IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 June 2023, accepted 2 July 2023, date of publication 10 July 2023, date of current version 25 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3293852

RESEARCH ARTICLE

Addressing Imbalance Problem for Multi Label Classification of Scholarly Articles

AIMAN HAFEEZ[1], TARIQ ALI[1], ASIF NAWAZ[1], SAIF UR REHMAN[1], AZHAR IMRAN MUDASIR[2], (Member, IEEE), ABDULAZIZ A. ALSULAMI[3], AND ALI ALQAHTANI[4]

[1]University Institute of Information Technology (UIIT), PMAS Arid Agriculture University, Rawalpindi 46000, Pakistan
[2]Department of Creative Technologies, Faculty of Computing and Artificial Intelligence, Air University, Islamabad 42000, Pakistan
[3]Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[4]Department of Networks and Communications Engineering, College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia

Corresponding author: Tariq Ali (tariq.ali@uaar.edu.pk)

This work was supported by the Deanship of Scientific Research, Najran University, through the Research Groups Funding Program under Grant NU/RG/SERC/12/9.

ABSTRACT Scientific document classification is an important field of machine learning. Currently, scientific document category identification is done manually. There are already defined taxonomies available for categorizing scientific documents, such as the Association for Computing Machinery Computing Classification System (ACM CCS) and Bibsonomy. These taxonomies facilitate authors in the categories of their manuscripts. The incorporation of research work from a variety of domains in the assignment takes on the form of a Multi-Label Classification (MLC). Using MLC, it is possible to assign more than one class to a single document. To address the problem of MLC in its entirety, two distinct methods are used (Problem Transformation and Algorithm Adaptation). The MLC dataset is transformed into one or more single-label datasets through the application of the problem transformation technique. Whereas, a single classifier is modified during the algorithm adaptation process so that it can predict multiple labels. Currently, document classification is done using various techniques in the literature, but none of them paid much attention to the problem of imbalance in Multi-Label Datasets (MLD). However, many effective techniques for dealing with imbalance are available in the literature. The goal of this study is to find an effective technique for balancing datasets before multi-label classification to get better predictions for the classes with fewer instances. Six MLDs, nine transformation techniques and seven classifiers are evaluated in this research work. The proposed research will result in a more accurate recommendation of a research topic for a document. For imbalanced MLDs, LPROS is the best resampling technique using statistical tests. When compared to the other classifiers, the BRkNN classifier is better for MLC. This research will facilitate the classification of documents into their respective classes which can be used by various citation indexes.

INDEX TERMS Multi label classification, imbalanced dataset, resampling, multi label classifier.

I. INTRODUCTION

Classification retrieves interesting patterns from data. The classification works for documents, photos, videos, and audio. Due to data differences, classification has several forms, including Binary [1], Multi Class [2], and MLC [3]. In binary, there are two classes, and an instance may be authentic or false (normal or abnormal). Multiclass offers more than two classes. Single- or Multi-Label. In a single label, the instance belongs to a single class. MLC is two or more class labels, with one or more per instance. Problem Transformation and Algorithmic Adaptation are two MLC techniques. In problem transformation, MLC is converted to binary classification. Binary Relevance (BR), Classifier Chain (CC), and Label Powerset (LP) are common transformation approaches. In algorithmic adaptation, create sophisticated classifiers to handle Multi-Label Datasets (MLD). Multi-Class Multi-Label Perceptron (MMP), Multi-Label k

The associate editor coordinating the review of this manuscript and approving it for publication was Agostino Forestiero.

Nearest Neighbor (MLkNN), and Ranking Support Vector Machine (Rank-SVM) are proposed adaption approaches.

Document classification, also known as document categorization, is a challenge faced by digital libraries [4], information science and the computer science community. The objective of this exercise is to place a document into any number of groups or categories. Either ''manually'' (also sometimes written as ''intellectually'') or algorithmically can do this task. In the field of machine learning, the classification problem can take on several different forms, including classification with multiple labels, as well as the closely related issue of classification with multiple outputs. Both of these problems involve assigning multiple target labels or labels, to each instance [5].

Scientific document classification is a critical issue in the field of computer science and digital libraries [6]. Customarily, for a new scientific document, its substances are analyzed and domain experts categorized those into a few characterized classifications plot an expansive sum of human assets have to go through on carrying out such assignment. The precision of information retrieval from a framework depends on the accurate organization of the document. Automated document labeling is also additionally getting to be more noteworthy with the start of computerized libraries and through tremendous increment within the number of documents on the internet.

Scientific document classification has numerous applications like allocating web categories to a web page or category to a library book. In expansion to information retrieval, rectifying classification makes a difference in finding expertise, analyzing patterns and the relevant document recommender conspire. The analysts create a gigantic number of specialized documents. These can be searchable over the internet by utilizing diverse search engine using Google and digital libraries. The general classification utilized within the research area of computer science is the Association for Computing (ACM) [7]. The manual labeling is getting difficult because of the multi-label assignment since huge number of domains. In addition, research of one domain may extend over to research of other domains [8]. In multi-label scientific document classification, multiple classes may be allocated to a single scientific document. Belonging to multiple categories, the issue of imbalance MLC emerges. This research contributes for imbalanced MLD's classification.

The primary issue with MLC is that it generates MLDs that have an uneven distribution of samples and the labels that correspond to those samples across the data space. This is the root cause of the problem. An unbalanced dataset presents a significant challenge in a variety of real-world applications, including the detection of fraudulent activity, the management of risks, and medical diagnostics. For instance, in the case of a disease diagnostic problem, the major purpose of the task is to identify people who are plagued with diseases. This is because occurrences of the condition are typically low in comparison to the percentage of the population that is healthy. Therefore, a good classification model is one that can correctly categorize unexpected patterns. In the field of single-label classification, the imbalanced class distribution has been the subject of extensive research carried out with the assistance of methods that are in widespread usages, such as resampling techniques. It is unable to directly solve the imbalanced problem in an MLC using the approaches that are currently available due to the imbalance between labels and label sets. For MLDs that have a greater number of labels, finding a solution to the imbalance problem can be more challenging [9].

The imbalance problem in an MLD can be approached from three different angles: the imbalance within labels, the imbalance between labels, and the imbalance among label sets [9]. When there is an imbalance between labels, each label will often have a very high percentage of samples that are considered to be negative and a very low number of samples that are considered to be positive. When there is a label imbalance in the MLD, which occurs when the number of ones (the positive class) in one label may be higher than the number of ones in the other label, the frequency of individual labels is taken into consideration. Every MLD instance is connected to several outputs or labels, and it is typical for some of them to be majority labels while others are minority labels. This means that there are significantly more positive samples associated with some labels than there are associated with other labels. The third type of label imbalance that typically occurs in MLD is the sparse frequency of label sets. This type of label imbalance might be difficult to detect. When the complete label set is taken into consideration, there is a possibility that the most common label sets are associated with the proportion of positive to negative samples for each class. Because there are fewer labels in MLDs, there is a greater probability that each label set will be unique. This shows that some label sets could be considered to be majority cases, while the remainder of label sets could be considered to be minority cases at the same time.

As a result of numerous labels being applied to a single document, the classes have become unbalanced, with the proportion of instances belonging to one class being significantly larger than that of the other class. The prediction for classes that have fewer instances is expected to be poorer. This challenge is made more difficult by the use of several resampling methods and various multi-label classifiers, in addition to the selection of a base classifier. As a result, in order to dig out and create improved predictions for scientific publications, an effective resampling technique in conjunction with an effective multi-label classifier is required

A. RESEARCH QUESTIONS

The following questions will be dealt in this research:

- What are the short comings of the available techniques for multi label scientific document classification?
- Which resampling methods improve the classification results for text data?
- Which multi label classifier is best for document classification?

B. RESEARCH CONTRIBUTIONS

The major contribution is to balanced six MLDs using nine resampling techniques and then classify them using seven multi-label classifiers. First, the dataset needs to be rebalanced through the application of problem transformation approaches which is the resampling of datasets.

Next, the multi-label classifiers will be applied after the transformation of all datasets. Finally, the results are evaluated on the evaluation metric of imbalanced datasets. Statistical analyses and comparisons are performed on the computed measure of various classifiers utilizing various resampling strategies [10].

The remainder of the paper is organized as follows: the background study and literature review regarding how MLC is dealt specifically about the scientific document classification is examined in Section II. Section III illustrates the proposed techniques for balancing of datasets and algorithms for classification. In Section IV, the experimental work is presented. It also includes the evaluation of resampling techniques and multi-label classifiers by using different statistical tests. Following the conclusion and future work in Section V.

II. LITERATURE REVIEW

Yan et al. have performed multi-label document classification with label ranking [11]. The author proposed a Long Short Term Memory (LSTM) approach based on the ranking of multi label. The proposed technique demonstrate for the classification of document compromising of repLSTM. It is a versatile approach for the process of representation of data, and rank LSTM, a unified process for learning-ranking. In the approach repLSTM, by incorporating document labels the supervised LSTM is utilized to learn a representation of the document. In the approach rankLSTM, according to the semantic tree, the rearranged order of documents. These semantics are consistent and relevant to the successive learning of the proposed approach. The complete training of the model is based on the labels that are predicted. The representation of the document has two parts: global features (globalrep) are present in the first part, whereas local features (localrep) are present in the second part. For experimental work, three datasets were used which are Medical Literature (MEDLINE), Enron corpus and RCV1 with two feature representations BOW and WE. 10 baseline models, seven cutting-edge: Three are component methods: SVM, Naïve Bayesian (NB), Predicting Clustering Trees (PCT), Hierarchy of Multi-Label Classifier (HOMER), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), MLkNN. LSTM-flat, RNN-Rank, and RNN-LSTM are utilized. For performance evaluation, three metrics Recall, Precision and F-measure are used. The result shows that the proposed model gets high performance for the classification of document assignments.

Assigning multiple labels to scientific publications from a given taxonomy is digitally performed [4]. Also evaluated different similarity measures for text data. For this purpose, the Particle Swarm Optimization (PSO) algorithm is proposed that could assign a label from taxonomy. In the proposed solution, the convergence of MLD into single-label dataset is done first. For convergence, PSO algorithm used that provides high convergence rate along with global and local optimum. For experimental work, ACM and Journal of Universal Computing (J.UCS) datasets are used having 86116 and 1460 papers, 11 and 13 classes and 137679 and 3044 labels respectively. Conversion of multi-label to single-label datasets is performed using max, min, ran and single selection techniques. After conversion, the classification is performed with four similarity measures with the proposed algorithm. Jaccard, Levenshtein, Jaro Winkler and Euclidean similarity measures are used. For comparison, NB, ZeroR, Sequential Minimal Optimization (SMO), Kstar and J48 algorithms were used for the transformed dataset in Weka. The result shows that the proposed approach gives 15% better accuracy than state-of-art algorithms. In similarity measures, the Jaccard text similarity measure gives better results as compared to the other three. In the future, the algorithms can be empirically validated for time complexity as well.

Huang et al. classified the entire hierarchical category structure accurately. Hierarchical Attention-based Recurrent Neural Network (HARNN) is proposed [12]. From this architecture, the document can be automatically annotated level by level. To get the hierarchical structure and representation of texts, first applied Documentation Representing Layer (DRL) on all categories and documents. Hierarchical Attention-based Recurrent Layer (HARL) simulates hierarchical interactions from the top down. The attention mechanism considers both each text's contribution to each category and the next level category that will be influenced. HAM determines dependencies between hierarchical levels. After that, a Hybrid Predicting Layer (HPL) was used to combine level and category predictions. Patent documentation and educational activities are employed for experimental work. These datasets comprise categorical and text documents. Clus-HMC, HMC-LMLP, HMCN-F, and HMCN-R are four state-of-the-art models. Precision, Recall, micro-F1, and AU(PRC) are evaluated. HMC-LMLP is local, HMCN-F, HMCN-R hybrid, and Clus-HMC global. HARNN performs well on both datasets on all criteria. HARNN is more effective and accurate at solving hierarchical HMTC tasks. Future labels won't be incomplete. Also, employ more effective ways to depict category hierarchies. Bi-HAM model will capture hierarchical data in both directions. The proposed method will be used to predict protein function in biochemistry.

Zhang et al. have investigated how effectively and accurately recommend the reviewer to review the submitted paper [13]. For this purpose, the method of Multi-Label Classification using Hierarchical and Transparent Representation named (Hiepar-MLC) is proposed. To begin, Hiepar-MLC is responsible for the expression of semantic information regarding both the submitted work and the reviewer. The approach that was suggested is based on a two-level

bidirectional gated recurrent unit, which also has an attention mechanism applied to it. It was able to collect the hierarchical information of two levels by basing it on word-sentence-document relationships in order to support labels. The problem with the recommendations of paper reviewers was then changed into a classification of multiple-label problems. This guides the learning process through multiple research labels. The author also proposed the framework for the selection of the most specific reviewers from the predicted results of several labels named as Multi-Label Based Reviewer Assignment (MLBRA). The dataset from ACM Digital Library builds for paper reviewers. The profiles of reviewers created for their publications. For experimental work, the dataset of 931707 computer science papers utilized in which 13449 papers of year 2017 are used to review. From these papers, the information of authors, title and abstract, date of publishing and research labels of each paper is used. The 22575 reviewers were used as a training dataset while the paper published in year of 2017 was used as a testing dataset. For 1944-label space, PfastreXML, the method of classification of multi-label selected for the training of multi-label model. For performance comparison, six baseline models LDA, CNN, LSTM, Bi-LSTM, Word2vec and Bidirectional Encoder Representation from Transformers (BERT) are used. For performance evaluation, Recall and Normalized Discounted Cumulative Gain (NDCG) metrics are used. Result shows that Hiepar-MLC outperforms the existing paper reviewer recommendation approaches and learning methods for representation.

The quandary of a multi-label hierarchical classification in the context of research papers addressed assigning a set of relevant labels to papers from the hierarchy [6]. This framework includes a co-training algorithm that exploits the content and bibliographic information. In this framework, for the learning of different views of labeled data, two hierarchical multi-label classifiers are utilized that select the most confident unlabeled sample iteratively and add it to the labeled set. Maximum Agreement and Labels Cardinality As a selection criterion, consistency was used for the selection of confident unlabeled samples. Then, applying the oversampling method for rebalancing the label distribution of the initial label set reduced the issue of label imbalance. The oversampling technique was Multi-Label Synthetic Minority Oversampling Technique (MLSMOTE). Random Forest (RF) is utilized as a fundamental binary classifier. For experimental work, they utilized the 3,170 scientific papers that were recaptured from the ACM digital library. For the automated extraction of the scientific paper, the Connotate tool was utilized. For performance evaluation, Micro-F1 and Macro-F1 metrics are used. The results show that the proposed approach outperforms the maximum agreement and supervised approaches in some ways. In the future, some more imbalance techniques can be used to improve performance.

Salunkhe & Bhowmick have studied the classification of multi label data. For the classification of multi label data various machine learning techniques discussed in this paper [14]. The comparison of several machine-learning techniques was done, which involved two approaches: algorithm adaptation and problem transformation. The Naïve Multinomial Bayes (NMB) and the logistic regression models were utilized for this purpose. In NB, the binary mask is taken over several labels. A classification technique for learning many labels that uses the ''One vs Rest'' framework. Logistic Regression was performed on the data for each cluster, taking into account all of the labels. As a dataset for the experimental work, 159571 different cases with comments were employed. Jigsaw is responsible for the production of the dataset, and users can access it through Kaggle. This contains hateful remarks that are obscene, insulting, threatening, and contains info that promotes identity hatred. For performance assessment, the metrics accuracy, precision, recall and F1 score were utilized. The result shows that logistic regression performs best as it achieved 0.98 accuracy, 0.97 precision and 0.99 F1 score. In future, higher performance can be attained using a more robust classification model From the literature review, it has been concluded that most researchers had not dealt with the problem of imbalance dataset for scientific document classification. In addition, few of them considered it and proposed techniques for rebalancing of the dataset for classification. Nevertheless, biasness and information loss restrict these methods. It has been considering a smaller number of categories from the dataset, regarding categorization of scientific document. However, classification with balanced dataset is important for providing more accurate automated category to scientific document. However, classification with balanced dataset is important for providing more accurate automated category to scientific documents. Therefore, the need is to propose most effective technique for balancing the imbalance dataset for classifying the multi label scientific document. Another issue that requires additional research is the best way to choose the base classifiers when using MLC. This is as a result of the fact that various combinations of base classifiers can potentially perform in a variety of different ways depending on the problem domain that is being considered.

III. MATERIAL AND METHODS

The currently used text classification schemes for scientific publications are not equipped to deal with multi-label assignment of labels. The majority of the currently available methods are evaluated either using small data sets or converted to single-label classification. It is necessary to apply multiple labels, based on a particular taxonomy, to a single document. The proposed solution uses multiple datasets that come from a variety of domains having different taxonomies. The step of the methodology that has been proposed in Figure 1, involves rebalancing the MLD dataset using a variety of resampling methods. In the second phase is the classification process, multiple multi-label classifiers are applied to predict class labels. The obtained results are subjected to one last round of statistical analysis, which is carried out using performance evaluation metrics. Statistical

methods are currently being utilized in order to explore the best resampling technique and multi-label classifier.

A. DATA DESCRIPTION

Different text datasets from multiple domains have been selected for this research. It includes the ACM, Bibtex, Eurlex, J.UCS, Reuters and TMC datasets. A description of each dataset is given below.

- **ACM:** The ACM dataset is about the computer science research articles from the ACM taxonomy. The original dataset contains 86116 instances with 137679 attributes and 11 labels. In our experiments, we used 31400 instances with 16250 attributes and 3 labels [15].
- **Bibtex:** The bibtex dataset is concerned with the social bookmarking and publishing sharing system, and it is annotated with a selection of tags taken from the Bibsonomy database. It contains 7395 instances with 1836 attributes and 159 labels.
- **Eurlex:** The Eurlex dataset is about the documents of European Union Law. The documents may be legislation, treaties, case law and legislative proposals. It contains 19350 instances with 5000 attributes and 201 labels.
- **J.UCS:** The J.UCS dataset is about research articles. It contains 1112 instances with 3928 attributes and 13 labels [16].
- **Reuters:** The Reuters dataset is about articles. It contains 6000 instances with 500 attributes and 103 labels.
- **Tmc:** The Tmc dataset is about aviation reporting systems having aviation safety reports. It contains 28600 instances with 500 attributes and 22 labels.

B. RESAMPLING

The strategies for addressing the imbalance in the MLC can be broken down into four distinct categories. These include classifier adaptation, ensemble approaches, resampling approaches, cost-sensitive approaches and classifier adaptation.

In the process of classifier adaptation, the algorithms could be categorised as dedicated if they directly learn the imbalance distribution from the classes contained within the datasets. Cross-Coupling Aggregation (COCOA) [17], Imbalanced Multi-Instance Multi-Label Radial Basic Function (IMIMLRBF) [18], Imbalanced Multi-Modal Multi-label Learning (IMMML) [19], Min-Max Modular Network with Support Vector Machine (M3-SVM) [20], Two Stage Multi-Label Hyper Network (TSMLHN) [21], Un Balanced Multi-Label Relief Feature (UBML-ReliefF) [22], and Weakly Supervised Multilabel Learning for Class Imbalance (WSML-CIB) [23] are some of the adaptation methods that have been proposed in the published research.

In cost-sensitive approaches, various cost metrics are utilized to describe the costs of any one particular misclassified sample, with the end goal of reducing the overall cost as much as possible. In most cases, these strategies are utilized in the correction of imbalanced learning by associating a high misclassifying cost with the underrepresented classes. At the data level as well as the algorithmic level, cost-sensitive approaches can be implemented by taking into consideration the higher costs associated with the incorrect classification of minority samples in comparison to majority samples. In the research that has been done, cost-sensitive methods such as Costsensitive Positive Negative Label (CPNL) [24], Cost Sensitive Rank Support Vector Machine (CSRankSVM) [25], and Sparse Oblique Structured Hellinger Forests (SOSHF) have been proposed [26]. The problem of class inequality is addressed by employing a loss that is sensitive to costs.

In ensemble approaches, several different base models are combined into one in order to produce the most accurate predictive model possible. When it comes to single-label classification, the use of ensembles comprised of multiple classifiers is effective. Several different multi-label classifiers are trained by the ensemble of multi-label classifiers. Therefore, each of the trained classifiers is unique and is capable of making a variety of predictions across multiple labels. The following are some of the cost ensemble approaches that have been proposed in the research literature: Binary Relevance Inverse Random under-sampling (BR-IURS, Ensemble Classifier Chain Random Under sampling (ECCRU3), Ensemble Multi Label (EML), Human Protein Subcellular Location Prediction (HPSLPred), and Stacked Multi-Label k Nearest Neighbor (SMLkNN) [27].

The pre-processing of the MLDs is the foundation upon which resampling strategies are constructed. This method is one of the most frequently employed techniques for dealing with imbalanced data. They are a part of the classifier-independent group and have the objective of developing new versions of MLDs that are more well-balanced. Undersampling eliminates samples associated with the majority label, while oversampling creates new samples associated with the minority label. Resampling methods can be based on either undersampling or oversampling, or they can involve both of these actions simultaneously. These methods can also be categorised further into two distinct sub-groups: random methods and heuristic methods. The distinction between the two is based on the procedure that is used to add or remove samples. Since the existing resampling methods are not directly applicable to MLC, the random resampling approach that is used for MLC employs different methods than the ones that are used in single-label classification. Methods of randomized resampling that apply to MLC can be based on the BR methods, LP transformation, imbalance measures and other such methods. The following are some examples of multi-label random resampling methods that can be found in published works: Label Powerset Random Oversampling (LPROS), Label Powerset Random Under sampling (LPRUS), Multi-Label Random Oversampling (MLROS), Multi-Label Random Under sampling (MLRUS), Resampling Multi-Label Datasets by Decoupling Highly Imbalanced (REMEDIAL), and Resampling Multi-Label Datasets by Decoupling Highly

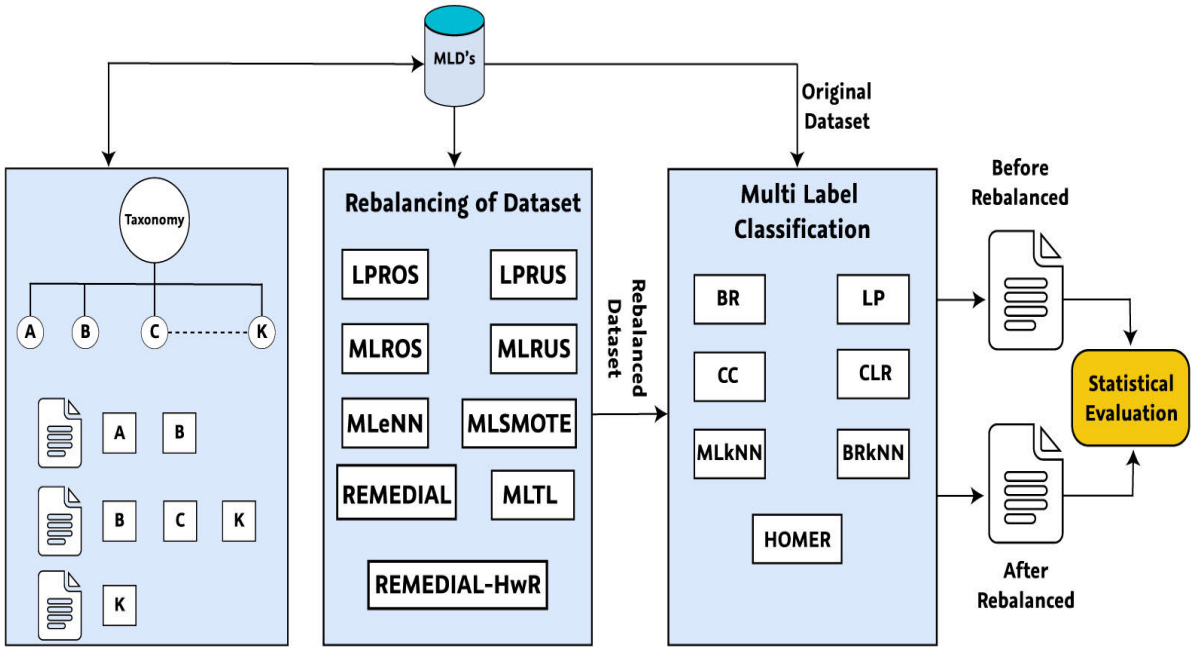


FIGURE 1. Proposed Approach for Multi-label Classification.

Imbalanced-HwR (REMEDIAL-HwR) [28], [29], [30]. The instances that were chosen heuristically, as opposed to being chosen at random, were either deleted or cloned during the multi label heuristic resampling process. The following are some examples of multi label heuristic resampling approaches that can be found in published works: Multi Label edited Nearest Neighbor (MLeNN), MLSMOTE and Multi Label Tomek Link (MLTL) [31].

C. MULTI LABEL CLASSIFIERS

After the balancing of datasets, we have classified datasets by using seven multi-label classifiers. The multi-label classifiers are Binary Relevance (BR), Label Powerset (LP), Classifier Chain (CC), Calibrated Label Ranking (CLR), Hierarchy of Multi-Label Classifier (HOMER), Multi-Label k Nearest Label (MLkNN) and Binary Relevance k Nearest Neighbor (BRkNN).

1) BINARY RELEVANCE (BR)

It is arguably the strategy that provides the least amount of complexity when it comes to learning from data that contains multiple labels [13]. For this strategy to be successful in resolving the MLL issue, it must first be segmented into several independent binary learning tasks (one per class label). When it comes to gaining knowledge from training instances that contain multiple labels, BR is likely the most straightforward approach. Using this approach, the problem of learning multiple labels is partitioned into separate instances of the binary learning problem.

2) LABEL POWERSET (LP)

The LP method is a technique that does not require an excessive amount of complexity and has the advantage of taking into consideration the correlations between labels. When completing a classification task with a single label, LP treats each subset of L that is present in the training set as a distinct class value. This subset of L is known as a label set, and it is referred to in the previous sentence as a label set. LP is an intriguing research method because it has the benefit of considering label correlations, which makes it more accurate. This benefit makes LP an interesting form of study. Sometimes it is feasible to obtain higher levels of performance by doing things in this manner. On the other hand, LP runs into problems in application domains that have a significant number of labels and training instances. This is because the training set often comprises a significant number of label sets. This work proposes randomly dividing the initial set of labels into several small-sized label sets, and then using LP to train a corresponding multi-label classifier based on the label sets that were generated. This is done to solve the issues that have been brought up in the past concerning LP. Because of this, the resulting single-label classification jobs are easier to compute, and the distribution of their class values is less uneven than it was in the past [3].

3) CLASSIFIER CHAIN (CC)

Binary classifiers are utilized in the CC, much like they are in the Binary Method (BM) [32]. Classifiers are linked together in the form of a chain, with each classifier solving the BR problem associated with the label $l_j \in L$. Each classifier is connected after the previous one in the form of a connection. The expansion of the feature space occurs as the 0/1 label associations of all of the preceding links in the chain are added to the feature space of each subsequent link in the chain. Remember the notation for a training example, which

is *(x, S)*, where S is a binary feature vector expressed as $(l_1, l_2, \ldots l_{|L|,}) \in \{0, 1\}^{|L|}$ and x is an instance feature vector. As a direct consequence of this, a string of binary classifiers, represented by the notation C_1 and $C_{|L|}$, is produced. Given the feature space, it is the responsibility of each classifier in the chain, denoted by the letter C_j, to learn about the binary connection of the label l_j and to make predictions about it. These predictions are augmented by all of the preceding binary relevance predictions that were made by classifiers l_1, and lj 1 in the chain. The process of classification starts at C_1 and goes down the chain: C_1 determines $\Pr(l_i|x)$, and every classifier that follows after it, from C_2 to $C_{|L|}$, predicts $\Pr(l_j|x_i, l_1,\ldots,l_{j-1})$ This chaining method passes label information from one classifier to the next. As a direct consequence of this, CC is in a position to take into consideration label correlations, which enables it to side step the difficulty presented by BM's reliance on labels [4].

4) CALIBRATED LABEL RANKING (CLR)

In CLR, the goal is to learn a mapping from instances x from an instance space X to rankings x (total strict orders) over a finite set of labels $L = \{\lambda 1, \ldots., \lambda c\}$, where $\lambda i x \lambda j$ denotes that, for instance, x, label λi is preferred overλj. The problem that needs to be solved in order to accomplish this is known as the mapping learning problem. Learning mappings can be applied in a variety of contexts, including CLR. The skill of learning total stringent ordering can be applied to the process of calibrated label rating. A ranking can be represented by a permutation as long as there is a unique permutation τ such that $\lambda i\ x\ \lambda j$ iff $\tau(\lambda i) < \tau(\lambda j)$, where $\tau\lambda i$ signifies the position of the label λi in the ranking. In other words, if there is a unique permutation, then the ranking can be represented. A permutation can be used to indicate a rating as long as there is at least one permutation that is not identical to any other permutation. The label λ that will be placed in position i will be represented by the notation$\tau - l$, which will be written in parentheses (i). From this point forward, we shall refer to the target space of all permutations over c labels as Sc. When doing MLC, each training example x is assigned a subset of the total available labels. This assignment is denoted by the notation $P_x \subseteq L$. As a consequence of this, the group of preferences referred to as Rx is implicitly defined in the following manner: $Rx = \{\lambda_i x \lambda_j | \lambda_i \epsilon Px, \lambda_j \epsilon LPx$. It is at this point that the projection of the output space onto the first l components takes place.

5) HIERARCHY OF MULTI LABEL CLASSIFIER (HOMER)

The methodology of divide-and-conquer is utilized in the design of the HOMER algorithm [3]. An MLC task with a large number of labels, denoted by the value L, is transformed into a tree-shaped hierarchy of more straightforward MLC tasks, each of which deals with a smaller number, denoted by the value k, of labels. This is the central idea. For the MLC of a new instance x, HOMER begins with hroot and then employs a recursive process that sends x to the multi-label classifier hc of a child node c only if c is included among the predictions made by hparent (c). At some point in the future, this process may result in the multi-label classifier(s) just above the corresponding leaf predicting one or more single labels. In this particular instance, the output of the suggested methodology is the union of these predicted single labels, whereas in other circumstances, the empty set is returned.

6) MULTI LABEL K NEAREST NEIGHBOR (MLKNN)

For this classifier, the MLkNN method uses kNNs. Given an instance x and the label set $Y \subseteq Y$ associated with it. Let's say that yx is the category vector for x, and that its lth component, $yx(l)$ $(l \epsilon Y)$, has a value of *1* if l is greater than Y, but a value of 0 otherwise. In addition, the set of KNNs of x that were found in the training set will be denoted by the letter *"N" (x)*. MLkNN begins by locating its kNNs *N (t)* in the training set for every test instance t that it encounters. Let's say that the event *Hl 1* is the one in which t possesses the label l, and *Hl 0* is the one in which t does not possess the label l. In addition, we will refer to the occurrence of the event denoted by $Elj(j \epsilon \{0, 1, \ldots., K\})$ as the fact that, among the kNNs of t, there are precisely j instances that bear the label l.

7) BINARY RELEVANCE K NEAREST NEIGHBOR (BRKNN)

BRkNN is a variant of the kNN method that is theoretically identical to utilizing BR in conjunction with the kNN algorithm. BRkNN was developed by Microsoft Research and is a registered trademark of Microsoft [33]. The kNN algorithm was modified to create BRkNN, which can be considered a better version of kNN. The BRkNN algorithm is an extension of the kNN algorithm that was designed so that separate predictions may be generated for each label after a single search of the k nearest neighbors. The goal of the BRkNN algorithm is to improve the accuracy of the predictions provided by the kNN algorithm. This was done to prevent unnecessary computations that would have taken up a lot of time. Because of this, BRkNN is L times faster than BR plus kNN when it is being tested, which is a fact that may be particularly crucial in fields that require short response times and have a big number of labels.

D. STATISTICAL EVALUATION

Data collection and interpretation are the two main components of statistical analysis, which aims to identify patterns and trends in the data. Finding trends is the purpose of statistical analysis. Here we see which resampling technique and multi-label classifier is best for document classification. For this purpose, we use the Wilcoxon Test for resampling techniques and the Friedman Test for the ranking of classifiers.

1) WILCOXON STATISTICAL TEST

The rank sum test and the signed-rank test are the two variations of the Wilcoxon test, which compares two groups that have been paired together. The purpose of the test is to establish whether or not two or more sets of pairs can be distinguished from one another in a manner that can be

TABLE 1. Description of Datasets.

Dataset	Domain	Instances	Attributes	Labels
ACM	CS Articles	31400	16250	3
Bibtex	Social Publications	7395	1836	159
Eurlex	European Law Documents	19350	5000	201
J.UCS	Scholarly Articles	1112	3938	13
Reuters	Benchmark Articles	6000	500	103
Tmc	Aviation Safety Reports	28600	500	22

substantiated using statistical evidence. Both iterations of the model begin with the presumption that the pairs that make up the data come from dependent populations. This means that they track the same person or share price at different points in time and locations. In this section, we make use of it to determine which method of resampling is the most effective [31].

2) FRIEDMAN STATISTICAL TEST

The Friedman Test is an alternative to the repeated measures Analysis of Variance (ANOVA) that does not rely on metrics. It is used to assess whether or not there is a statistically significant difference between the means of three or more groups in which the same subjects appear in each group. Each group must have the same participants for this test to be valid. In particular, it is applied to the task of determining whether or not there is a difference between the averages of three or more different groups. The Friedman Test is commonly utilized in either of these two situations: first, when determining the mean scores of subjects over three or more time periods; and second, when determining the mean scores of subjects across three or more conditions [31].

IV. RESULTS AND DISCUSSIONS

In this section, the experimental setup of the proposed approach which includes the description and characteristics of datasets, experimental environment, selection of base classifier, parameters for classifiers and evaluation metrics are described along with results and their discussions.

A. DATASETS

We have used the six benchmarks MLDs based on documents for classification presented in Table 1.

1) CHARACTERISTICS OF DATASETS

Before constructing a classification model to address a particular issue, it is important to analyze the datasets to understand the variables' relationships and identify a viable model. As part of the MLD preparations, the relationship between the labels, their frequency, and the imbalance ratio is tabulated in Table 2. The multi-label imbalance dataset includes label distribution, label relationships, and imbalance measurements. When it comes to fundamental measurements, we have to take the number of instances, the number of characteristics, as well as the number of labels and label sets into consideration.

Label Distribution: In label distribution, we have to look at the cardinality and density of the labels. Cardinality: The number of average labels connected to each instance as given in Equation 1.

$$\text{Card(D)} = \frac{1}{n}\sum_{i=1}^{n}|X_i| \tag{1}$$

In contrast, D is a MLD, n is the total number of labels, and X is the set of labels, with X_i standing for the ith label in the label set.

Density: The cardinality of the collection of labels divided by the total number of possible labels as given in Equation 2.

$$\text{Den(D)} = \frac{1}{|X|}xCard(D) \tag{2}$$

Imbalance Level Measures: In imbalance measures, we have to look at how much the dataset is imbalance. To accomplish this, we can compute the imbalance ratio for each label, as well as the mean imbalance ratio, the maximum imbalance ratio, and the coefficient of variation of the imbalance ratio.

Imbalance Ratio per Label: It is determined by dividing the specific label from the label set by the label that constitutes the majority in the label set as given in Equations 3 and 4. The most frequent label will have an IRLbl of 1, while the remaining labels will have an IRLbl that is greater than 1. The specific label has a higher imbalance ratio the higher the value of IRLbl.

$$\text{IRLbl}\,(\sigma) = \frac{max_{\acute{\sigma}\in X}(\sum_{i=1}^{n}h(\acute{\sigma}, X_i))}{\sum_{i=1}^{n}h(\sigma, X_i)} \tag{3}$$

$$h\,(\sigma, X_i) = \left\{\begin{array}{l} 1\ \sigma \in X_i \\ 0\ \sigma \notin X_i \end{array}\right\} \tag{4}$$

In this case, D is a MLD, n is the total number of labels, X is the set of labels, with Xi being the ith label of the label set, and is the label for which IRLbl will be calculated.

Mean Imbalance Ratio: It is measured as the average imbalance ratio among all labels in the MLD that significantly benefit from the resampling techniques to balance the datasets as given in Equation 5.

$$\text{MeanIR} = \frac{1}{p}\sum_{\sigma\in X} IRLbl(\sigma) \tag{5}$$

Maximum Imbalance Ratio: It is measured as the ratio of the most frequent label against the rarest label as given in Equation 4.6.

$$\text{MaxIR} = max_{\sigma\in X}(IRLbl(\sigma)) \tag{6}$$

Coefficient of variation of IRLbl: It is measured as the similarity of the level of imbalance between all labels as given in Equation 7 and 8. This value should be greater than 0.5 to

TABLE 2. Characteristics of Datasets.

Datasets	Cardinality	Density	MeanIR	MaxIR	CVIR
ACM	1.000	0.333	1.191	1.544	0.257
Bibtex	2.402	0.015	12.498	20.431	0.405
Eurlex	2.213	0.011	536.976	4290.0	2.135
J.UCS	1.660	0.128	8.072	37.900	1.345
Reuters	1.462	0.014	54.081	477.000	1.922
Tmc	2.22	0.101	17.314	41.980	0.814

get significant benefits from the resampling techniques to balance the datasets.

$$\text{CVIR} = \frac{IRLbl\varphi}{MeanIR} \tag{7}$$

$$IRLbl\varphi = \sqrt{\sum_{\sigma \in X} \frac{(IRLbl\,(\sigma) - MeanIR)^2}{p-1}} \tag{8}$$

In general rule, any MLD that has a MeanIR value that is larger than 1.5 (on average, 50% more samples with majority label compared to minority label) and a CVIR value that is greater than 0.2 (with 20% variance in the IRLbl values) should be considered to be imbalanced.

The most important aspects of these datasets are outlined in Table 2. The datasets, which were selected very carefully to accurately represent a range of different levels of mean imbalance rate, may contain a variety of imbalance levels for us to observe. The ACM datasets have lower MeanIR and CVIR that is 1.191 and 0.257 respectively. Therefore, it isn't considered an imbalanced dataset. Concerning the given datasets, Bibtex, J.UCS, and Tmc are the datasets with the least amount of imbalance, as indicated by their respective MeanIR values (12.498, 8.072, and 17.314) and CVIR values (0.405, 1.345 and 0.814). With a MeanIR of 536.976 and CVIR of 2.135, the Eurlex dataset is by far the one with the most significant imbalance. In addition, we can make the observation that the Reuters dataset also exhibits a significant degree of imbalance.

B. EXPERIMENTAL ENVIRONMENT

The MULAN framework is the most frequent repository used by a large number of authors in their publications that are concerned with MLC. The MULAN framework, which is an open-source package for MLC and was put to use in evaluating the multi-label classifiers.

1) SELECTION OF BASE CLASSIFIER

One of the most effective regression and classification algorithms available is called Random Forest (RF), and it is used extensively. The fact that the algorithm is so straightforward makes it a compelling option for text classification. Significant advantages over other machine learning models include its capacity to manage high dimensional data and its high performance even when dealing with imbalanced datasets. The 'wisdom of the crowd' is used as the foundation for RF's decision-making process, which incorporates a large number of decision trees. When making the ultimate choice, it takes into account the average or means of the results obtained from all of the decision trees, which results in more accurate predictions than the decision tree itself [34].

2) PARAMETERS FOR MULTI-LABEL CLASSIFIERS

For the classification of resampled datasets, we used eight classifiers with specific parameters. In BR, we used RF as the base classifier with no extension, 10 neighbors and Euclidean as the distance function. In CLR, we used RF as a base classifier with standard voting mode and binary as a type of output. In CC, we used RF as a base classifier. In HOMER, we used RF as a base classifier, BR as a multi-label learner with 3 clusters along with a balanced clustering method. In LP, we used RF as a base classifier. In MLkNN, we used smooth 1, 10 neighbors along with Euclidean as the distance function. In BRkNN, we used extension type none, number of neighbors 10 and Euclidean as a distance function. All of the tests were carried out with cross-validation utilizing five different levels.

Regarding the resampling techniques, we set 25% to resize rates for undersampling and oversampling. In MLeNN, we used 0.5 as the threshold value and ranking label combination in MLSMOTE.

3) PERFORMANCE EVALUATION METRICS

The micro-averaged F-score is used for performance evaluation because it is still considered to be one of the most significant and widely applied metrics in the literature on imbalanced learning [31]. Micro Averaging is computed globally, considering all class labels and instances as given in Equation 9.

$$EM_{micro} = EM\left(\sum_{\sigma=1}^{n} \text{FP}\sigma, \sum_{\sigma=1}^{n} \text{TP}\sigma, \sum_{\sigma=1}^{n} \text{FN}\sigma, \sum_{\sigma=1}^{n} \text{TN}\sigma\right) \tag{9}$$

C. RESULTS

In this section, we will discuss the results achieved from our experimental work. To analyze the result from different aspects, first, we present the MeanIR of datasets before and after the resampling methods. Then we calculated the experimental result using micro averaged F measure. Then we applied the Wilcoxon statistical test for resampling techniques to check out which resampling technique is best for imbalanced MLDs. Then we applied Friedman statistical test to check out the best classifier for the classification of imbalance MLDs. The bold highlighted values present the lower imbalance ratios.

The graphical representation of J.UCS dataset after applying all the resampling techniques is shown in Figure 2. The x-axis represents the number of labels in J.UCS dataset. The y-axis represents the number of examples in a particular class of J.UCS dataset. The original dataset shows the distribution of the total number of examples in the 13 labels which is 1846. After applying LPROS and MLROS, the number of examples increased significantly up to 2583 and 2567 respectively.

While applying MLSMOTE and REMEDIAL-HwR the number of examples is increased slightly up to 1916 and 1858 respectively. With the transformation of LPRUS and MLRUS, the number of examples decreased slightly up to 1488 and 1313. While applying MLeNN and MLTL, the number of examples decreased significantly up to 453 and 571. Moreover, the number of examples remains the same after applying REMEDIAL

The results of the classification are displayed in Table 4 respectively. Following the implementation of the resampling strategies, these findings correspond to the micro-averaged F-score that was calculated from the classification experiments. The most useful results of resampling techniques are presented in bold text and classifiers are in italics.

In Table 4, we have seen the best resampling technique against each classifier column-wise. For BR classifier, MLROS is best as it gives the highest result of 0.7812. For BRkNN classifier, MLeNN is best as it gives the highest result of 0.6340. For CC classifier, LPROS is best as it gives the highest result of 0.7679. For CLR classifier, LPROS is best as it gives the highest result of 0.7587. For HOMER classifier, MLROS is best as it gives the highest result of 0.7823. For LP classifier, MLROS is best as it gives the highest result of 0.7770. For MLkNN classifier, MLeNN is best as it gives the highest result of 0.6597. We have seen the best classifier against each resampling technique row-wise. For the original dataset, CC is best as it gives the highest result of 0.7374. For LPROS, CC classifier is best as it gives the highest result of 0.7679. For LPRUS, CC classifier is best as it gives the highest result of 0.7303. For MLeNN, BR and HOMER classifiers are best as they both give the same highest results of 0.7522. For MLROS, HOMER classifier is best as it gives the highest result of 0.7823. For MLRUS, CC classifier is best as it gives the highest result of 0.7114. For MLSMOTE, CC classifier is best as it gives the highest result of 0.7347. For MLTL, CC classifier is best as it gives the highest result of 0.7411. For REMEDIAL, CC classifier is best as it gives the highest result of 0.7347. For REMEDIAL-HwR, CC classifier is best as it gives the highest result of 0.7347. Overall MLROS gives better results on three classifiers. The results of CC classifier are the best among the seven classifiers.

In Table 5, we have seen the best resampling technique against each classifier column-wise. For BR classifier, LPROS is best as it gives the highest result of 0.6989. For BRkNN classifier, MLROS is best as it gives the highest result of 0.1618. For CC classifier, LPROS is best as it gives highest result of 0.7414. For CLR classifier, MLROS is best as it gives the highest result of 0.6326. For HOMER classifier, LPROS is best as it gives the highest result of 0.6534. For LP classifier, MLTL is best as it gives the highest result of 0.3250. For MLkNN classifier, LPROS is best as it gives highest result of 0.3278. We have seen the best classifier against each resampling technique row-wise. For original dataset, HOMER is best as it gives the highest result of 0.3115. For LPROS, CC classifier is best as it gives the highest result of 0.7414. For LPRUS, HOMER classifier is best as it gives the highest result of 0.2909. For MLeNN, HOMER classifier is best as it gives the highest result of 0.2927. For MLROS, BR classifier is best as it gives the highest result of 0.6577. For MLRUS, HOMER classifier is best as it gives the highest result of 0.3000. For MLSMOTE, HOMER classifier is best as it gives the highest result of 0.3168. For MLTL, CC classifier is best as it gives the highest result of 0.3559. For REMEDIAL, HOMER classifier is best as it gives the highest result of 0.1627. For REMEDIAL-HwR, HOMER classifier is best as it gives the highest result of 0.1697. Overall LPROS gives better results on four classifiers. The results of HOMER classifier are the best among the seven classifiers.

In Table 6, we have seen the best resampling technique against each classifier column-wise. For BR classifier, LPROS is best as it gives the highest result of 0.8326. For BRkNN classifier, MLTL is best as it gives the highest result of 0.5603. For CC classifier, LPROS is best as it gives the highest result of 0.8323. For CLR classifier, LPROS is best as it gives the highest result of 0.7912. For HOMER classifier, LPROS is best as it gives the highest result of 0.7832. For LP classifier, LPROS is best as it gives highest result of 0.8814. For MLkNN classifier, MLTL is best as it gives the highest result of 0.6016. We have seen the best classifier against each resampling technique row-wise. For the original dataset, CC is best as it gives the highest result of 0.6347. For LPROS, LP classifier is best as it gives the highest result of 0.8814. For LPRUS, CC classifier is best as it gives highest result of 0.5928. For MLeNN, CC classifier is best as it gives highest result of 0.6347. For MLROS, HOMER classifier is best as it gives the highest result of 0.6878. For MLRUS, CLR classifier is best as it gives the highest result of 0.6623. For MLSMOTE, CC classifier is best as it gives the highest result of 0.6288. For MLTL, BR classifier is best as it gives the highest result of 0.6699. For REMEDIAL, CLR classifier is best as it gives the highest result of 0.6382. For REMEDIAL-HwR, CC classifier is best as it gives the highest result of 0.6288. Overall LPROS gives better results on five classifiers. The results of CC classifier are best among the seven classifiers.

In Table 7, we have seen the best resampling technique against each classifier column-wise. For BR classifier, LPROS is the best as it gives the highest result of 0.7621. For BRkNN classifier, REMEDIAL-HwR is the best as it gives the highest result of 0.9000. For CC classifier, LPROS is best as it gives the highest result of 0.7736. For CLR classifier, LPROS is best as it gives the highest result of 0.7586. For HOMER classifier, LPROS is best as it gives the highest result of 0.7362. For LP classifier, LPROS is the best as it gives the highest result of 0.7360. For MLkNN classifier, LPROS is best as it gives the highest result of 0.6053. We have seen the best classifier against each resampling technique row-wise. For the original dataset, HOMER is best as it gives the highest result of 0.4906. For LPROS, CC classifier is best as it gives the highest result of 0.7736. For LPRUS, HOMER classifier is best as it gives the highest result of 0.4789. For MLeNN, BR classifier is best as it gives the

TABLE 3. Resampling Techniques Applied on before/after Dataset.

Resampling Techniques	Datasets				
	ACM	Bibtex	Eurlex	J.UCS	Reuters
Original	1.191	12.498	536.976	8.072	54.081
LPROS	1.058	12.410	219.058	4.628	29.363
LPRUS	1.116	12.561	398.565	6.919	45.908
MLeNN	1.172	11.511	536.976	8.894	56.218
MLROS	1.015	10.908	112.721	3.957	12.318
MLRUS	1.020	13.121	782.148	8.210	74.004
MLSMOTE	1.150	11.223	410.925	6.060	50.457
MLTL	1.189	35.676	308.378	19.580	73.139
Remedial	1.191	12.498	536.976	8.072	50.081
Remedial-HwR	1.180	11.608	410.925	8.065	53.297

FIGURE 2. Visualization of J.UCS Dataset with all Resampling Techniques.

highest result of 0.6205. For MLROS, CC classifier is best as it gives the highest result of 0.6803. For MLRUS, HOMER classifier is best as it gives the highest result of 0.4883. For MLSMOTE, HOMER classifier is best as it gives the highest result of 0.4612. For MLTL, BR classifier is best as it gives the highest result of 0.6183. For REMEDIAL, HOMER classifier is best as it gives the highest result of 0.2660. For REMEDIAL-HwR, HOMER classifier is best as it gives the highest result of 0.2370. Overall LPROS gives better results on six classifiers. The results of HOMER classifier are the best among the seven classifiers.

In Table 8, we have seen the best resampling technique against each classifier column-wise. For BR classifier, LPROS is best as it gives the highest result of 0.8038. For

TABLE 4. Experimental Results for ACM.

Resampling Techniques	Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.6737	0.5108	0.7374	0.6718	0.7247	0.7278	0.6295
LPROS	0.7673	0.5961	0.7679	0.7587	0.7673	0.7613	0.6351
LPRUS	0.7188	0.5971	0.7303	0.7106	0.7188	0.7148	0.6201
MLeNN	0.7522	0.6340	0.7438	0.7425	0.7522	0.7431	0.6597
MLROS	0.7812	0.6007	0.7511	0.7160	0.7823	0.7770	0.6151
MLRUS	0.7083	0.5643	0.7114	0.7005	0.7088	0.6995	0.5973
MLSMOTE	0.7247	0.6084	0.7347	0.7153	0.7247	0.7278	0.6295
MLTL	0.7322	0.6101	0.7411	0.7218	0.7322	0.7208	0.6365
REMEDIAL	0.7247	0.6084	0.7347	0.7153	0.7247	0.7278	0.6295
REMEDIAL-HwR	0.7247	0.6084	0.7347	0.7153	0.7247	0.7278	0.6295

TABLE 5. Experimental Results for Bibtex.

Resampling Techniques	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.2586	0.1233	0.2498	0.2515	0.3115	0.1759	0.2149
LPROS	0.6989	0.1442	0.7414	0.5234	0.6534	0.1780	0.3278
LPRUS	0.2401	0.0930	0.2348	0.2311	0.2909	0.1695	0.1945
MLeNN	0.2214	0.1082	0.2178	0.2089	0.2927	0.1670	0.2104
MLROS	0.6577	0.1618	0.6547	0.6326	0.5778	0.1820	0.3217
MLRUS	0.2490	0.1199	0.2089	0.2465	0.3000	0.1982	0.2130
MLSMOTE	0.2844	0.0573	0.2776	0.2870	0.3168	0.2025	0.0644
MLTL	0.3433	0.1218	0.3559	0.2583	0.3010	0.3250	0.3115
REMEDIAL	0.0719	0.0359	0.0709	0.0823	0.1627	0.0876	0.1332
REMEDIAL-HwR	0.0997	0.0184	0.0969	0.0994	0.1697	0.1056	0.0221

BRkNN classifier, MLROS is best as it give the highest result of 0.4868. For CC classifier, LPROS is best as it gives the highest result of 0.8014. For CLR classifier, LPROS is best as it gives the highest result of 0.2490. For HOMER classifier, MLROS is best as it gives the highest result of 0.4980. For LP classifier, LPROS is best as it gives the highest result of 0.7487. For MLkNN classifier, MLROS is best as it gives the highest result of 0.5127. We have seen the best classifier against each resampling technique row-wise. For the original dataset, LP is best as it gives the highest result of 0.4584. For LPROS, BR classifier is best as it gives the highest result of 0.8038. For LPRUS, CC classifier is best as it gives the highest result of 0.4251. For MLeNN, HOMER classifier is best as it gives the highest result of 0.4812. For MLROS, CC classifier is best as it gives the highest result of 0.5638. For MLRUS, CC classifier is best as it gives the highest result of 0.4812. For MLSMOTE, HOMER classifier is best as it gives the highest result of 0.4531. For MLTL, CC classifier is best as it gives the highest result of 0.4941. For REMEDIAL, MLkNN classifier is best as it gives the highest result of 0.3354. For REMEDIAL-HwR, MLkNN classifier is best as it gives the highest result of 0.3396. Overall LPROS gives

TABLE 6. Experimental Results for Eurlex.

Resampling Techniques	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.6329	0.5217	0.6347	0.3979	0.6282	0.5213	0.5634
LPROS	0.8326	0.5093	0.8323	0.7912	0.7832	0.8814	0.5479
LPRUS	0.5909	0.4841	0.5928	0.5919	0.5924	0.5281	0.5270
MLeNN	0.6329	0.5217	0.6347	0.6161	0.6218	0.6132	0.5634
MLROS	0.6712	0.5585	0.6704	0.6814	0.6878	0.5623	0.5948
MLRUS	0.6592	0.5419	0.6599	0.6623	0.6012	0.5918	0.5868
MLSMOTE	0.6281	0.5231	0.6288	0.6513	0.5918	0.5742	0.5621
MLTL	0.6699	0.5603	0.6671	0.6123	0.6142	0.6231	0.6016
REMEDIAL	0.6329	0.5217	0.6347	0.6382	0.5738	0.5346	0.5634
REMEDIAL-HwR	0.6281	0.5231	0.6288	0.6211	0.5228	0.5713	0.5621

TABLE 7. Experimental Results for J.UCS.

Resampling Techniques	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.4080	0.1876	0.4365	0.4170	0.4906	0.4248	0.3568
LPROS	0.7621	0.1976	0.7736	0.7586	0.7362	0.7360	0.6053
LPRUS	0.3632	0.1804	0.3989	0.4097	0.4789	0.4178	0.3343
MLeNN	0.6205	0.4536	0.5936	0.6145	0.5928	0.5500	0.5477
MLROS	0.6625	0.3604	0.6803	0.6656	0.6618	0.6396	0.5384
MLRUS	0.4048	0.2322	0.4407	0.4161	0.4883	0.4257	0.3419
MLSMOTE	0.3718	0.1544	0.3967	0.3809	0.4612	0.4041	0.3167
MLTL	0.6183	0.3684	0.6215	0.6068	0.6098	0.5918	0.5462
REMEDIAL	0.1929	0.1032	0.1920	0.2022	0.2660	0.2346	0.1391
REMEDIAL-HwR	0.2002	0.9000	0.2022	0.2051	0.2610	0.2370	0.1469

better results on four classifiers. The results of CC classifier are the best among the seven classifiers.

D. STATISTICAL ANALYSIS

In this section, we will analyze the resampling techniques that improve the classification results and which resampling technique is best by using Wilcoxon Test stating as a null hypothesis that the F-score is higher after using each resampling technique with 0.05 as p-value threshold. For the best classifier of classification, the ranking principle is calculated for multi-label classifiers by using the Friedman test in which the results of classifiers are ranked and an average rank is calculated against each dataset.

In Table 10, we have shown the z-score and p-value against each classifier. We have observed that LPROS, LPRUS, MLeNN, MLROS, MLTL, REMEDIAL and REMEDIAL-HwR resampling techniques improved the results of classification significantly as their z-score is negative and p-value is less than threshold which is 0.05. Regarding MLRUS and MLSMOTE, as p-value crossed the threshold the null

TABLE 8. Experimental Results for Reuters.

Resampling Techniques	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.4372	0.3783	0.4527	0.1336	0.3527	0.4584	0.4157
LPROS	0.8038	0.3855	0.8014	0.2490	0.3921	0.7487	0.4484
LPRUS	0.4147	0.3610	0.4251	0.1348	0.3282	0.3930	0.3956
MLeNN	0.4448	0.3830	0.4550	0.1364	0.4812	0.4271	0.4327
MLROS	0.5631	0.4868	0.5638	0.1318	0.4980	0.5108	0.5127
MLRUS	0.4630	0.4039	0.4812	0.1518	0.4212	0.4564	0.4409
MLSMOTE	0.4286	0.3754	0.4396	0.1333	0.4531	0.4520	0.4182
MLTL	0.4852	0.4185	0.4945	0.1575	0.4621	0.4649	0.4664
REMEDIAL	0.3037	0.3090	0.3090	0.1040	0.3012	0.3279	0.3354
REMEDIAL-HwR	0.2970	0.3277	0.3076	0.1076	0.3076	0.3187	0.3396

TABLE 9. Experimental Results for Tmc.

Resampling Techniques	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
Original	0.7864	0.5592	0.7885	0.7865	0.6995	0.7338	0.6389
LPROS	0.8853	0.6357	0.8857	0.8845	0.8663	0.7420	0.7373
LPRUS	0.7713	0.6057	0.7725	0.7718	0.7653	0.7134	0.6449
MLeNN	0.8292	0.6649	0.8302	0.8267	0.8142	0.7825	0.6994
MLROS	0.8656	0.6166	0.8659	0.8638	0.8501	0.7152	0.6877
MLRUS	0.7492	0.5992	0.7514	0.7512	0.7467	0.6905	0.2760
MLSMOTE	0.7680	0.3569	0.7684	0.7683	0.7563	0.6829	0.4984
MLTL	0.8739	0.6386	0.8752	0.8678	0.8552	0.8432	0.6724
REMEDIAL	0.3863	0.1454	0.3845	0.2190	0.2321	0.3711	0.2870
REMEDIAL-HwR	0.3857	0.1924	0.3810	0.3012	0.2991	0.3695	0.2093

TABLE 10. Wilcoxon Statistical Test for Micro Averaged F-score Result.

Resampling Techniques	z-Score	p-value
LPROS	-5.4829	0.00001
LPRUS	-2.6445	0.0083
MLeNN	-3.8431	0.00012
MLROS	-5.5204	0.00001
MLRUS	-1.0753	0.28014
MLSMOTE	-0.9768	0.32708
MLTL	-4.3513	0.00001

hypothesis is rejected by Wilcoxon Test. Due to negative z-score, there are not enough evidences to claim that these resampling techniques didn't improve the results. In general, considering all resampling techniques, we conclude that MLROS is the most effective resampling technique as it has the lowest z-score and p-value. In Table 11, we observed the results of the ranking principle on each dataset and then the average rank. We have observed that BRkNN is the best classifier for ACM, Bibtex, Eurlex J.UCS and Tmc datasets. For Reuters, CLR is the best classifier. In general, considering all datasets, we conclude that BRkNN is the best classifier. The graphical visualization of the ranking of classifiers is shown in Figure 3.

Apparently, in Table 11, we have observed that LPROS is the best resampling technique and CC is the best classifier for imbalanced MLDs. While according to statistical evaluation, MLROS is the best resampling technique and BRkNN is the best classifier for imbalanced MLDs. There-

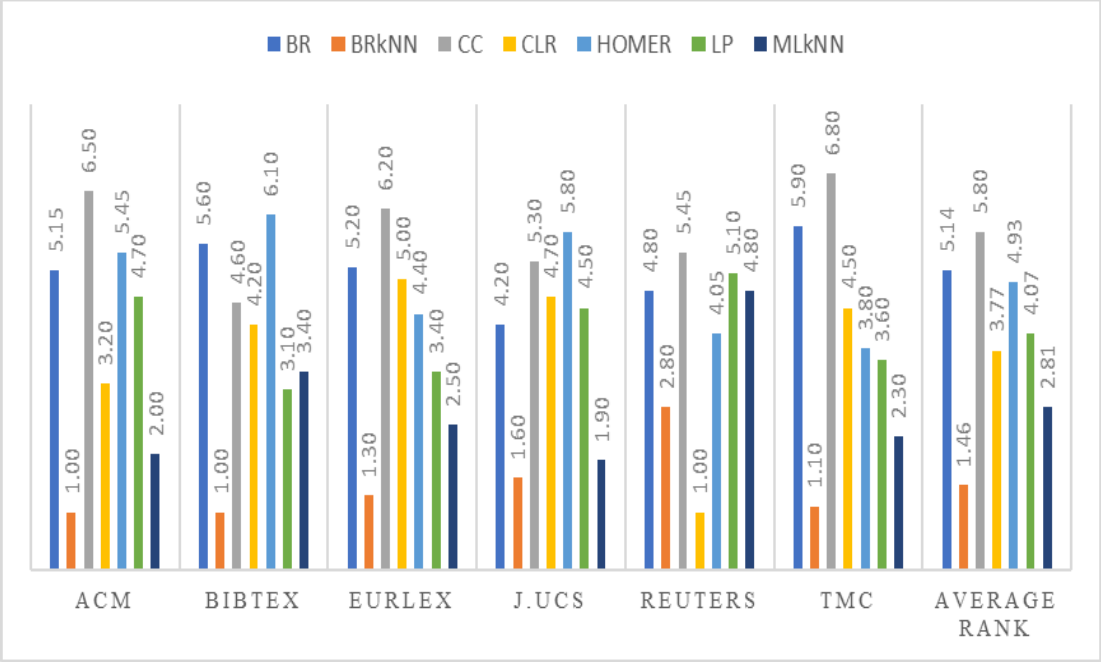


FIGURE 3. Ranking of Multi-Label Classifiers using Friedman Test.

TABLE 11. Classifiers Rank using Friedman Test.

Datasets	Classification Classifiers						
	BR	BRkNN	CC	CLR	HOMER	LP	MLkNN
ACM	5.15	1.00	6.50	3.20	5.45	4.70	2.00
Bibtex	5.60	1.00	4.60	4.20	6.10	3.10	3.40
Eurlex	5.20	1.30	6.20	5.00	4.40	3.40	2.50
J.UCS	4.20	1.60	5.30	4.70	5.80	4.50	1.90
Reuters	4.80	2.80	5.45	1.00	4.05	5.10	4.80
Tmc	5.90	1.10	6.80	4.50	3.80	3.60	2.30
Average Rank	5.14	*1.46*	5.80	3.77	4.93	4.07	2.81

fore, we have concluded our results on the basis of statistical evaluation. Therefore, the results are concluded based on statistical evaluation.

Document categorization is widely regarded as one of the most important subfields that fall under the umbrella of data mining. The documents are organized according to the predetermined hierarchy of categories that are displayed in the form of taxonomy. Because of the nature of the manuscript and the outsized taxonomy, this particular issue has emerged as a topic of intense interest for academic investigation. In addition, its significance can be gauged by counting the number of different contexts in which the findings of this study can be put to beneficial use. Through the course of the problem, the various aspects of classification, such as single-label, multi-label, and transformation classification, are carefully examined. Text classification has emerged as a fundamental requirement for an increasing number of applications over the past few years. When a label is assigned to more than one class, the taxonomy becomes more difficult to understand.

The topic of assigning labels to scientific documents is another significant research area. At present, the authoring scientific documents involves the application of labels to their papers in a manual fashion. The scientific papers are versatile enough to fit into a variety of categories.

Two approaches are available for solving the classification challenge presented by multi-label documents. Conversion of multi-label transformation problems and algorithm adaptation methodologies are among the techniques. On six different actual datasets, the problem of the transformation was implemented by using resampling techniques and thoroughly evaluated. We employed nine resampling techniques to transform the imbalanced MLDs and rebalanced them for classification. These techniques are LPROS, LPRUS, MLROS, MLRUS, MLTL, MLeNN, MLSMOTE, REMEDIAL and

REMEDIAL-HwR. For classification, we used seven multi label classifiers such as BR, LP, CLR, CC, HOMER, MLkNN and BRkNN.

From the results of various experiments as well as statistical evaluation, we concluded that MLROS is the most effective resampling technique for imbalanced MLDs. The BRkNN is a better classifier for the classification of imbalanced MLDs.

In the future, the research can be expanded to balance the dataset from external knowledge-based sources. As we have seen that through resampling techniques, we may lose important information as well as the addition of irrelevant information.

V. CONCLUSION AND FUTURE WORK

Text classification has developed in recent years as a fundamental requirement for a growing variety of applications. When a label is assigned to more than one class, the taxonomy becomes more difficult to understand. Regarding the topic of assigning labels to scientific documents, this is another significant research area. At the present time, the authoring of scientific documents involves the application of labels to their papers in a manual fashion. The scientific papers are versatile enough to fit into a variety of categories.

There are two ways available for resolving the classification problem posed by documents with multiple labels. Conversion of multi label transformation problems and algorithm adaptation methodologies are among the techniques. On six different actual datasets, the problem of the transformation was implemented by using resampling techniques and thoroughly evaluated. We employed nine resampling techniques to transform the imbalanced MLDs and rebalanced them for classification. These techniques are LPROS, LPRUS, MLROS, MLRUS, MLTL, MLeNN, MLSMOTE, REMEDIAL and REMEDIAL-HwR. For classification, we used seven multi label classifiers such as BR, LP, CLR, CC, HOMER, MLkNN, and BRkNN. From the results of various experiments as well as statistical evaluation, we concluded that MLROS is the most effective resampling technique for imbalanced MLDs. The BRkNN is a better classifier for the classification of imbalanced MLDs. In future, the research can be expanded to balance the dataset from external knowledge-based sources. As we have seen that through resampling techniques, we may lose important information as well as the addition of irrelevant information.

REFERENCES

[1] R. Kumari and S. K. Srivastava, "Machine learning: A review on binary classification," *Int. J. Comput. Appl.*, vol. 160, no. 7, pp. 11–15, Feb. 2017.

[2] M. Aly, "Survey on multiclass classification methods," *Neural Netw.*, vol. 19, pp. 1–9, Nov. 2005.

[3] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. J. Data Warehousing Mining*, vol. 3, pp. 1–13, Jul. 2007.

[4] T. Ali and S. Asghar, "Multi-label scientific document classification," *J. Internet Technol.*, vol. 19, no. 6, pp. 1707–1716, 2018.

[5] H. Borko and M. Bernick, "Automatic document classification," *J. ACM*, vol. 10, no. 2, pp. 151–162, Apr. 1963.

[6] A. Masmoudi, H. Bellaaj, K. Drira, and M. Jmaiel, "A Co-training-based approach for the hierarchical multi-label classification of research papers," *Expert Syst.*, vol. 38, no. 2, Jun. 2021, Art. no. e12613.

[7] N. Coulter, J. French, E. Glinert, T. Horton, N. Mead, R. Rada, A. Ralston, C. Rodkin, B. Rous, and A. Tucker, "Computing classification system 1998: Current status and future maintenance report of the CCS Update Committee," *Comput. Rev.*, vol. 39, pp. 1–62, Feb. 1998.

[8] T. Ali and S. Asghar, "Efficient label ordering for improving multi-label classifier chain accuracy," *J. Nat. Sci. Found. Sri Lanka*, vol. 47, no. 2, p. 175, May 2019.

[9] A. D. Bosco, R. Vieira, B. Zanotto, and A. Paula, "Ontology based classification of electronic health records to support value-based health care," in *Proc. Brazilian Conf. Intell. Syst.*, 2021, pp. 359–371.

[10] A. More, "Survey of resampling techniques for improving classification performance in unbalanced datasets," 2016, *arXiv:1608.06048*.

[11] Y. Yan, Y. Wang, W.-C. Gao, B.-W. Zhang, C. Yang, and X.-C. Yin, "LSTM multi-label ranking for document classification," *Neural Process. Lett.*, vol. 47, pp. 117–138, May 2017.

[12] W. Huang, E. Chen, Q. Liu, Y. Chen, Z. Huang, Y. Liu, Z. Zhao, D. Zhang, and S. Wang, "Hierarchical multi-label text classification: An attention-based recurrent network approach," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1051–1060.

[13] M. Zhang, Y. Li, H. Yang, and X. Liu, "Towards class-imbalance aware multi-label learning," *IEEE Trans. Cybern.*, vol. 52, no. 6, pp. 4459–4471, Jun. 2022.

[14] S. Salunkhe and K. Bhowmick, "Comparing different machine learning techniques for classifying multi label data," *J. Univ. Shanghai Sci. Technol.*, vol. 23, no. 1, pp. 39–44, Jan. 2021.

[15] A. P. Santos and F. Rodrigues, "Multi-label hierarchical text classification using the ACM taxonomy," in *Proc. 14th Portuguese Conf. Artif. Intell. (EPIA)*, 2009, pp. 553–564.

[16] M. T. Afzal, W. Balke, H. Maurer, and N. Kulathuramaiyer, "Improving citation mining," in *Proc.1st Int. Conf. Networked Digit. Technol.*, 2009, pp. 116–121.

[17] M.-L. Zhang, Y.-K. Li, X.-Y. Liu, and X. Geng, "Binary relevance for multi-label learning: An overview," *Frontiers Comput. Sci.*, vol. 12, no. 2, pp. 191–202, 2018.

[18] C. Li and G. Shi, "Improvement of learning algorithm for the multi-instance multi-label RBF neural networks trained with imbalanced samples," *J. Inf. Sci. Eng.*, vol. 29, no. 4, pp. 765–776, 2013.

[19] J. He, H. Gu, and W. Liu, "Imbalanced multi-modal multi-label learning for subcellular localization prediction of human proteins with both single and multiple sites," *PLoS One*, vol. 7, no. 6, Jun. 2012, Art. no. e37155.

[20] K. Chen, B.-L. Lu, and J. T. Kwok, "Efficient classification of multi-label and imbalanced data using min-max modular classifiers," in *Proc. IEEE Int. Joint Conf. Neural Netw. Proc.*, Jul. 2006, pp. 1770–1775.

[21] K. W. Sun and C. H. Lee, "Addressing class-imbalance in multi-label learning via two-stage multi-label hypernetwork," *Neurocomputing*, vol. 266, pp. 375–389, Nov. 2017.

[22] Y. Xie, D. Li, D. Zhang, and H. Shuang, "An improved multi-label relief feature selection algorithm for unbalanced datasets," in *Proc. Int. Conf. Intell. Interact. Syst. Appl.*, pp. 141–151, 2017.

[23] F. Luo, W. Guo, and G. Chen, "Addressing imbalance in weakly supervised multi-label learning," *IEEE Access*, vol. 7, pp. 37463–37472, 2019.

[24] G. Wu, Y. Tian, and D. Liu, "Cost-sensitive multi-label learning with positive and negative label pairwise correlations," *Neural Netw.*, vol. 108, pp. 411–423, Dec. 2018.

[25] P. Cao, X. Liu, D. Zhao, and O. Zaiane, "Cost sensitive ranking support vector machine for multi-label data learning," in *Proc. Int. Conf. Hybrid Intell. Syst.*, 2016, pp. 244–255.

[26] Z. A. Daniels and D. N. Metaxas, "Addressing imbalance in multi-label classification using structured Hellinger forests," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1826–1832.

[27] M. A. Tahir, J. Kittler, and A. Bouridane, "Multilabel classification using heterogeneous ensemble of multi-label classifiers," *Pattern Recognit. Lett.*, vol. 33, no. 5, pp. 513–523, Apr. 2012.

[28] P. Sadhukhan and S. Palit, "Reverse-nearest neighborhood based over-sampling for imbalanced, multi-label datasets," *Pattern Recognit. Lett.*, vol. 125, pp. 813–820, Jul. 2019.

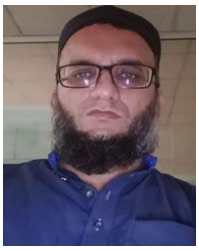
[29] F. Charte, A. Rivera, M. J. D. Jesus, and F. Herrera, "A first approach to deal with imbalance in multi-label datasets," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.*, 2013, pp. 150–160.

[30] F. Charte, A. Rivera, M. Jesus, and F. Herrera, "Concurrence among imbalanced labels and its influence on multilabel resampling algorithms," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.*, 2014, pp. 110–121.
[31] F. Charte, A. Rivera, M. Jesus, and F. Herrera, "MLeNN: A first approach to heuristic multilabel undersampling," in *Proc. Int. Conf. Intell. Data Eng. Automated Learn.*, 2014, pp. 1–9.
[32] R. Pereira and Y. Carlos, "MLTL: A multi-label approach for the Tomek Link undersampling algorithm," *Neurocomputing*, vol. 383, pp. 95–105, Mar. 2020.
[33] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *J. Mach. Learn.*, vol. 85, no. 3, pp. 333–359, Dec. 2011.
[34] E. Spyromitros, G. Tsoumakas, and J. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Proc. Hellenic Conf. Artif. Intell.* Berlin, Germany: Springer, 2008, pp. 1–6.
[35] N. Jalal, A. Mehmood, G. S. Choi, and I. Ashraf, "A novel improved random forest for text classification using feature ranking and optimal number of trees," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 6, pp. 2733–2742, Jun. 2022.

AIMAN HAFEEZ received the M.I.T. and M.S.C.S. (Hons.) degrees with specialization in artificial intelligence from the University Institute of Information Technology (UIIT), Arid Agriculture University, Rawalpindi, Pakistan, in 2018 and 2022, respectively. She is currently a Computer Science Lecturer with the UIIT, PMAS Arid Agriculture University. Her research interests include machine learning, data mining, and natural language processing.

TARIQ ALI received the master's degree in computer science from Muhammad Ali Jinnah University, Islamabad, Pakistan, in 2009, and the Ph.D. degree in computer science from Abasyn University Islamabad, in 2019. He is currently an Assistant Professor with the University Institute of Information Technology (UIIT), PMAS Arid Agriculture University. His research interests include document classification, multi label classification, machine learning, and deep learning.

ASIF NAWAZ received Ph.D. degree from International Islamic University, Islamabad, in 2019. He is currently an Assistant Professor with the University Institute of Information Technology (UIIT), PMAS Arid Agriculture University, Rawalpindi. He is also a Potential Researcher in the field of data mining, machine learning, social media analysis, and text mining. He has published more than 40 research articles in well-reputed international journals.



SAIF UR REHMAN received the M.C.S. degree (Hons.) from the Institute of Computing and Information Technology, Gomal University, Dera Ismail Khan, Pakistan, in 2005, and the M.S. and Ph.D. degrees in computer science, in 2010 and 2019, respectively. He is currently an Assistant Professor with the University Institute of Information Technology (UIIT), PMAS Arid Agriculture University, Rawalpindi, Pakistan. He has more than eight years of industry experience and involved in Java, ASP.Net, C#, and Oracle. He is an advisor in PPSC, and Examiner in PPSC and KP-PSC. He has published five book chapters and more than 60 research articles in different renowned international Q1 and Q2 journals. His research interests include data mining, graph mining, social graph analysis, and big data analytics.

AZHAR IMRAN MUDASIR (Member, IEEE) received the master's degree in computer science from the University of Sargodha, Pakistan, and the Ph.D. degree in software engineering from the Beijing University of Technology, China. From 2012 to 2017, he was a Senior Lecturer with the Department of Computer Science, University of Sargodha. He is currently an Assistant Professor with the Department of Creative Technologies, Faculty of Computing and Artificial Intelligence, Air University, Islamabad, Pakistan. He is also a Renowned Expert in image processing, healthcare informatics, and social media analysis. He contributed to more than 40 research articles in well-reputed international journals and conferences. He is an editorial member and a reviewer of various journals, including IEEE Access, *Cancers* (MPDI), IGI Global, and *Journal of Imaging*. He has more than nine years of national and international academic experience as a full-time Faculty, teaching courses in software engineering, and core computing. His research interests include image processing, social media analysis, medical image diagnosis, machine learning, and data mining.

ABDULAZIZ A. ALSULAMI received the master's degree in computer science from Western Michigan University, Kalamazoo, MI, USA, in 2017, and the Ph.D. degree in computer and information systems engineering from Tennessee State University, Nashville, TN, USA, in 2021. He is currently an Assistant Professor with the Department of Information Systems, King Abdulaziz University, Saudi Arabia. His research interests include the security of the cyber-physical systems (CPS), the security of the Internet of Things (IoT), artificial intelligence (AI), and machine learning (ML).

ALI ALQAHTANI received the Ph.D. degree in computer engineering from Oakland University, Rochester, MI, USA, in 2020. He is currently an Assistant Professor with Najran University (NU). His research interests include machine learning in general and deep learning in image and signal processing, wireless vehicular networks (VANETs), wireless sensor networks, and cyber-physical systems.