IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 15 June 2023, accepted 30 June 2023, date of publication 4 July 2023, date of current version 10 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3292248

RESEARCH ARTICLE

Optimizing Document Classification: Unleashing the Power of Genetic Algorithms

GHULAM MUSTAFA[1], ABID RAUF[2], AHMAD SAMI AL-SHAMAYLEH[3], MUHAMMAD SULAIMAN[4], WAGDI ALRAWAGFEH[5], MUHAMMAD TANVIR AFZAL[1], AND ADNAN AKHUNZADA[5], (Senior Member, IEEE)

[1]Department of Computing, Shifa Tameer-e-Millat University, Islamabad 44000, Pakistan
[2]Department of Computer Science, University of Engineering and Technology, Taxila 47080, Pakistan
[3]Department of Network and Cybersecurity, Faculty of Information Technology, Ah-Ahliyya Amman University, Amman 19328, Jordan
[4]Department of Computer Science, University of Stavenger, 9990 Stavenger, Norway
[5]College of Computing and IT, University of Doha for Science and Technology, Doha, Qatar

Corresponding author: Wagdi Alrawagfeh (wagdi.alrawagfeh@udst.edu.qa)

We extend our heartfelt gratitude and appreciation to Qatar National Library for their generous support in providing Open Access funding for this research.

ABSTRACT Many individuals, including researchers, professors, and students, encounter difficulties when searching for scholarly documents, papers, and journals within a specific domain. Consequently, scholars have begun to focus on document classification problem, offering various methods to address this issue. Researchers have utilized diverse data sources, such as citations, metadata, content, and hybrids, in their approaches.In these sources, the meta-data-based approach stands out for research paper classification due to its availability at no cost. Various scholars have employed different metadata parameters of research articles, including the title, abstract, keywords, and general terms, for research paper classification. In this study, we chose four meta-data-based features such as, title, keyword, abstract, and general terms from the SANTOS dataset, which was prepared by ACM. To represent these features numerically, we employed a semantic-based model called BERT instead of the commonly used count-based models. BERT generates a 768-dimensional vector for each record, which introduces significant time complexity during computation. Additionally, our proposed model optimizes the features using a genetic algorithm. Optimal feature selection performances a crucial role in this domain, enhancing the overall accuracy of the document classification system while reducing the time complexity associated with selecting the most relevant features from this large-dimensional space. For classification purposes, we employed GNB and SVM classifiers. The outcomes of our study exposed that the combination of title and keywords outperformed other combinations.

INDEX TERMS Document classification (DC), Word2Vector (W2V), bag of word (BOW), term frequency (TF), association for computing machinery (ACM), machine learning (ML).

I. INTRODUCTION

From the past couple of years' research glut on the web is rapidly dilated. This massive volume makes it difficult for recommender systems to find suitable research articles for user-posed queries. Aside from that, classification of research publications has grabbed the scientific community's attention [1]. This research article classification could be facilitated in a type of ways, including assisting scholars and professors in 1) identifying relevant research articles and 2) finding appropriate articles to explain the background concept of the proposed study etc.

Different repositories, such as Google Scholar and Digital Libraries, are used by most users to find relevant research articles. However, currently, the data present on the web are unstructured in nature, which hampers the process of finding appropriate data against a user-posed query. As mentioned, repositories return millions of generic hits. For understanding, lets consider the scenario in which, when we pose a query for research paper recommendation on Google Scholar, 2.8 million papers were displayed. If a user is an expert researcher, reading an average of five papers per day will take

The associate editor coordinating the review of this manuscript and approving it for publication was Chong Leong Gan.

approximately 158 years for that user, which is impossible for him. The above result is due to the fact that papers are not categorized or indexed according to their appropriate classifications scheme like ACM. We believe that if these systems are categorized consistently, their performance will improve. If we can classify papers, this will be helpful in the cases discussed above.

Many strategies for classifying research articles have been presented in the literature. These techniques are characterized as citation, metadata, content-based, or hybrid techniques [2], [3], [4], [5], [6]. Metadata based approaches are important among them due to its nature (always free available online). Research scholars have utilized various metadata parameters, such as title, keyword, abstract, and general terms, individually, as well as their combination in research paper classification techniques [7], [8], [9].

In text mining the representation of the text is one of the core issue [10]. The purpose of this is to numerically change the unstructured text input into documents that can be quantified statistically. In the literature, state-of-the-art techniques utilize traditional count-based approaches such as 1) BOW, 2) TF, and 3) TFIDF [6], [7], [8], [11], [12], [13]. As a result, these techniques have disregarded the semantic and contextual information contained in words, leading to inaccurate classification of research articles. After 2013, some semantic-based techniques have been proposed by different researchers such as 1) Word2Vec [14], [23], 2) Glove [15], 3) Fasttext [16], and 4) BERT [17]. These techniques can recognize the context of words in a research article, such as semantic and grammatical similarities, as well as correlations with other words. Owing to the increasing use of these techniques by researchers in different domains, the document classification community started the utilization of these techniques in their studies [14], [18], [19], [20] which presented promising results. One of the issues related to these techniques is the large length of the vector generated against a single word in a text. For example, W2V generated a vector of 300 length against a single word, similar to BERT, which generated a vector of 768 length against a single word, and so on. Such a large length requires too much time to perform a classification activity.

Therefore, we used the BERT algorithm for text representation because of its bidirectional nature, which captures semantic and contextual information of the term. Therefore, we used a machine learning model for feature optimization because, with more features, its complexity increases. Optimal feature selection can perform an essential role in this field and improve the overall accuracy of the document classification system. One of the most modern algorithms for feature selection is the genetic algorithm. This followed the natural phenomenon of biological evolution. This is a stochastic method for feature optimization. Organisms have genes that evolve over time or successive generations. By evolving, they can adapt better to the environment. It is a heuristic optimization method inspired by natural evolution procedures. The population of individuals was created using this algorithm. A new population is created after every generation by selecting individuals who fit in the problem domain. Then, the individuals are recombined, and different operations are performed using different operators, such as mutation and crossover.

TABLE 1. Categories description.

Category	Description
A	General Literature
B	Hardware
C	Computer System Organization
D	Software
E	Data
F	Theory of Computation
G	Mathematics of Computing
H	Information Systems
I	Computing Methodologies
J	Computer Application
K	Computing Milieux

For experimental purposes, this study used freely available metadata parameters and a combination of research articles such as 1) title, 2) abstract, 3) keywords and 4) General terms. For this, we used the ACM dataset developed by Radrigues and Santos et al. [12]. The dataset contains metadata parameters of research documents in the field of computer science. From this set of metadata parameters, we selected four metadata parameters (1) title, 2) abstract, 3) general terms and 4) keywords). To represent our dataset text in numeric form, we employed the mostly used semantic-based technique, Word2Vec (explained in the methodology section). Similar to [5], [6], and [47], we also used the ACM categorization system to classify the research articles by assigning a suitable label. The ACM Computing Classification System (CCS) is a classification scheme developed by the Association for Computing Machinery (ACM), which is utilized by different ACM journals to organize subjects by area. ACM CCS is divided into three levels. This study utilized the top-level categories (presented in Table 1) for the classification of research articles.

For the research article classification task, we used the SVM and Naive Bayes classifiers (explained in the Methodology section). From the results of our experiments, we achieved a 0.83 percent classification accuracy for title and keyword combinations. Moreover, we found that by increasing the generation in genetic algorithm the average accuracy also increases.

The rest of the manuscript is structured as follows. Section II elaborates on the state-of-the-art techniques proposed in the research article's classification domain. Section III describes the proposed method. Section IV explains the results of the experiment in detail. Finally, Section V presents the conclusions of the study.

II. LITERATURE REVIEW

In the literature review section, we explain existing techniques proposed by different researchers in the document classification domain. These state-of-the-art approaches in

the literature can be divided into two broad categories: 1) content-based features and 2) metadata-based features.

Content-based approaches have mostly focused on the overall Content of the research article and contain an introduction, headings, methodology, and conclusions that are not freely available. While the metadata based approaches have focused on metadata of the research article and it contain title, keywords, author name etc, which are available freely. This is why researchers mostly move towards metadata features instead of content-based features because of the subscription requirement.

H. Nanba et al. [21] proposed a technique for research article classification that utilizes citation links and type-based features. The authors developed the PRESRI tool based on this study to classify research papers. This tool uses the words of the title and the authors 'names as features. The tool takes the features as input and classifies the research article based on the cited article mentioned in the reference section of the query paper.

Taheriyan et al [22], have presented an approach for classification of research document based on analysis of their interrelationship. The authors have utilized common reference-based parameters, common authors, and citations in their study. Using this, a relationship graph was created in which nodes represent research papers, and linkages between those nodes establish the relationship between research articles. The results revealed that the results were very good in the case of dense and close-packed graphs.

Some researchers [23], [24], [25], have utilized the bibliography section of the research article to identify the category of a paper. These studies have focused on the assumption that most researchers cite articles of similar domains or categories. To prove this claim, researchers have utilized a dataset collected from the Journal of UCS. The authors matched the references stored in the database with the extracted references of the research articles.

In another study, the authors presented a Bayesian-based approach for research article classification [2]. For the experiment, the author utilized 400 research articles collected from conferences based on education as a dataset and categorized them into four categories: 1) cognition issues, 2) e-learning, 3) teacher instruction, and 4) intelligent coaching systems. The authors revealed that keyword metadata can be used for classifying research articles. The limitation of this approach is its dependence solely on the keywords. In another study [26], researchers classified articles into more than one class based on phrase-to-text connectedness. They utilized three several evaluation measures to evaluate their proposed technique: 1) the well-known characteristic of the likelihood of term generation BM25, 2) cosine relevance score between typical vector area representations of the texts coded with tf-idf weighting, and 3) an in-house characteristic of the conditional probability of symbols averaged over matching fragments in suffix trees representing texts and phrases, CPAMF. Furthermore, the authors collected abstracts of research articles from the ACM Digital Library for experimental purposes. The results of their experiments revealed that the CPAMF results were better than those of the cosine measure and BM25 by a healthy margin. For research article categorization, some authors have proposed hybrid approaches [19], [27], [28], [29], [30]. In these approaches, feature extraction is performed utilizing DL techniques and classification based on ML and DL methods. The outcome of these approaches is excellent when utilizing the entire content of the research articles. In the case of research article classification, Balys and Rudzkis [31] presented a study on the automatic classification of research articles using applied mathematical analysis of probabilistic distributions of the scientific terms in texts. Moreover, some works, such as [32], focused on ML algorithms to develop subject classification rules for research article classification. However, such approaches used the entire content of the research documents to extract the features and developed a classifier, which is a time-consuming task [33]. This study [34] utilized the entire content of the article and proposed an approach based on the logistic regression and naive Bayes algorithms. For the experiments, they employed two datasets from the computer science area, which have already been labeled as CiteSeerX and arXiv. The outcome of the study revealed that F1 Score on the arXiv and the CiteSeerX datasets are 0.95 & 0.75 respectively. Luo [35] employed a support vector machine model for categorizing English texts in articles. The authors performed several analytical experiments to verify the selected classifiers using English documents. They employed a dataset comprising 1033 text documents. The results of the experiment revealed that the Rocchio classifier reported good results when the size of the feature set was small, and the SVM outperformed the others. Furthermore, they observed that, as the feature set increased to 4000, the classification rate exceeded 90%.

Lai at el [36] presented a technique for categorizing text based on Recurrent Convolutional Neural Networks. This approach uses a recurrent structure to obtain contextual information during learning. To capture contextual information, they used a recurrent structure model to learn word representations. The approach also employs a max-pooling layer that identifies important words for text classification. The approach also used a pre-trained word-embedding model. Moreover, they performed a comparison with existing approaches (RNN and CNN).The approach was evaluated using four different English and Chinese datasets: 1) 20Newsgroups, 2) ACL Anthology Network, 3) Fudan set, and 4) Stanford Sentiment Treebank. The results of the experiments show that the RCNN achieved a better score (0.96 Macro F1) than the existing approaches.

Kim and Curry [37] proposed a technique for sentence classification by utilizing the Convolutional neural network. This approach first trains a CNN with a single convolution layer on the word vectors. These vectors were obtained using the pre-trained Google news model. In this approach, they performed some tuning of the parameters of a network, which

yielded good results. Moreover, the author also discussed and evaluated different variations of models, such as static and nonstatic CNN. The approach was evaluated on six different benchmark datasets: MR, SST-1, and TREC. The outcome of the study revealed that the approach has been performed extraordinarily (0.93) on CNN static variation with little tuning in network parameters.

Zhou et al. [38] presented a study on text classification based on phrase level features. The approach combines CNN and LSTM to propose a unified C-LSTM model. First, the approach learns phrases using CNN layers; afterwards, such a dense high-level representation is provided as an input to the LSTM network to learn long-term dependencies. To evaluate the approach, they used two different datasets: Stanford Sentiment Treebank (Movie Review) and TREC (Question-type data). The outcome of the study revealed that C-LSTM (0.878 Reported on SST Dataset and 0.94 on TERC) performed extraordinarily compared with CNN (0.872 Reported on SST Dataset and 0.93 on TERC) and LSTM (0.866 Reported on SST Dataset and 0.93 on TERC) on both datasets.

Liu et al. [39] proposed an approach for multi-label text classification based on deep learning model. This approach uses a family of CNN models to propose a new model, XML-CNN. In this approach, they used a max-pooling function of the CNN model to obtain a large amount of information from different areas of a document. Moreover, the approach uses a bottleneck layer for representation and reduces the dimension size of the model. To evaluate the model, they used six different benchmark datasets (Wiki-30k, Amozon-670k, and others). The authors also used cross-entropy loss as the evaluation measure, which is suitable for multi-label classification. After a comparative evaluation, XML-CNN performed extraordinarily on all benchmark datasets.

Conneau et al. [40] proposed a text classification technique using a deep CNN. The architecture presented by this approach works on the character level of a text. For convolution and pooling operations, the author used a maximum window size of up to 3 to better understand the hierarchal representation of a sentence. For evaluation purposes, they utilized six freely available datasets. During the evaluation, they increased the depth of the architecture by increasing the number of convolutional layers. They reported that when the number of layers increased from 9 to 17 and then to 29, the results steadily improved. Moreover, they also reported that the max pooling mechanism in CNN performed better than the others. The outcome of the research shows that the proposed approach is more effective than existing approaches.

Kowsari et al. [41] have proposed a Hierarchical Deep Learning technique for text classification. The approach uses a combination of deep learning models to allow both overall and specialized learning at different levels of document hierarchy. The architecture of HDLTex for every Deep Learning model contains a DNN (eight hidden layers), RNN (GRU and LSTM have been used), and CNN (eight hidden layers with different numbers of filter sizes). Three different datasets were used to evaluate the proposed approach. After evaluating the proposed approach on these three different datasets, they reported that a combination of RNN at the upper level produces high accuracy (up to 0.94), and DNN or CNN produces at a lower level with high accuracy (up to 0.92) as compared to existing approaches (SVM, Naïve Bayes, etc.). The results of the study revealed that deep learning algorithms provide sufficient improvement in document classification.

Another hierarchical text classification framework (HR-DGCNN) was proposed by Peng et al. [42]. This approach utilizes a Deep Graph Convolutional neural network. In this approach, they first converted all the text into the form of graph-of-words, after which they used the convolution operation to convolve the word graph. Essentially, this representation captures long-distance semantics, either of which are in consecutive or nonconsecutive forms. Moreover, they demonstrated that in deep learning, we used a recursive regularization technique for large-scale hierarchical text classification. For the evaluation, they used two different RCV1 and NY Times datasets. The outcome of the study revealed that the results were comparably better than those of the existing approaches.

Lee and Dernoncourt [43] presented a technique for short text classification by utilizing the combination of RNN & CNN. This approach is composed of two main parts. In the first part, the vector is generated using either the RNN or CNN architecture for every short text. Second, they classify the current text based on the current vector representation, as well as a few preceding short text representations. To evaluate the model, they used three different datasets: DSTC 4 (Dialog State Tracking Challenge 4), MRDA (ICSI Meeting Recorder Dialog Act Corpus), and SwDA (Switchboard Dialog Act Corpus). They used different vector dimensions for different datasets, such as 300 vector dimensions (using pretrained Word2Vec) for the DSTC4 dataset and 200 for MRDA and SwDA, and with these dimension vectors, existing approaches achieve good results. After a detailed evaluation and comparison, they reported that their approach performed well on all three datasets.

After a comprehensive review of the literature, we identified the following major points:

1) Most researchers used the content of the research article due to the richness of its features.

2) Due to the unavailability of the full content of the research article, some of the researchers later moved towards the meta-data of the research article used as a feature.

3) To represent a text, researchers have utilized both count- and semantic-based techniques.

4) With an increasing number of features, its complexity increases, and thus feature optimization is an important task for document classification. In the literature, most researchers have selected either all the features or made some manual combinations of features instead of performing optimization.

Therefore, in this study, we employed a genetic algorithm for feature selection.

III. METHODOLOGY

We developed an effective methodology to address the highlighted problem comprehensively. Figure 1 shows the overall structure of the proposed methodology. Initially, we selected a comprehensive dataset of the computer science domain from ACM, prepared by Radrigues and Santos [12] in 2009. Subsequently, we extracted metadata features from the selected dataset. For the removal of noisy data from the extracted features, some basic preprocessing steps were performed on the dataset. In the next step, we converted our textual data into numeric forms. Before classification, we employed a genetic algorithm for the selection of the best features from the huge length dimension features generated during the conversion of text into numeric. Subsequently, we divided our dataset into 80:20 ratios for training and testing purposes. For classifier training we have utilized the training data for evaluating our model we have employed the test data. After the training session, we provided the input to the system in the form of a new sample of research articles, which can predict the category on the basis of a given input. Subsequently, the predicted results were compared with the actual results of the given input sample by the system, and the results were presented using various evaluation measures.

A. DATASET

Dataset selection plays a pivotal role for experimental purposes. For our study, we selected a dataset of computer science domains prepared by Radrigues and Santos [12] in 2009 and collected documents from the ACM digital library. This dataset was selected because it contained research articles belonging to different research areas in the computer science domain. This dataset will help in a comprehensive evaluation of our model. The dataset contains 11 categories at the root level: 1) General Literature (A), 2) hardware (B), 3) computer system organization (C), 4) software (D), 5) data (E), 6) TOC (F), 7) computing mathematics (G), 8) Information Systems (H), 9) computing methodologies (I), 10) computer applications (J), and 11) Computing Milieux (K). The dataset contains instances from all these categories, with a total of 86116 records. The individual record contained five metadata: tile, abstract, keywords, general terms, and classification labels. In this study, we utilized the first four metadata parameters as the feature set, and the last one was used as a classification label. In the dataset, some records belonged to more than one category. These records were converted into a single label by creating a replica with the second or third label and placing its label in the classification label column. Table 2 presents the statistics for the database.

B. FEATURES EXTRACTION

The dataset comprises various metadata features of research articles, such as 1) title, 2) abstract, 3) keywords, and 4) General Terms, which are mostly freely available. For our study, we selected these metadata parameters for the following reasons.

- The titles of research articles contain some key terms of specific domains that provide hints in finding a category of a paper.
- Generally, abstracts are not in the list of metadata parameters, but due to its free availability, we can use it with metadata parameters and it contains the main theme of paper that can assist in finding a category of paper.
- The remaining parameters, general terms, and keywords are explicitly provided by the authors, which mostly contain information regarding relevant areas. For a comprehensive evaluation of all metadata features, we also made all possible combinations of the above selected metadata features, which are given presented in the Table3.

TABLE 2. Database statistic.

Dataset Statistics	Records
Total No of records	86116
Total No of records with title attribute	86116
Total No of records with abstract attribute	53963
Total No of records with multiple categories	54994
Total No of records contain one or more keywords	23971
Total No of records contain one or more general terms	51574

C. PRE-PROCESSING

Before proceeding to the experimental phase, we transformed our dataset into a particular format. For this purpose, we applied some basic preprocessing techniques, which are described below.

D. NOISE REMOVAL

Noise removal is the initial preprocessing step. This is an inevitable problem that influences the overall process of data mining, gathering, and preparation of data, which can result in the formation of errors. There are two main sources of noise [44]: implicit errors generated by measurement tools such as various types of sensors. The second is random errors generated by batch processes or by professionals when the data are collected, such as in a document digitalization process. Random errors generated during data collection from the ACM digital library were observed. These errors can adversely affect the overall performance of the proposed technique in terms of accuracy [46]. Generally, noise is divided into two categories: class noise and attribute noise [44]. Class noise contains (contradictory and mislabeled examples), whereas attribute noise contains (Erroneous, Missing & Do not care values). In our dataset, the noise was in the form of missing values. In the literature, different techniques have been used to handle missing values, such as 1) deletion of records [37], 2) Mean Substitution [45], and 3) last observation carried forward [46]. Owing to the small number of records containing missing values, we employed the removal method as it is the easiest and most best method for the

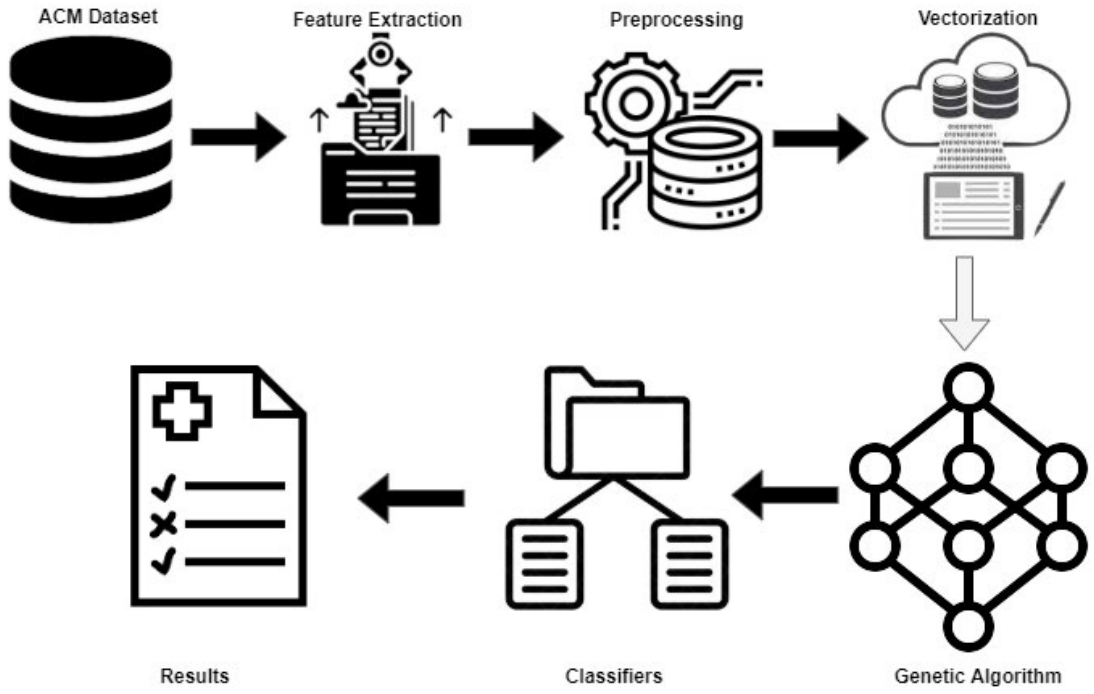

FIGURE 1. Architecture diagram.

TABLE 3. All possible features.

Dataset	Uni Features	Bi Features	Tri Features	All Features
ACM Dataset	Title, Abstract, Keywords, General Terms	Title and Abstract, Title and Keywords, Title and General terms, Abstract and Keywords, Abstract and General Terms, Keywords and General Terms	Title, Abstract and Keywords, Title, Abstract and General Terms, Title, Keywords and General Terms, Abstract, keywords and General Terms	Title, Abstract, Keywords and General Terms

handling missing value records. Therefore, by employing this technique, we deleted the records containing missing values.

E. STOP WORDS REMOVAL

Stop-word removal is one of the most important factors used for the optimization of data analytic processes. To attain good results, superfluous terms must be removed with little or no semantic relevance. To implement this, we can remove it by simply storing the words in the list that are considered strop words and compare them with the target text. However, the NLTK library in Python also provides a stop word list stored in several languages. In our case, we have utilized an English-based dictionary in the stop word list that is already defined, just matched with the target text from which stop word removal is required.

F. WORD EMBEDDING MODEL

Similarity measures, ML, and DL algorithms mostly take inputs in the form of numeric vectors. Therefore, before performing any operation, we require a method to transform our textual features into numeric vectors. Various transformation methods have been proposed in the literature. These techniques are broadly divided into two categories: count- and sematic-based. The most widely used count-based approaches in the literature are1) BOW or TF, 2) one-hot encoding, and 3) TFIDF. The limitation of these approaches is that they capture information that completely depends on the frequency of terms that occur in the document and ignore the semantic and contextual meanings of the term. Some of proposed techniques [6], [7], [11], [13], [46], [47] in literatures have utilized count based techniques such as 1) TF, 2) BOW, and 3) TFIDF, etc. for the research article classification. These techniques completely ignore the semantic and contextual meanings of the terms. Therefore, it might be possible to assign incorrect labels to research documents. To address this issue, numerous techniques that consider the semantic and contextual information of terms have been presented in the literature. From the literature, we identified

a renowned word-embedding technique employed in several domains [1], [47], [48], [49], [50]. This technique has been used to represent document vocabularies. This technique has the potential to capture the semantic and contextual meanings of terms in a research document. Word2Vec is one of the most renowned techniques for the learning word embeddings using shallow neural networks. This technique was proposed by Mikolov et al. in 2013 using Google [47]. For converting text into vectors, it reads text from only one side, that is, from left to right, which is basically the disadvantage of word2vec, which means that it reads text from left to right and not vice versa. In 2018, Jacob Devlin and his colleagues from Google proposed the technique name BERT [17] for the transformation of text into vectors. BERT represents Bidirectional Encoder Representations from Transformers. It is a machine learning technique based on transformers and is used for natural language processing (NLP). It was pre-trained using Google. One of the main advantages of using BERT is that, as opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the transformer encoder reads the entire sequence of words at once. Therefore, it is considered bidirectional, although it would be more accurate to say that it is nondirectional. This characteristic allows the model to learn the context of a word based on its surroundings (the left and right sides of the word). This model generated a vector containing 786 elements. Each record contain different number of sentences, which are further transformed into a single vector by taking the average vectors against the individual vectors of a sentence.

G. FEATURES OPTIMIZATION

In this study, GA is used for optimal feature selection in a supervised manner because the fitness value is calculated with the help of the class label. In the proposed work, the GA model is composed of state-of-the-art operators: initial population generation, fitness function, parent selection, crossover, and population creation for the next generation. The Figure 2 shows the generic steps of the GA.

1) INITIAL POPULATION

The initial population consists of a specific number of chromosomes. The number of chromosomes in the initial population and the structure of the chromosomes vary from problem to problem. In the proposed study, the number of chromosomes in the initial population was 100, and the structure of the chromosome consisted of column numbers that were randomly selected from the given number of columns. The structure of the chromosomes is shown in Figure 3.

Each component of the chromosome is called a gene. In the proposed method, the structure of chromosomes is a one-dimensional array and the number of genes in the chromosomes is 100. Each gene represents a column number of the dataset. For example, the gene at index 0 consists of eight, It represent 8th column of the preprocessed dataset. On each chromosome, none of the two genes consisted of the same column number. The Figrue 4 represent the Algorithm used to create the initial population. In Algorithm the line one create an empty chromosome. In line 2, the loop is used to iterate the population creation. One chromosome was generated in each iteration. In line 3, an empty chromosome is created to store the number of genes that are stored later in the population. Line 4 consists of a loop, which iterates until the expected number of genes for the chromosome is complete. A random gene was generated in line 5. In line 6, the randomly generated genes are compared. If a generated gene already exists in the chromosome, then the control is shifted to line 5. If the generated gene does not exist in the chromosome, it is added to the chromosome in line 7. In Line 8, a complete chromosome is added to the population.

2) FITNESS FUNCTION

The fitness function was used to determine the fitness value of each chromosome. The fitness value represents the extent to which the chromosome is fit as a solution to the given problem. In the proposed method, the accuracy is used as a fitness value for each chromosome. In the proposed method, two classifiers were used individually in the fitness function. These classifiers are the Support Vector Machine, and Gaussian Naïve Bayes. After determining the fitness value, chromosomes in the population were sorted in descending order according to their fitness values.

3) PARENT SELECTION

Tournament selection is used for the best parent selection in the population (algorithm presented in Figure 5). The parent selection procedure obtains two variables as input: the first is chromosomes along with their fitness value, and the second is the number of parents to be selected.

In Figure 5 the first line of the procedure, an empty list was created to store the selected parents. In the second line for the loop, in each iteration of the loop, one parent is selected and stored in the list. In Line 3, the first random parent is selected. In line 4, the loop is used to generate three additional chromosomes. An optimal parent, according to the fitness values, was selected from the four random parents. In the last line, the selected parent is stored on the list.

4) CROSSOVER

Crossover play an important role in diversification as a reproduction operator. In crossover, the first parent is divided in half, and the second half is directly transferred to the new chromosome. The remaining genes are selected from the second parent in such a manner that gene duplication is avoided in new chromosome. The Figure 6, 7 diagram illustrates the crossover mechanism and its algorithm respectively.

5) MUTATION

In mutation half of the chromosome from parent is directly transferred to parent chromosome, and the remaining half is completed from randomly generated gene. Genes are added to

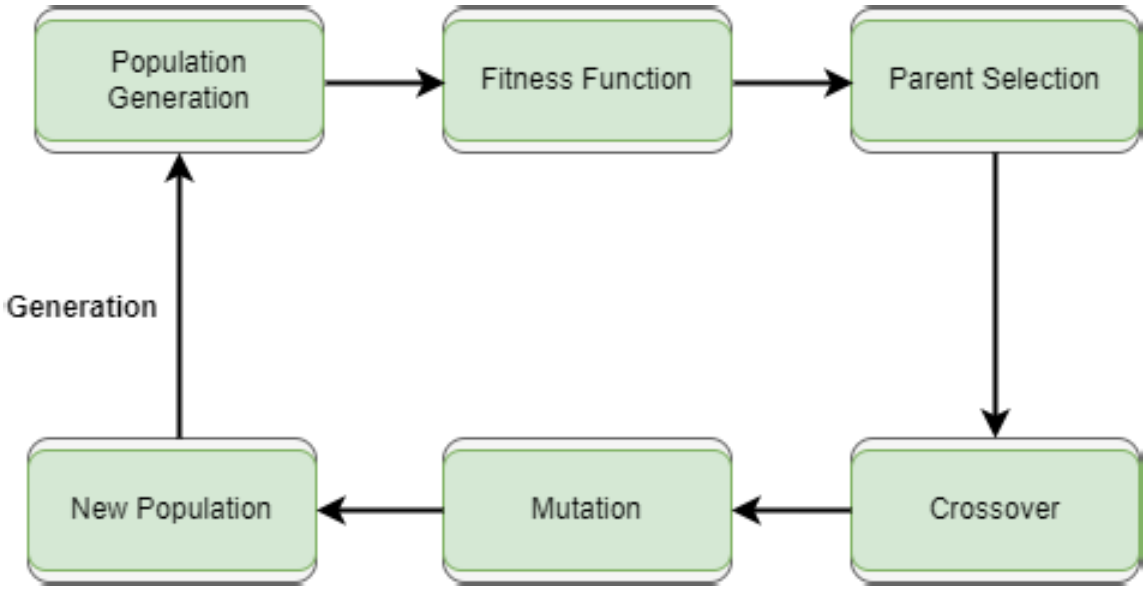

FIGURE 2. Architecture diagram.

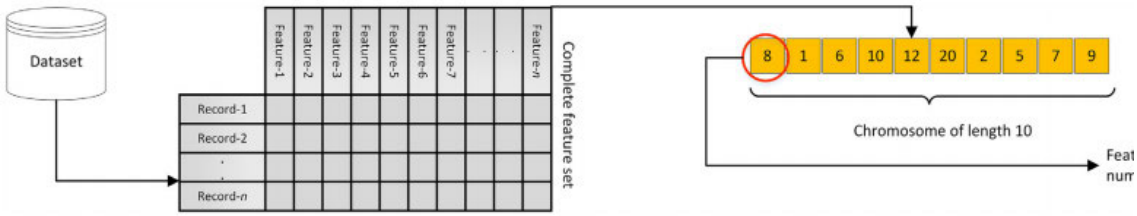

FIGURE 3. Chromosome structure.

Algorithm 1 Create Population

Require: popSize, chromSize

$chromSize \leftarrow Zeros(popSize, chromSize)$
for each $i \in popSize$ **do**
  $genes \leftarrow []$
  while $len(genes) \neq chromSize$ **do**
    $r \leftarrow random.randit(0, featureSize - 1)$
    if $r \notin genes$ **then**
      $genes.append(r)$
    end if
  end while
  $chromArray[i, :] \leftarrow genes$
end for

Ensure: $chromArray$

FIGURE 4. Create population.

Algorithm 1 Parent Selection

Require: chromosomes, noofparents

$chosen \leftarrow []$
for each $k \in noofparents$ **do**
  $select \leftarrow random.randint(0, len(chromosomes[0]) - 1)$
  for each $i \in 3$ **do**
    $new_select \leftarrow random.randint(0, len(chromosomes[0]) - 1)$
    if $chromosome[select, 100] < chromosome[new_select, 100]$ **then**
      $select \leftarrow new_select$
    end if
  end for
  $chosen.append(chromosome[select, 0 : 100].tolist())$
end for

Ensure: $chosen$

FIGURE 5. Parent selection.

chromosomes in such a manner to avoid duplication of gene in chromosome. The Figure 8 represent mutation.

After mutation, the generation is completed. Before starting the next-generation population, the next generation must be created. Twenty percent of elite chromosomes are directly selected in the population for the next generation. Thus, the best chromosome remained in this population. A 20% new random population was generated to add diversity to the population, which will help in exploration. A 30% chromosome was generated with the help of crossover addition in the population. A chromosome of 30% was added to the mutation operator. This new population will be used in the next generation.

H. CLASSIFIERS

To assess our proposed method, we used SVM and Naive Bayes classifiers using the PyCharm tool. The SVM model is

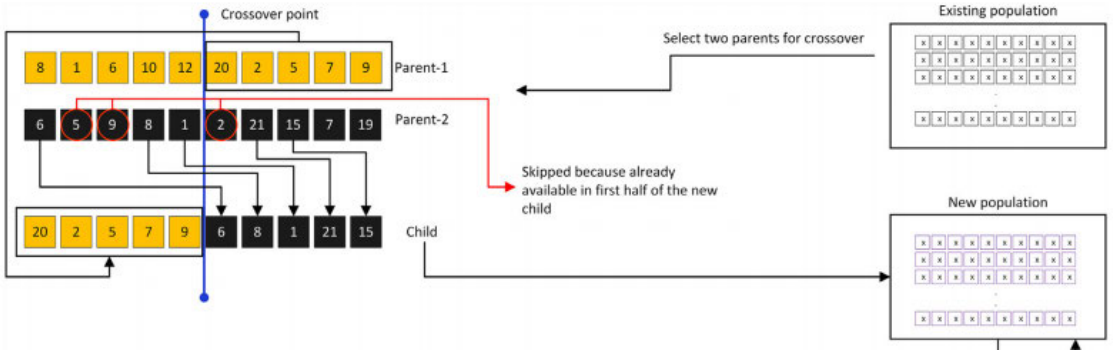

FIGURE 6. Crossover flow.

Algorithm 1 Crossover Algorithm

Require: chromosomes1, chromosomes2
1: $newchrom \leftarrow chromosomes1[int(len(chromosomes1)/2 len(chromosomes1)]$
2: $count \leftarrow 0$
3: **while** $len(newchrom) < chromosomes1$ **do**
4: **if** $chromosome2[count] notin newchrom$ **then**
5: $newchrom.append(chromosome2(count))$
6: **end if**
7: $count \leftarrow count + 1$
8: **end while**
Ensure: $newchrom$

FIGURE 7. Crossover algorithm.

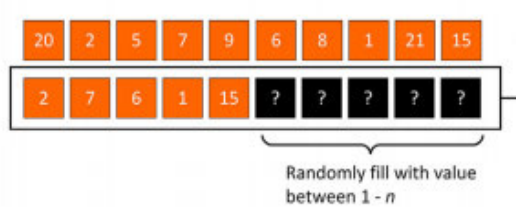

FIGURE 8. Mutation.

based on supervised learning, which originated in statistical learning theory, and is used for regression and classification tasks. Furthermore, SVM is a global classification model that generates non-overlapping partitions and typically uses all attributes. The entity space was partitioned into a single pass, yielding flat and linear partitions. SVMs are based on maximum margin linear discriminants and are similar to probabilistic approaches in that they consider the attribute dependencies. Naive Bayes is a simple technique for building classifiers that assign class labels to instances, which are represented as vectors of feature values, and the class labels are chosen from a finite set. There is no single algorithm for training such classifiers; rather, there is a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of one feature is independent of the value of any other feature given the class variable.

I. TRAINING

We divided our dataset into an 80:20 ratio: 80 for training purposes and 20 for testing purposes. We employed supervised learning algorithms for training.

J. CLASSIFICATION

After the training session in the classification process, we provided the unknown sample as an input to the system, in which the classifier can predict the category of the paper on the basis of its training. The system then performed a comparison between the predicted category and the actual category of the document and reported the result using evaluation measures.

K. EVALUATION

To evaluate our proposed technique, we utilized a well-known evaluation called accuracy. The reason of choosing these evaluation measure is its frequent usage in literature [24], [48], [50].

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

The representation of TP is (True Positive), TN(True Negative), FP(False Positive) and FN(False Negatvie). The accuracy is the ratio of correct prediction against total prediction. The output values of all of the above measures were in the range (0 – 1). Values 0 and 1 present the lowest (0%) and highest (100%) results, respectively.

IV. RESULT

In this section, we exhibit details of the results attained by applying the suggested methodology. We evaluated our datasets for multiclass classification. Moreover, we conducted experiments on individuals and combinations of metadata features. For experimentation, we used the Radrigues and Santos [12] dataset explained in the methodology section. The results of our experiment are as follows:

A. SINGLE METADATA PARAMETERS

First, we evaluated all the metadata features individually and tried to identify the best parameter that contributed more than the other metadata. For the evaluation of our proposed techniques, we collected 500 records from each category (B, C, D, F, G, H, I, J, and K) from the ACM datasets. Subsequently, we divided our dataset into an 80:20 ratio for

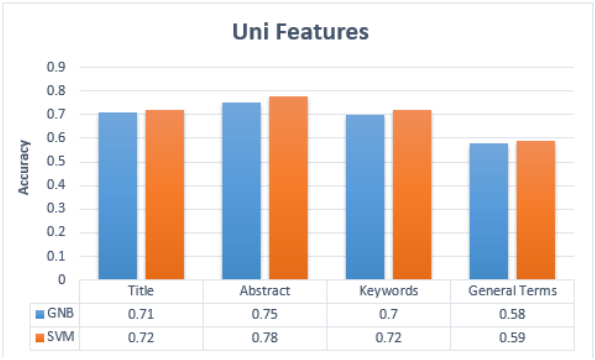

FIGURE 9. Individual metadata.

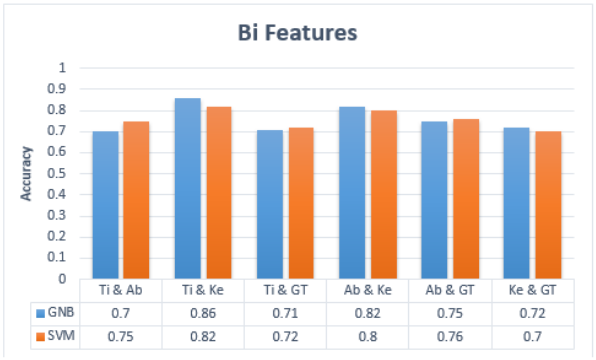

FIGURE 11. Double metadata.

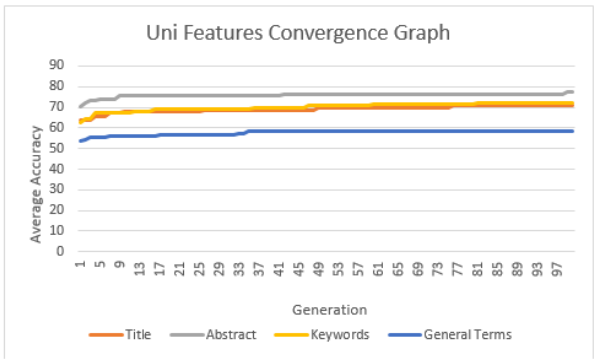

FIGURE 10. Convergence graph against individual meta data.

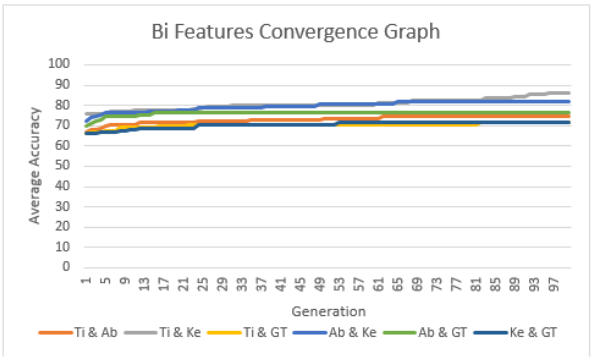

FIGURE 12. Convergence graph of double metadata.

training and testing purposes. Eighty percent of the data were used for training and 20 percent were used for validation. For each individual metadata feature, the accuracy was calculated using two different classifiers (GNB and SVM). The Figure 9 present the average Accuracy of all individual metadata parameter on GNB and SVM classifiers. From the Figure 9 we have analyzed that the abstract parameter achieved highest average accuracy (0.78) compared with the other metadata parameters. Because the abstract represents a summary of a research work, it contains words that specifically denote a particular subject or area. Figure 10 shows a convergence graph of the metadata parameters. From Figure10, we observe that as the number of generations increases, the average accuracy also increases. Therefore, it might be expected that if we further increase the number of generations from 100 to 200, the result will be affected.

B. DOUBLE METADATA PARAMETERS

In the double metadata parameter, every possible combination of the two metadata parameters is exploited to obtain the average accuracy against the classifiers. For the evaluation of our proposed techniques, we collected 500 records from each category (B, C, D, F, H, I, and K) from the ACM datasets. After that we have divided our dataset into 80:20 ratio for training and testing respectively. The 80 percent data used for training purpose and 20 percent is used for validation purpose. For every individual metadata feature accuracy was calculated on two different classifiers (GNB and SVM). Figure 11 presents the average accuracy of all double metadata parameter combinations for the GNB and SVM classifiers. From the Figure 11, we determined that the title and keywords combination achieved the highest average accuracy (0.86) compared to other metadata parameters. Similarly, the abstract and keyword combination achieved the second highest accuracy, which was 0.82. As the abstract represent summary of a research work so it encompasses such like terms which particularly denote the specific subject or area. Moreover, the Figure 10 shows the convergence graph of the metadata parameters. We have analyzed from the above result that adding the keyword metadata with title metadata and abstract can improve the results of multiclass classification of the research articles. The basic reason for the improvement of classification is that, while adding keyword metadata, it provides some more specific words that represent the subject of the paper. These words combine with Title and abstract words and classify the research article more accurately as compare to individual title and abstracts words Figure 12 shows a convergence graph of the double metadata parameters. The convergence graph against the double metadata parameter shows behaviors similar to those of the individual metadata parameters; when the generation increases, the average accuracy also increases.

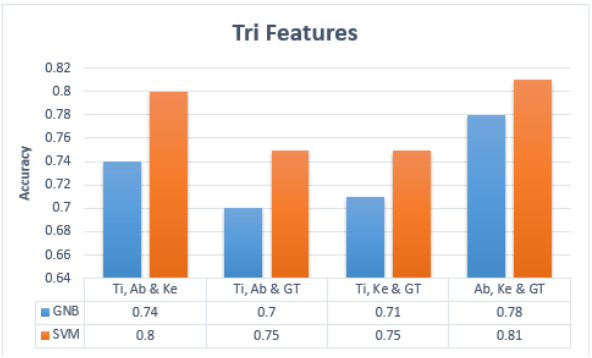

FIGURE 13. Triple metadata features.

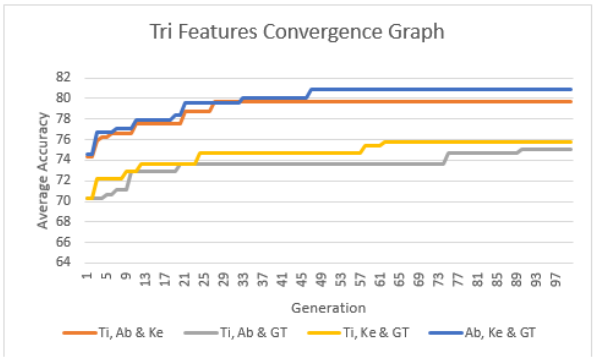

FIGURE 14. Convergence graph.

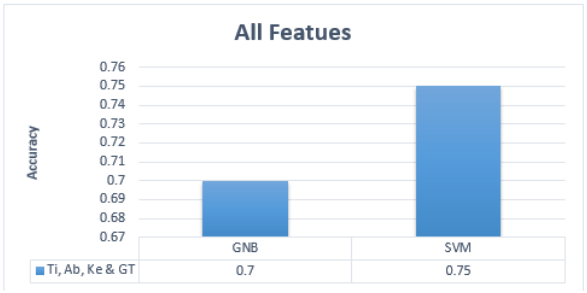

FIGURE 15. All features.

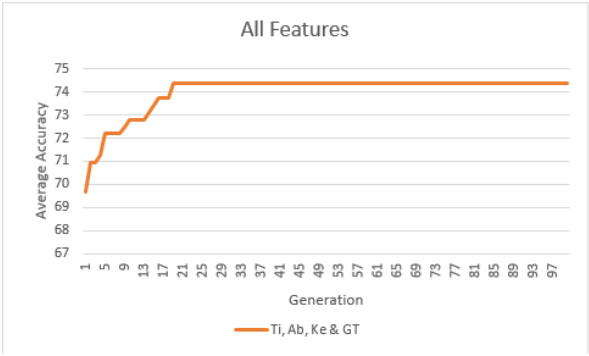

FIGURE 16. All features.

C. TRIPLE METADATA PARAMETERS

In the triple metadata parameter, every possible combination of three metadata parameters is exploited to obtain the average accuracy against the classifiers. For the evaluation of our proposed techniques, we have collected 500 records from each category (B, C, D, F, H, I and K) from ACM datasets. After that we have divided our dataset into 80:20 ratio for training and testing respectively. The 80 percent data used for training purpose and 20 percent is used for validation purpose. For every individual metadata feature accuracy was calculated on two different classifiers (GNB and SVM). Figure 13 presents the average accuracy of all double metadata parameter combinations for the GNB and SVM classifiers.

From the Figure13, we have determined that the abstract, keywords, and general terms combination achieved the highest average accuracy (0.81) compared to other metadata parameters. Similarly, the title, abstract, and keyword combination achieved the second highest accuracy (0.80). Figure 14 shows a convergence graph of the double metadata parameters. The convergence graph against double metadata parameter shows the similar behaviors as in individual metadata parameters that when generation increase the average accuracy also increase.

D. ALL COMBINE METADATA PARAMETERS

In this step, we combine all the features and then exploit them to obtain the average accuracy of the classifiers. For the evaluation of our proposed techniques, we collected 500 records from each category (B, C, D, F,G, H, I, and K) from the ACM datasets. After that we have divided our dataset into 80:20 ratio for training and testing respectively. The 80 percent data used for training purpose and 20 percent is used for validation purpose. For every individual metadata feature accuracy was calculated on two different classifiers (GNB and SVM). Figure 14 presents the average accuracy of all double metadata parameter combinations for the GNB and SVM classifiers. From the Figure 15, we can conclude that, by combining all metadata features, the average accuracy of the system decreases. We observed from the overall results that adding the general terms parameter decreases the results. Moreover, the Figure 12 shows the convergence graph of the double metadata parameters. The convergence graph against all combined features (presented in Figure 16) shows similar behaviors, as shown by the other, with a slight difference that after reaching a specific generation, the system does not increase the results.

V. COMPARISON

The document classification community has proposed multiple approaches to perform multiclass classification. These methodologies have employed the whole content of the research articles, while others have preferred to harness metadata parameters owing to the unavailability of content. In this work, we also utilized the freely available metadata 1) Title, 2) Abstract 2) Keywords, and 3) General Terms for

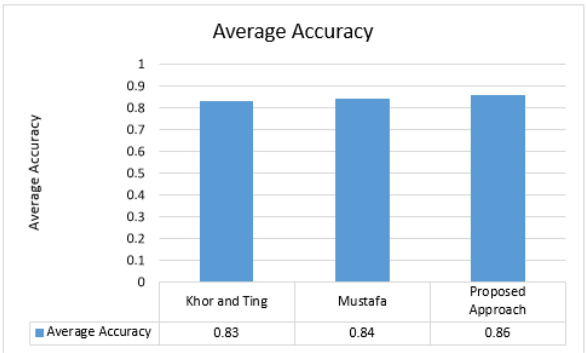

FIGURE 17. Comparison.

performing multi-class classification. The proposed approach is compared with two approaches: Khor and Tang [2], which also utilize the metadata of the research articles. For evaluation Khor and Tang collect 400 educational conference's papers and performed SLC onto four topics such as "Intelligent Tutoring System," "Cognition," "E-Learning" and "Teacher Education." This approach uses different classifiers for classification and achieves an average accuracy of up to 0.83. The second approach is that of Mustafa et al. [11], who also utilized the metadata parameters of the ACM dataset. The comparison results are shown in Figure 17.

The above figure shows that, until now, we have reached 0.86, which is higher than Khor, Ting, and Mustafa et al. However, this experiment was conducted on a sample dataset of 500 papers from each category, generating only 100 generations. In the future, we will extend our dataset and increase the number of generations using a genetic algorithm for the optimization of feature selection. We believe that we will obtain a significant improvement after increasing the number of generations as well as the size of the dataset.

VI. CONCLUSION

In the scientific domain, categorizing research documents into their already defined categories is a key research issue that assists in various scientific aspects. For this task, we extracted four metadata features from the ACM dataset that are freely available and have built all possible combinations of these features. This study is an extension of our previous work; the main focus of our study is on feature optimization and the use of the BERT technique for text representation. For feature optimization, we employed a genetic algorithm. To date, we have tested our approach on a sample dataset, and now we extend our dataset to a large scale. Moreover, in the genetic algorithm, we used only 100 generations, in that we observed that when the generation increases, the average classification accuracy also increases. Now, we will attempt to increase the generation from 100 to 200 or 300.

VII. FUTURE WORK

In this paper, we have conducted document classification on the root level categories of ACM hierarchy. In our next paper, we aim to expand this approach by incorporating second and third level categories from the ACM hierarchy for document classification.

ACKNOWLEDGMENT

The authors would like to express their sincere thanks Al-Ahliyya Amman University and the University of Engineering and Technology Peshawar, Pakistan, for their invaluable assistance and support, which greatly facilitated the execution of this study. Their contributions were instrumental in the successful completion of this research work.

REFERENCES

[1] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Paper recommender systems: A literature survey," *Int. J. Digit. Libraries*, vol. 17, no. 4, pp. 305–338, Nov. 2016.
[2] K.-C. Khor and C.-Y. Ting, "A Bayesian approach to classify conference papers," in *Proc. Mex. Int. Conf. Artif. Intell.* Apizaco, Mexico: Springer, Nov. 2006, pp. 1027–1036.
[3] B. Tang, H. He, P. M. Baggenstoss, and S. Kay, "A Bayesian classification approach using class-specific features for text categorization," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 6, pp. 1602–1606, Jun. 2016.
[4] N. H. N. Le and B. Q. Ho, "A comprehensive filter feature selection for improving document classification," in *Proc. 29th Pacific Asia Conf. Lang., Inf. Comput.*, 2015, pp. 169–177.
[5] K. Chekima, C. K. On, R. Alfred, G. K. Soon, and P. Anthony, "Document categorizer agent based on ACM hierarchy," in *Proc. IEEE Int. Conf. Control Syst., Comput. Eng.*, Nov. 2012, pp. 386–391.
[6] T. Wang and B. C. Desai, "Document classification with ACM subject hierarchy," in *Proc. Can. Conf. Electr. Comput. Eng.*, Apr. 2007, pp. 792–795.
[7] P. K. Flynn, "Document classification in support of automated metadata extraction form heterogeneous collections," Ph.D. dissertation, Dept. Comput. Sci., Old Dominion Univ., Norfolk, VA, USA, 2014.
[8] N. A. Sajid, M. T. Afzal, and M. A. Qadir, "Multi-label classification of computer science documents using fuzzy logic," *J. Nat. Sci. Found. Sri Lanka*, vol. 44, no. 2, p. 155, Jun. 2016.
[9] D. T. Manoj, "A Bayesian approach to classify conference papers," in *Proc. 8th Int. Conf. Sci. Technol. Eng. Math. (ICONSTEM*, Jan. 2023, pp. 1–5.
[10] G. Mustafa, A. Rauf, B. Ahmed, M. T. Afzal, A. Akhunzada, and S. Z. Alharthi, "Comprehensive evaluation of publication and citation metrics for quantifying scholarly influence," *IEEE Access*, vol. 11, pp. 65759–65774, 2023.
[11] G. Mustafa, M. Usman, L. Yu, M. T. Afzal, M. Sulaiman, and A. Shahid, "Multi-label classification of research articles using Word2 Vec and identification of similarity threshold," *Sci. Rep.*, vol. 11, no. 1, pp. 1–20, Nov. 2021.
[12] F. Rodrigues and A. P. Santos, "Multi-label hierarchical text classification using the ACM taxonomy," in *Text Mining Appl. (TeMA) Track EPIA*. Aveiro, Portugal: Department of de Informática, FCT/UNL Quinta da Torre P-2829-516 CAPARICA, 2009.
[13] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Sydney, NSW, Australia: Springer, May 2004, pp. 22–30.
[14] F. Liu, L. Zheng, and J. Zheng, "HieNN-DWE: A hierarchical neural network with dynamic word embeddings for document level sentiment classification," *Neurocomputing*, vol. 403, pp. 21–32, Aug. 2020.
[15] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[16] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.
[17] S. Z. Faiz, "Feature selection for document classification," Ph.D. dissertation, Dept. Inf. Technol., Capital Univ., Jhumri Telaiya, Jharkhand, 2021.
[18] M. Usman, G. Mustafa, and M. T. Afzal, "Ranking of author assessment parameters using logistic regression," *Scientometrics*, vol. 126, no. 1, pp. 335–353, Jan. 2021.
[19] S. A. Devi and S. Siva, "A hybrid document features extraction with clustering based classification framework on large document sets," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 7, pp. 1–11, 2020.

[20] R. A. Stein, P. A. Jaques, and J. F. Valiati, "An analysis of hierarchical text classification using word embeddings," *Inf. Sci.*, vol. 471, pp. 216–232, Jan. 2019.
[21] H. Nanba, N. Kando, and M. Okumura, "Classification of research papers using citation links and citation types: Towards automatic review article generation," *Adv. Classification Res. Online*, vol. 11, no. 1, pp. 117–134, Nov. 2011.
[22] M. Taheriyan, "Subject classification of research papers based on interrelationships analysis," in *Proc. Workshop Knowl. Discovery, Modeling Simulation*, Aug. 2011, pp. 39–44.
[23] N. A. Sajid, T. Ali, M. T. Afzal, M. Ahmad, and M. A. Qadir, "Exploiting reference section to classify paper's topics," in *Proc. Int. Conf. Manage. Emergent Digit. EcoSyst.*, Nov. 2011, pp. 220–225.
[24] N. A. Sajid, M. Ahmad, and M. T. Afzal, "Exploiting papers' reference's section for multi-label computer science research papers' classification," *J. Inf. Knowl. Manage.*, vol. 20, no. 1, Mar. 2021, Art. no. 2150004.
[25] J. Alshehri, M. Pavlovski, E. Dragut, and Z. Obradovic, "Aligning comments to news articles on a budget," *IEEE Access*, vol. 11, pp. 18900–18909, 2023.
[26] E. Chernyak, "An approach to the problem of annotation of research publications," in *Proc. 8th ACM Int. Conf. Web Search Data Mining*, Feb. 2015, pp. 429–434.
[27] M. N. Asim, M. U. G. Khan, M. I. Malik, A. Dengel, and S. Ahmed, "A robust hybrid approach for textual document classification," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 1390–1396.
[28] J. Abreu, L. Fred, D. Macêdo, and C. Zanchettin, "Hierarchical attentional hybrid neural networks for document classification," in *Proc. Int. Conf. Artif. Neural Netw.* Munich, Germany: Springer, Sep. 2019, pp. 396–402.
[29] C. Xu and R.-Z. Xia, "EEG signal classification and feature extraction methods based on deep learning: A review," in *Proc. 2nd Int. Conf. Big Data, Inf. Comput. Netw. (BDICN)*, Jan. 2023, pp. 186–189.
[30] R. Wang and Y. Shi, "Research on application of article recommendation algorithm based on Word2 Vec and tfidf," in *Proc. IEEE Int. Conf. Electr. Eng., Big Data Algorithms (EEBDA)*, Feb. 2022, pp. 454–457.
[31] V. Balys and R. Rudzkis, "Statistical classification of scientific publications," *Informatica*, vol. 21, no. 4, pp. 471–486, Jan. 2010.
[32] S. J. Cunningham and B. Summers, "Applying machine learning to subject classification and subject description for information retrieval," in *Proc. 2nd New Zealand Int. Two-Stream Conf. Artif. Neural Netw. Expert Syst.*, Nov. 1995, pp. 243–246.
[33] A. I. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 273–292, Jun. 2019.
[34] T. Zhou, "Automated identification of computer science research papers," Ph.D. dissertation, School Comput. Sci., Univ. Windsor (Canada), Windsor, ON, Canada, 2016.
[35] X. Luo, "Efficient english text classification using selected machine learning techniques," *Alexandria Eng. J.*, vol. 60, no. 3, pp. 3401–3409, Jun. 2021.
[36] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 29, no. 1, Feb. 2015, pp. 1–7.
[37] J.-O. Kim and J. Curry, "The treatment of missing data in multivariate analysis," *Sociol. Methods Res.*, vol. 6, no. 2, pp. 215–240, Nov. 1977.
[38] C. Zhou, C. Sun, Z. Liu, and F. C. M. Lau, "A C-LSTM neural network for text classification," 2015, *arXiv:1511.08630*.
[39] J. Liu, W.-C. Chang, Y. Wu, and Y. Yang, "Deep learning for extreme multi-label text classification," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 115–124.
[40] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," 2016, *arXiv:1606.01781*.
[41] K. Kowsari, D. E. Brown, M. Heidarysafa, K. J. Meimandi, M. S. Gerber, and L. E. Barnes, "HDLTex: Hierarchical deep learning for text classification," in *Proc. 16th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2017, pp. 364–371.
[42] H. Peng, J. Li, Y. He, Y. Liu, M. Bao, L. Wang, Y. Song, and Q. Yang, "Large-scale hierarchical text classification with recursively regularized deep graph-CNN," in *Proc. World Wide Web Conf. World Wide Web (WWW)*, 2018, pp. 1063–1072.
[43] J. Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," 2016, *arXiv:1603.03827*.
[44] X. Zhu and X. Wu, "Class noise vs. attribute noise: A quantitative study," *Artif. Intell. Rev.*, vol. 22, no. 3, pp. 177–210, Nov. 2004.
[45] N. K. Malhotra, "Analyzing marketing research data with incomplete information on the dependent variable," *J. Marketing Res.*, vol. 24, no. 1, pp. 74–84, Feb. 1987.
[46] R. M. Hamer and P. M. Simpson, "Last observation carried forward versus mixed models in the analysis of psychiatric clinical trials," *Amer. J. Psychiatry*, vol. 166, no. 6, pp. 639–641, Jun. 2009.
[47] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.
[48] A. U. Dey, S. K. Ghosh, E. Valveny, and G. Harit, "Beyond visual semantics: Exploring the role of scene text in image understanding," *Pattern Recognit. Lett.*, vol. 149, pp. 164–171, Sep. 2021.
[49] L. Xiao, G. Wang, and Y. Zuo, "Research on patent text classification based on Word2 Vec and LSTM," in *Proc. 11th Int. Symp. Comput. Intell. Design (ISCID)*, vol. 1, Dec. 2018, pp. 71–74.
[50] Q. Pan, H. Dong, Y. Wang, Z. Cai, and L. Zhang, "Recommendation of crowdsourcing tasks based on Word2vec semantic tags," *Wireless Commun. Mobile Comput.*, vol. 2019, pp. 1–10, Mar. 2019.

GHULAM MUSTAFA received the B.S. degree in software engineering from COMSATS University Islamabad, Abbottabad, and the M.S. degree (Hons.) in computer science from the Capital University of Science and Technology, Islamabad. He is currently pursuing the Ph.D. degree in computer science with the University of Engineering and Technology, Taxila, Pakistan. He has been associated with academia and industry for the last six years. Currently, he is a Senior Lecturer with the Faculty of Computing, Shifa Tameer-e-Millat University, Islamabad. Previously, he was with the CS Department, Capital University of Science and Technology, as an Associate Lecturer and a Junior Lecturer, for the last five years. He has been a Web Frontend Designer and a Backend Developer with A&F Solution Software House. Moreover, he is also an active freelancer with academia for the last 4 years, doing projects in different languages, such as python, java, and C++. In his academic career, he has taught different computer science labs, such as Introduction to Programming Lab (C++), Object Oriented Programming Lab (C++), Advanced Computer Programming Lab (JAVA), Database System Lab, Data Structure Lab (C++), Computer Communication and Network Lab (CNN), and Web development Lab, while serving as a Junior Lecturer. Moreover, he has also taught different computer science courses, such as introduction to programming, object-oriented programming, discrete structure, theory of automata, and software engineering, while serving as a Senior Lecture and an Associate Lecture. During the B.S. degree, he received the two gold medals for his first position in Abbottabad Campus and got first position in all seven campuses of CUI, in 2017, and the Gold Medal due to his excellent academic performance in the entire degree duration, during the M.S. study.

ABID RAUF received the M.Sc. degree in statistics from Quaid-i-Azam University, Islamabad, Pakistan, in 2000, the M.S. degree in information security from Sichuan University, Chengdu, China, in 2007, and the Ph.D. degree from the National University of Sciences and Technology (NUST), Islamabad, in 2021. He is currently a Lecturer with the Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan. He has expertise in the area of information security and privacy. His current research interests include cryptographic protocols and algorithms, security in the Internet of Things (IoT), security in wireless sensor networks and privacy, machine learning, and privacy and covert communications.

AHMAD SAMI AL-SHAMAYLEH received the master's degree in information systems from The University of Jordan, Jordan, in 2014, and the Ph.D. degree in artificial intelligence from the University of Malaya, Malaysia, in 2020. He is currently an Assistant Professor with the Faculty of Information Technology, Al-Ahliyya Amman University, Jordan. His current research interests include artificial intelligence, human–computer interaction, the IoT, Arabic NLP, Arabic sign language recognition, language resource production, the design and evaluation of interactive applications for handicapped people, multimodality, and software engineering.

MUHAMMAD SULAIMAN received the B.S. degree in computer science from COMSATS University Islamabad (CUI), Wah Campus, Rawalpindi, Pakistan, in 2016, and the M.S. degree in computer engineering from Ghulam Ishaq Khan Institute (GIKI), Swabi, Pakistan, in 2019. He is currently pursuing the Ph.D. degree in computer science with the University of Stavanger (UiS), Norway.

WAGDI ALRAWAGFEH received the Ph.D. degree in computer sciences (multi-agent systems) from the Memorial University of Newfoundland Canada. He is currently serving as an Assistant professor with the College of computing and IT, University of Doha for Science and Technology, Qatar. He has more than ten years of experience working with environmental aspects directly related to education, programming, and information technology.

MUHAMMAD TANVIR AFZAL received the Ph.D. degree (Hons.) in computer science from the Graz University of Technology, Austria, and the M.Sc. degree in computer science from Quiad-i-Azam University, Islamabad, Pakistan. He was associated with academia and industry at various levels for the last 20 years. Currently, he is the Director and a Professor with the Shifa School of Computing and Director Campus at one of the largest campus of Shifa Tameer-e-Millat University, Islamabad. Previously, he was the Director and a Professor with the Namal Institute Mianwali. He was a Professor, an Associate Professor, and an Assistant Professor of computer sciences with the Capital University of Science and Technology, Islamabad. Furthermore, he was with NESCOM, COMSATS University Islamabad, and JinTech Islamabad. He has authored more than 125 research articles, including more than 50 published articles in impact factor-leading journals in the field of data science, information retrieval and visualization, semantics, digital libraries, artificial intelligence, and scientometrics. He has authored two books and has edited two books in computer science. His cumulative impact factor is more than 123, with citations of over 1,300. He played a pivotal role in making collaborations between MAJU-JUCS, MAJU-IICM, and TUG-UNIMAS. He has conducted more than 100 curricular, co-curricular, and extra-curricular activities in the last five years, including seminars, workshops, and national competitions (ExcITeCup), and invited international and national speakers from Google, Oracle, IICM, IFIS, and SEGA Europe. Under his supervision, more than 70 postgraduate students (M.S. and Ph.D.) have defended their research theses successfully and several M.S. and Ph.D. students are pursuing their research with him. He served as the master trainer and the program director at a national level training for a public sector organization in Pakistan on human factors engineering and conducted training of over 150 hours for experts from the industry. He remained a master trainer for data science training's in leading industrial organizations. He was a recipient of multiple international research funding. He received the Gold Medal during the M.S. study. He served as the Ph.D. symposium chair, the session chair, the finance chair, a committee member, and an editor for several IEEE, ACM, Springer, and Elsevier international conferences and journals. He is serving as the Editor-in-Chief for a reputed impact factor journals, such as *Journal of Universal Computer Science*.

ADNAN AKHUNZADA (Senior Member, IEEE) is an accomplished cybersecurity specialist and consultant boasting extensive industrial experience and expertise. He has advised some of the largest ICT companies worldwide, effectively securing multi-million-dollar projects. He has made significant contributions to the cybersecurity landscape through impactful published research, successful commercial products, and holding U.S. patents. As a recognized authority in the field, his expertise spans the design of innovative SIEM systems, IDS, IPS, threat intelligence platforms, secure protocols, AI for cybersecurity, secure future internet, adversarial, and privacy-preserving machine learning. With a combination of proven industry success, technical prowess, and a dedication to advancing cybersecurity, he remains at the forefront of the field, continuously shaping its future. With over a decade of experience in the field, he is highly regarded as a Professional Member of ACM.