As a result, we had to deliberate specific articles belonging to a class toward which they didn't quite fit, making the database-less trustworthy than the other two sites. The manual classification of the Khaleej database relates to the conventional procedure of tags available on the website and grouping them into one of the seven categories listed above.

3.2 alarabiya.net

This research exited through each article in the primary site and its sub-domains, such as Aswaq and Ahadath, for the AlArabiya database. After then, all of the articles were divided into seven categories. The two categories (Culture and Iran News) lacked sufficient data compared to other websites. The "Iran News" category blended with the "Politics" topic, resulting in an excellent training database. Consequently, the number of existing categories has been reduced to five after removing the “Culture” class. The items gathered are current as of early 2020.

3.3 akhbarona.com

Several articles on the topics required for research were gathered, but one topic [Religion] only had 1/2 as many as the others. To fill the gaps, we used a comparable newspaper site to gather the rest of that classification (Alanba.com).

4 Method

4.1 Text pre-processing methods

A pre-processing stage is required for text files. Tokenization, document conversion, stop-word exclusion, and stemming were common tasks in the pre-processing step. Stemming removes all suffixes and affixes from a file to obtain its root. Because the Arabic language has so many ways of encoding text, three stemming methods were used: light stemming, Khoja stemming approaches, and raw text (no stemming). Text pre-processing has also been employed to clean up the database by eliminating all non-Arabic content. While working with text gathered from the internet, this method is highly suggested. The scraping articles must then be cleaned by removing punctuation, isolated chars, elongation, qur'anic symbols, Arabic numerals, Latin letters, and other signs.

The extraction and selection of features will be the next step. The effect of text pre-processing procedures on TC is assessed in this stage, precisely the effect of employing stemming from Arabic TC. The term weigh been used to represent each text as a weighing vector in this assignment; this is also known as the bag of words approach. This investigation determines the effect of pre-processing text assignments on TC, mainly stemming from term weighting on Arabic text. The term weighting methods are shown in table 1, in which the tf-idf method has been employed for this research. This method contains two terms, which are described as follows:

1. Term-Frequency (tf): It keeps track of how many times a word occurs in a document. Since each article was different in length, a word may appear in a larger text significantly more commonly than in smaller ones.

2. Inverse Document-frequency (idf): It determines a word’s importance by weighing common phrases and scaling up rare ones.

4.2 Selected ML methods for TC

In this research, three ML methods were selected for automatic TC. These techniques have been shown effective in English TC. The first method is Naïve Bayes, the second is SVM, and the third is the C4.5 algorithm. Moreover, the block diagram of the presented method is illustrated in Fig. 2.

4.3 Naïve Bayes (NB)

An ML approach known as Naïve Bayes (NB) is used in TC. The NB theorem has been used to create probabilistic classifiers called NB. In terms of TC, NB is a simple and effective method. The fact that NB is a highly scalable algorithm is one of its key features. Simply put, the NB classifier posits that some characteristics in a class have no bearing on the presence of any other characteristics. This theory has been adopted in evaluating the document class below Eqs. (1) and (2).

$$Q^* = argmax_Q P\left({}^{Q}/_{D}\right) \tag{1}$$

$$Q^* = argmax_Q P\left({}^{Q}/_{D}\right) \times \left({}^{p(Q)}/_{p(D)}\right) \tag{2}$$

Where Q is the class and D is the document, this classification method looks at the relationship between each characteristic and each class in a database, presuming that all characteristic values are independent. It evaluates each feature individually and calculates a conditional probability for the relationships between characteristic values and class. The category with the

most excellent probability score is the most significant as the anticipated class.

4.4 Support Vector Machine (SVM)

Among the popular text classifiers, the commonly used one is support vector machines (SVMs). SVMs are among the strategies used in supervised ML. SVMs employ a training technique to create a classifier which will be employed to allocate a new unknown text to one of many predetermined categories. SVMs could be utilized to classify data in both linear and non-linear ways. SVMs could also be used for both supervised and unsupervised learning. SVMs produce a hyperplane or a series of hyperplanes, which are then employed for classification. Moreover, in SVMs, the classes are in the form of a hyperplane, shown in Eq. (3).

$$S.G + b = 0 \tag{3}$$

Where S is the vector's weight, G is the input vector, and b denotes bias.

4.5 Decision Tree (C4.5)

In learning, a decision tree is a predictive framework. The primary purpose of a decision tree was to integrate a framework capable of predicting the target variables value. The variables established during the training phase have been exploited to forecast targeted variables in such a decision tree. A decision tree was among the most basic classification representations. To address categorization difficulties, a decision tree employs simple and obvious concepts. A decision tree was made up of a set of qualities in general. There are various kinds of decision trees, including ID3 and C4.5.

C4.5 is an enhancement to the ID3 decision tree-based method, which uses a top-down technique to build the tree. It makes a decision tree using the conquer-and-divide technique, which divides the input vector into local areas predicated on a distance measure. It employs information theory to choose properties for the intermediate nodes and roots. The C4.5 method constructs a decision tree by starting at the root point and repeating the procedure until leaf nodes are found. C4.5 is among the most effective data mining classifiers. C4.5 seems to be a statistical classification system. From a group of training databases, C4.5 creates a decision tree. C4.5 ranks possible tests based on gain ratio and knowledge gain. C4.5 is made up of four stages, which are detailed below:

1. Assign a feature as a root.

2. Create a branch for each value.

3. Put the database in the branch.

4. Continue the second step until all classes have a similar value.

Formulas employed in the C4.5 method are represented in Eqn (4) and (5).

$$V = \sum_{i=1}^{n} - P_i \times log_2 P_i \tag{4}$$

Where, V denotes Entropy, and P denotes the proportion of class in the output.

$$Gain\ (V, F) = S \times \sum_{i=1}^{n} \left({|V_i|}/{|V|} \right) \times V \tag{5}$$

Where V is the set of cases, F is the class attribute, $|V_i|$ is the number of cases to iteration, and $|V|$ represents the number of cases in the set.

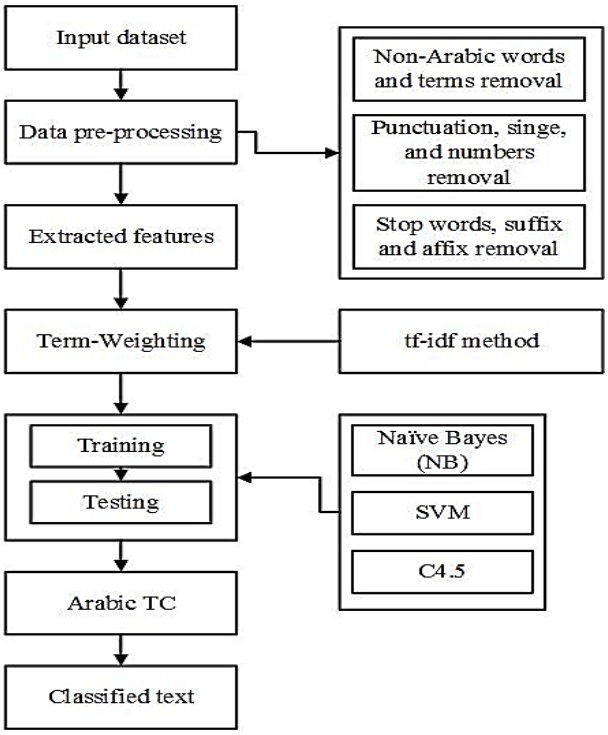


Figure 2. Presented method's block diagram

5 Evaluation metrics

The effectiveness of TC is evaluated using well-established criteria, enabling for contrast with existing approaches. Several factors, including the system's operation, influence the selection of an appropriate evaluation metric. Furthermore, evaluation measures are critical in assessing the outputs of classification models. In this study, we evaluated our outcomes using the F1 score, precision, accuracy, and recall.

5.1 Recall

The recall is described as the number of texts accurately allocated to a class divided by the overall number of texts genuinely belonging to the class. True negative (T_N) denotes the number of texts that aren't assigned to a certain class. The number of texts incorrectly assigned to a specific class is described as false negative (F_N). True positive (T_P) represents the number of texts in a given class that are appropriately identified. In contrast, false positive (F_P) indicates the number of texts incorrectly classified for a certain class.
The recall is computed by Eq. (6),

$$Recall = {}^{T_P}/_{T_P + F_N} \tag{6}$$

5.2 Accuracy

The ratio between the number of accurately classified texts and the total number of texts is known as accuracy. To assess the effectiveness of the learning techniques, accuracy has been employed as an evaluation metric for text categorization. The Accuracy metric measures the overall number of flows identified properly across all classes.
Accuracy is computed by Eq. (7),

$$Accuracy = {}^{T_P + T_N}/_{T_P + F_P + T_N + F_N} \tag{7}$$

5.3 Precision

The ratio of the number of texts accurately categorized as pertaining to class to the overall number of texts classified as about class is called precision. Precision is an important parameter of TC outcomes because it is delicate to inaccurate classification. Moreover, inaccurate categorization resulted in fewer precision results.
Precision is computed by Eq. (8)

$$Precision = {}^{T_P}/_{T_P + F_N} \quad (8)$$

5.4 F1-score

Recall and precision can be used simultaneously because large values for both metrics for a specific TC imply that the texts accurately classified the class about the total number of texts. The F1-score, Boundary F1 (BF), evaluated the recall and precision harmonic mean.
F1-score is computed by Eq. (9).

$$F1 - score = 2 \times \left({Recall \times Precision}/{Recall + Precision}\right) \quad (9)$$

6 Result

This section evaluates different pre-processing techniques and the effectiveness of ML methods in Arabic TC. The work has been completed using the Python programming language and machine configuration: CPU Speed: 3.20 GHz, Processor: Intel Core i7, RAM: 4GB, Operating System (OS): Windows 7. The lack of consistent open Arabic corpora is among the major difficulties with Arabic TC. Moreover, the majority of the text data comes from newspapers or websites. As a result, the ML algorithms' effectiveness is skewed toward such corpora, and extrapolating the frameworks to all Arabic text would've been tough. For Arabic TC, many characteristics like precision, F1-score, recall, and accuracy for three different datasets were evaluated to examine the planned work's effectiveness.

Table 4. Classification accuracy (%) of the presented model

Dataset	NB			SVM			C4.5		
	L[14]	K[15]	R[16]	L	K	R	L	K	R
Akhbarona	98.7	99.9	97.6	98.3	99.2	98.5	97.2	98.1	93.4
Khaleej	96.4	98.2	95.8	96.9	99.4	99.2	96.5	97.2	97.4
Arabiya	99.7	98.6	94.5	97.2	97.8	96.2	92.4	95.7	95.4

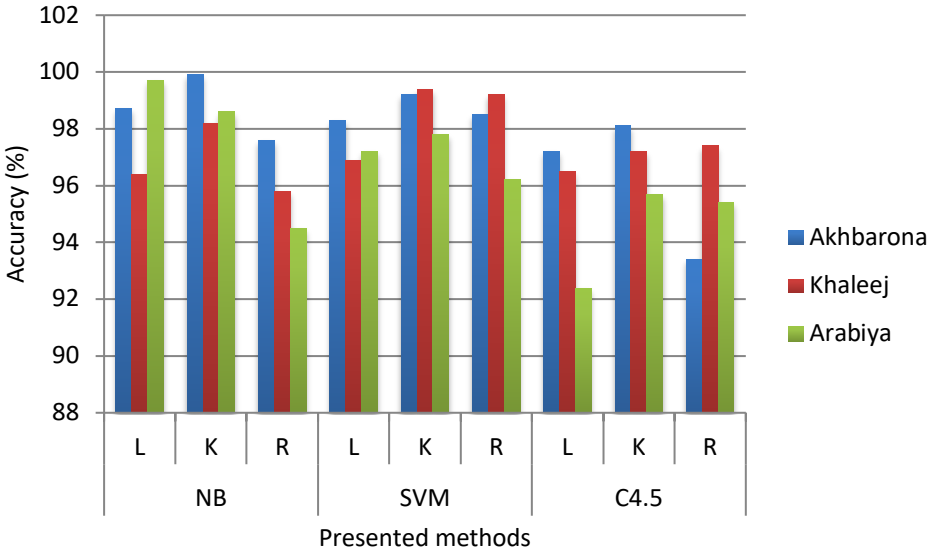
The effectiveness of the presented method is shown in tables 4 to 7; the first evaluation metric was accuracy, which is evaluated for three different datasets under three different ML methods. Table 4 illustrates the presented model's classification accuracy. The results indicated that SVM had attained higher classification accuracy than other methods. Moreover, the Arabiya database has achieved lesser accuracy than other databases. NB reached a higher classification accuracy of 99.9% for the Akhbarona database, and C4.5 attained a minimum frequency of 92.4% for the Arabiya dataset.

Table 5. Precision (%) for the presented method

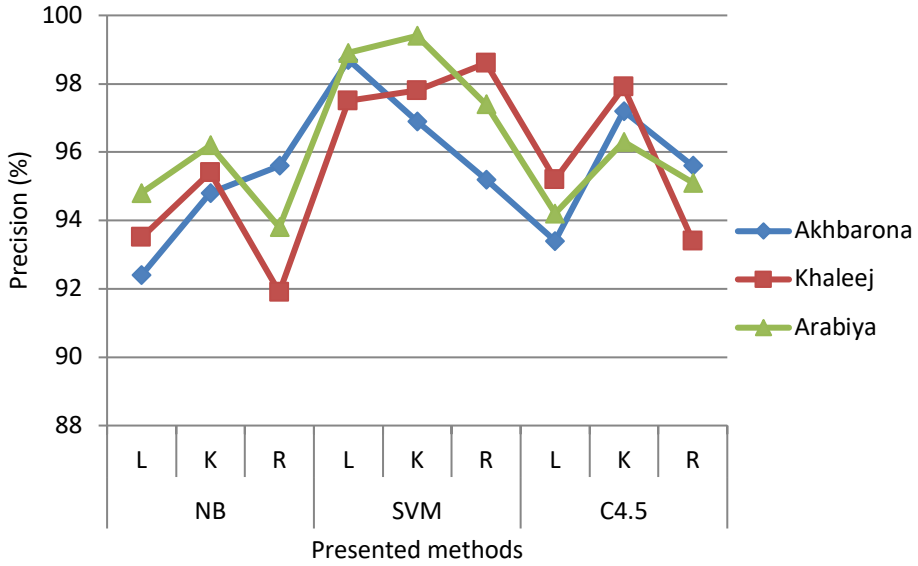
Dataset	NB			SVM			C4.5		
	L	K	R	L	K	R	L	K	R
Akhbarona	92.4	94.8	95.6	98.7	96.9	95.2	93.4	97.2	95.6
Khaleej	93.5	95.4	91.9	97.5	97.8	98.6	95.2	97.9	93.4
Arabiya	94.8	96.2	93.8	98.9	99.4	97.4	94.2	96.3	95.1

Table 5 illustrates the precision for the presented NB, SVM, and C4.5 methods for three different datasets. The result demonstrated that SVM had attained a maximum accuracy of 99.4% for the Arabiya dataset and minimum accuracy of 95.2% for the Akhbarona dataset. Moreover, a less precision of 91.9% was attained by NB for the Khaleej database. The NB and C4.5 methods achieved less accuracy compared to the SVM method.

[14] Light stemming
[15] Khoja stemming
[16] Raw-text



(a)



(b)

Figure 3. Evaluation metrics (a) Accuracy (b) Precision

The obtained accuracy and precision for the presented SVM, NB, and C4.5 method for three databases are shown in Fig. 3 (a) and (b), respectively. The result indicated that the SVM had attained higher performance in both evaluation metrics, i.e., accuracy and precision. NB also achieved higher precision and accuracy than the C4.5 method. Therefore, it is known that SVM is suitable for classifying Arabic text with higher accuracy.

Table 6. F1-score (%) for the presented method

Dataset	NB			SVM			C4.5		
	L	K	R	L	K	R	L	K	R
Akhbarona	95.6	96.8	94.9	97.2	98.4	97.3	96.4	97.5	92.2
Khaleej	96.2	97.4	96.2	95.7	99.7	98.8	97.9	97.5	96.6
Arabiya	95.7	94.9	92.8	95.4	96.4	95.5	93.4	95.2	93.7

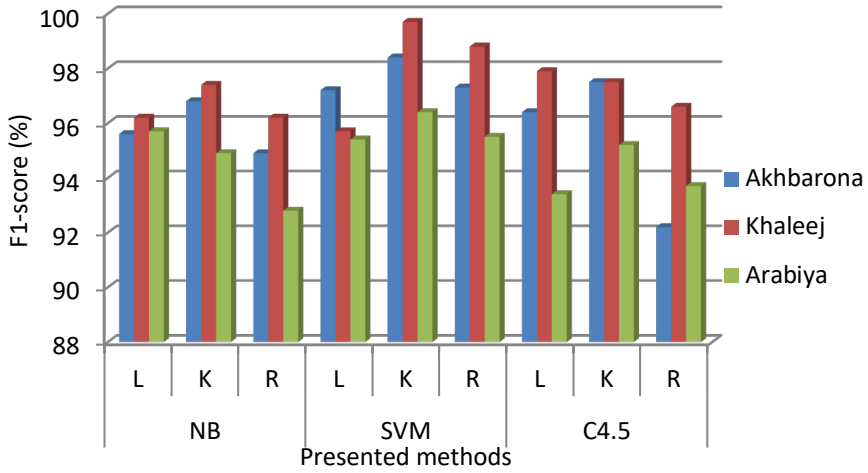
The attained F1-score and recall for the presented method are shown in tables 6 and 7, respectively. The F1-score is higher for the SVM method for the Khaleej dataset and less for the C4.5 method for the Akhbarona dataset. The maximum F1-

score attained is 99.7%, and the minimum F1-score attained is 92.2%.

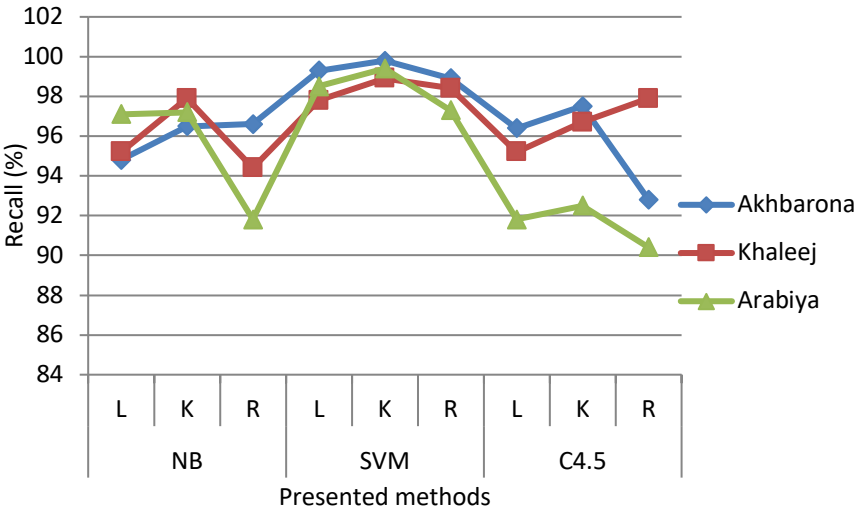
Table 7. Recall (%) for the presented method

Dataset	NB			SVM			C4.5		
	L	K	R	L	K	R	L	K	R
Akhbarona	94.8	96.5	96.6	99.3	99.8	98.9	96.4	97.5	92.8
Khaleej	95.2	97.9	94.4	97.8	98.9	98.4	95.2	96.7	97.9
Arabiya	97.1	97.2	91.8	98.5	99.4	97.3	91.8	92.5	90.4

The outcome of the presented method indicates that the SVM method has attained higher recall for the Akhbarona dataset, while the NB and C4.5 method has attained the required recall, but it is less than the SVM method. The maximum recall attained is 99.8%, which is higher than all other values and the minimum recall attained is 90.4%, which is less than all other measured values. Figure 4 (P) and (Q) indicate the attained F1-score and recall, respectively.



(P)



(Q)

Figure 4. Evaluation metrics (P) F1-score (Q) Recall

7 Conclusion

Arabic TC has been regarded as among the most difficult challenges to solve using ML methods. The pre-processing approaches used to generate the database are critical to attaining good reliability in Arabic TC. Thus, in this paper, the effect of a pre-processing technique on three ML algorithm's performance was investigated. Moreover, SANAD is

designed in this research for single-label Arabic text, which contains three datasets, namely Akhbarona, Khaleej, and Arabiya. The Akhbarona and Arabiya are imbalanced databases, while Khaleej is a balanced database. C4.5, Support Vector Machine (SVM), and Naïve Bayes (NB) are used to classify Arabic text in this research. TC was made more accessible with the use of ML algorithms. The best outcome was discovered while employing SVM and NB to compare the results of recall, F1-score, accuracy, and precision for all specified datasets. However, the datasets used in this research are fewer. Thus, in the future, the Arabic TC will be performed by Deep Learning (DL) methods to enhance the performance and to utilize higher datasets.

References

[1] B. Al-Salemi, S. A. Mohd Noah, and M. J. Ab Aziz, RFBoost: An improved multi-label boosting algorithm and its application to text categorisation, *Knowledge-Based Systems*, **103**, 104-117, 2016.

[2] M. Elloumi, M. A. Ahmad, A. H. Samak, A. M. Al-Sharafi, D. Kihara, and A. I. Taloba, Error correction algorithms in non-null aspheric testing next generation sequencing data, *Alexandria Engineering Journal*, **61(12),** 9819-9829, 2022.

[3] C. Saranyajothi and D. T. Thenmozhi, Machine Learning approach to Document Classification using Concept based Features, *International Journal of Computer Applications*, **118(20),** 33-36, 2015.

[4] A. El-Komy, O. R. Shahin, R. M. Abd El-Aziz and A. I. Taloba, Integration of computer vision and natural language processing in multimedia robotics application, *Information Sciences Letter*, **11(3)**, 765-775, 2022.

[5] M. Jiang, Y. Liang, X. Feng, X. Fan, Z. Pei, Y. Xue, and R. Guan, Text classification based on deep belief network and softmax regression, *Neural Computing and Applications*, **29(1)**, 61-70, 2018.

[6] A. A. Sewisy, M. H. Marghny, and A. I. Taloba, Fast Efficient Clustering Algorithm for Balanced Data, *International Journal of Advanced Computer Science and Applications*, **5(6)**, 123-129, 2014.

[7] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes and D. Brown, Text classification algorithms: A survey, *Information*, **10(4)**, 150, 2019.

[8] J. Huang and C. X. Ling, Using AUC and accuracy in evaluating learning algorithms, *IEEE Transactions on knowledge and Data Engineering*, **17(3)**, 299-310, 2005.

[9] F. Joseph and N. Ramakrishnan, Text categorization using improved K nearest neighbor algorithm, *International Journal for Trends in Engineering and Technology*, **4**, 65-68, 2015.

[10] K. Horecki, and J. Mazurkiewicz, Natural language processing methods used for automatic prediction mechanism of related phenomenon, *International Conference on Artificial Intelligence and Soft Computing*, Springer, Cham, 13-24, 2015.

[11] S. S. Ismail, R. F. Mansour, R. M. Abd ElAziz and A. I. Taloba, Efficient E-Mail Spam Detection Strategy Using Genetic Decision Tree Processing with NLP Features, *Computational Intelligence and Neuroscience*, 2022.

[12] S. Lai, L. Xu, K. Liu, and J. Zhao, *Recurrent convolutional neural networks for text classification*, Twenty-ninth AAAI conference on artificial intelligence, **29(1)**, 2267- 2273, 2015.

[13] V. Bijalwan, V. Kumar, P. Kumari, and J. Pascual, KNN based machine learning approach for text and document mining, *International Journal of Database Theory and Application*, **7(1)**, 61-70, 2014.

[14] Z. Yan, and C. Xu, *Combining KNN algorithm and other classifiers*, 9th IEEE International Conference on Cognitive Informatics, IEEE, 800-805, 2010.

[15] A. Rayan, A. I. Taloba, R. M. Abd ElAziz and A. Abozeid, IoT enabled secured fog-based cloud server management using task prioritization strategies, *International Journal of Advanced Research in Engineering and Technology*, **11(9)**, 697-708, 2020.

[16] D. Sharma and M. E. Cse, Stemming algorithms: a comparative study and their analysis, *International Journal of Applied Information Systems*, **4(3)**, 7-12, 2012.

[17] A. Elhadad, F. Alanazi, A. I. Taloba, and A. Abozeid, Fog Computing Service in the Healthcare Monitoring System for Managing the Real-Time Notification, *Journal of Healthcare Engineering*, 2022.

[18] Y. Saeys, I. Inza and P. Larranaga, A review of feature selection techniques in bioinformatics, *Bioinformatics*, **23(19)**, 2507-2517, 2007.

[19] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnork and D. Keim, *Combining automated analysis and visualization techniques for effective exploration of high-dimensional data*, In 2009 IEEE

Symposium on Visual Analytics Science and Technology, 59-66, 2009.

[20] M. Belazzoug, M. Touahria, F. Nouioua and M. Brahim, An improved sine cosine algorithm to select features for text categorization, *Journal of King Saud University-Computer and Information Sciences*, **32(4)**, 454-464, 2020.

[21] M. Alhawarat and A. O. Aseeri, A superior Arabic text categorization deep model (SATCDM), *IEEE Access*, **8**, 24653-24661, 2020.

[22] A. Elnagar, R. Al-Debsi and O. Einea, Arabic text classification using deep learning models, *Information Processing & Management*, **57(1)**, 102-121, 2020.

[23] J. He, C. Wang, H. Wu, L. Yan and C. Lu, Multi-Label Chinese Comments Categorization: Comparison of Multi-Label Learning Algorithms, *Journal of New Media*, **1(2)**, 51-61, 2019.

[24] M. Paniri, M. B. Dowlatshahi and H. Nezamabadi, MLACO: A multi-label feature selection algorithm based on ant colony optimization, *Knowledge-Based Systems*, **192**, 105285, 2020.

[25] A. Kulkarni, M. Mandhane, M. Likhitkar, G. Kshirsagar, J. Jagdale and R. Joshi, *Experimental evaluation of deep learning models for marathi text classification*, In Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications, 605-613, 2022.

[26] D. N. Sudha, Semi Supervised Multi Text Classifications for Telugu Documents, *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, **12(12)**, 644-648, 2021.

[27] O. R. Shahin, H. H. Alshammari, A. I. Taloba and R. M. Abd El-Aziz, Machine Learning Approach for Autonomous Detection and Classification of COVID-19 Virus, *Computers and Electrical Engineering*, **101**, 108055, 2022.

[28] A. I. Taloba, R. M. Abd El-Aziz, H. M. Alshanbari and A. A. H. El-Bagoury, Estimation and Prediction of Hospitalization and Medical Care Costs Using Regression in Machine Learning, *Journal of Healthcare Engineering*, 2022.

[29] N. Aljedani, R. Alotaibi and M. Taileb, Hmatc: Hierarchical multi-label arabic text classification model using machine learning, *Egyptian Informatics Journal*, **22(3),** 225-237, 2021.