# Multi-label Scientific Document Classification

Tariq Ali[1], Sohail Asghar[2]

[1] Department of computer science, Abasyn University, Islamabad, Pakistan
[2] Department of computer science, COMSATS University, Islamabad, Pakistan
tariq.ali@uaar.edu.pk, sohail.asghar@comsats.edu.pk

## Abstract

Scientific document label identification is a significant research area having numerous applications like digital libraries. The author assigns a category or categories to their document manually. Likewise, categories are structured in taxonomy in the form of tree such as ACM CCS. The dilemma becomes more complex when a document belongs to multiple categories. The problem of manual assignment becomes more complicated when the number of expected labels increases. Moreover, the accession schemes are insufficient for solutions with higher accuracy on real scientific document datasets. One way to handle the multi-label classification is to change the problem into a single-label classification. Another way is the variation of the algorithm to handle multi-label classification. The focus of our research is on conversion. Moreover, we propose a solution stimulated from the particle swarm optimization algorithm that can consign a label from the taxonomy. A set of similarity measures is evaluated as well for documentation relatedness that are used in the proposed approach. The designed solution is evaluated on two documents dataset that are retrieved from J. UCS and ACM with an average accuracy of 77 percent as compared to the state of the art algorithms.

**Keywords:** Digital libraries, Multi-label classification, PSO, Text similarity

## 1 Introduction

Automated text categorization is becoming more important with a rapid increase in the number of documents on the web [1]. The research community is generating a large number of scientific documents. These document's are then available and can be accessed over the internet using search engines, digital libraries and citation indexes. There is a need to categorize and manage the enormous amount of documents (data) into a specific hierarchy or taxonomy [2]. Likewise, the level of documents relevance to a node in taxonomy will support in exploring the user's appropriate data in a proficient manner. The precision of gaining or retrieval of data is dependent on the precise organization of the documents [3]. Furthermore, the retrieval of data and correct taxonomy facilitates in evaluating development, finding expertise and the pertinent document recommender system. As to talk about classification, it is a two level approach. The foremost level initiates a model from the training set of instances while the succeeding level makes sure about the correctness of the classifier [4]. There are various methods for document classification, like Decision Tree [5], Naive Bays Classifier [4], Particle Swarm Optimization [6], Support Vector Machine [7], Term Frequency and Inverse Document Frequency based approaches [8].

An important step in document classification is category identification. At present, the initiators of scientific publications classify the appropriate categories (further written as the category/ies) to their documents manually. The general classification used in the research area is the Association for Computing Machinery Computing Classification System (ACM CCS) [9]. Category identification of a document is a complex job for a novel researcher, particularly if it relates to numerous fields of study. The manual label assignment is becoming complicated due to multi-label assignment in a large number of categories available in a taxonomy. Moreover, work in one domain overlap with others. A document can be allocated to several classes in multi-label document classification. This research associates the gap between users towards identifying the proper document category and proposes a possible categorization to the author's work without human intercession.

Existing text classification schemes for scientific publication lack in handling multi-label assignment of labels from taxonomy. Most of the existing schemes are evaluated on synthetic datasets. There is a need to assign multi-labels to a document from a given taxonomy. The solution proposed in this paper is particularly for the classification of scientific publications. The datasets used for evaluation are retrieved from Journal of Universal Computer Science [10] and ACM computing classification [11]. In the first phase of the proposed methodology, keywords and title of scientific publications are retrieved which are

*Corresponding Author: Tariq Ali; E-mail: tariq.ali@uaar.edu.pk
DOI: 10.3966/160792642018111906008

further pre-processed by applying stop words removal and stemming algorithm. For evaluation, the ACM computing classification scheme is used as a taxonomy. In the next phase the multi-label dataset is transformed into the single-label dataset by applying four different conversion techniques. In third phase an algorithm is proposed which predicts the class label of each test instance. The proposed solution is inspired from the well known particle swarm optimization (PSO) due to high numbers of features. PSO is better than other swarm based techniques, due to high convergence ratio. PSO provides both the local optima as well as a global solution[6]. Closely related documents within a category are treated as local optima and the chosen category is considered as global best in the problem domain. In the proposed algorithm, four different measures are being evaluated for text data. Final analysis is performed on the obtained results using different similarity measures and comparison is performed with the current state of the art algorithms.

## 2 Literature Review

There are two ways to deal with multi-label classification problem, one is the problem transformation method and another one is the algorithm adaptation method [12]. The focus of this research is on transformation method. In label transformation two main techniques exist; one is label power set and another is binary relevance. In label power set method a compound class with all possible combinations of available classes is defined while in binary relevance method an independent classifier for each class is defined. An algorithm based on binary Bayesian classifier and Bayesian network is proposed for multi-label classification [13]. In this approach first a Bayesian network is developed that model relationship between the class variable that learns from the data. In the second phase, classifier chain is developed. The results are generated by changing the order of classifiers on benchmark multi-label datasets. For a multi-label problem with *d* classes the Bayesian chain classifier uses *d* classifiers, one for each class, that are linked into a chain as shown in Equation 1 for *d* classes with *l* attributes [13].

$$p(C|x)=\prod_{i=1}^{d} p(C|i\ |pa(C|i\ ),x) \quad \textbf{(1)}$$

Binary Relevance method (BR) decomposes the multi-label classification MLC problem to the SLC problem. Classifier for each class is trained as modelled in Equation 2. This technique does not consider the label dependencies between labels [14].

$$Y \equiv [y_i,...,y_l] = [h_1(x),...,h_l(x)] where\ y_j \in \{0,1\} \quad \textbf{(2)}$$

To overcome the problem of label dependencies, alternative approaches exist, such as label Powerset (LP) method. A single classifier is learnt for each class label as $H:x \rightarrow P(L)$ where P represents the power set of labels L. Drawback of the approach is that it increases the label set exponentially.

Classifier chain is a recent technique to handle label dependencies. The classifier chain model consist of L classifier where each classifier is associated with a label. Training algorithms for an instance (x, S) where x is an instance and S is a subset of L represented by a binary feature vector $(l_1, l_2, ...., l_{|L|}) \in \{0,1\}^{|L|}$ [15].

Dependencies between labels are executed by defining a chain of classifier. The order of classifiers being executed play important role in overall accuracy, however, determining the correct order is an overhead. Low accuracy of exiting Algorithms i.e., BCC [13], BR [16], MCC [14], Rakel [17], CC [15], DPPNN [4], BRq [18] and CDT [19] on J.UCS dataset using Meka [20] is in Table 1.

**Table 1.** Accuracy of multi-label classifier on J.UCS dataset

| Classifier | Accuracy Enron Dataset | Accuracy J.UCS Dataset (All categories) | Accuracy J.UCS Dataset (5 categories) |
|---|---|---|---|
| BCC | 0.403 | 0.277 | 0.490 |
| BR | 0.388 | 0.319 | 0.455 |
| MCC | 0.414 | 0.331 | 0.505 |
| Rakel | 0.027 | 0.191 | 0.287 |
| CC | 0.414 | 0.331 | 0.506 |
| DPPNN | 0.321 | 0.261 | 0.317 |
| BRq | 0.434 | 0.293 | 0.447 |
| CDT | 0.40 | 0.336 | 0.460 |

Meka tool for MLC which is an extension to Weka tool is available [20]. This tool has a set of well known MLC algorithms using transformation techniques, an experiment was performed on the J.UCS dataset [10] having a collection of metadata associated with the research papers. It contains 1,460 instances labelled in 13 categories. Another dataset used in the experimentation is the Enron dataset [21], which is a collection of emails communicated between different Enron employees. It contains 1,702 emails categorized in 53 different classes [21]. Experimental results on text dataset are summarized in Table 1. Table 1 shows that most of the classifiers have low accuracy. The low accuracy is due to strict evaluation measure for MLC and due to sparseness in the dataset. Results of considering all categories and categories with sufficient number of documents are figured out in Table 1 on J.UCS dataset. Removing categories with less number of documents improves the classifier accuracy as compared in Table 1. Same problems of multi-label classification MLC are also reported in [22-24]. MLC for Czech news are reported with low accuracy. In this approach they have used the already available classifiers for multi-label news articles. High accuracy results for the same dataset are reported [25-

26], but the evaluation measures used are of SLC. They have reported 91% accuracy, but even the sample results reported in their paper do not depict 10 percent accuracy. The results are only compared with the cited results in the literature without implementation.

Ontology based multi-label document classification for economic article has been proposed [27]. They used an ontology constructed from the economics articles in the text classification process. In this approach they evaluated MLC using the transformation of a multi-label to a single-label, multi-label algorithms and hierarchical models. The authors claim the potential for the improvement of accuracy in all the three perspectives.

PSO has been applied for feature identification and existing single-label classifier are used with improved accuracy on numeric datasets [28-29]. The methodology to deal MLC and evaluation parameters are missing in these approaches. This approach uses Euclidean distance which gives poor results on text dataset. The results are only evaluated with the weighted KNN algorithm.

Structured document representation improves the classification correctness, as scientific publications are well prepared documents; therefore, it is necessary to classify them in taxonomy with high percentage of accuracy. Existing approaches lack in use of relevant information of both metadata and text available in the document. Some approaches towards classification rely on either keywords, or abstract or full text of the document. Only two approaches [30-31] focus on the classification of scientific publications. Due to the above mentioned problem of accuracy we have devised an approach that uses relevant information with improved classification accuracy, using the transformation approach.

## 3 A Proposed Framework for Multi-Label Document Classification

Document classification procedure is described in the proposed framework (Figure 1). At first, the dataset is initialized with the labelled instances. Metadata is extricated from each of the training instances. Taxonomy in tree form is also given as an input to the classification process. The multi-labelled dataset is then converted to a single-label dataset by using the four conversion techniques. In Figure 1, the user issues a test document (TD) for category identification. Features are extracted from the user input document in the extractor module. The selected input is pre-processed using stop words removal and stemmer algorithm [32] in the pre-processing module of the framework. From the extracted data, unnecessary words are removed by the *stop word removal* module. The pruned set of words after removal is given to the *stemmer* module. The input features of test document are then represented as vectors.

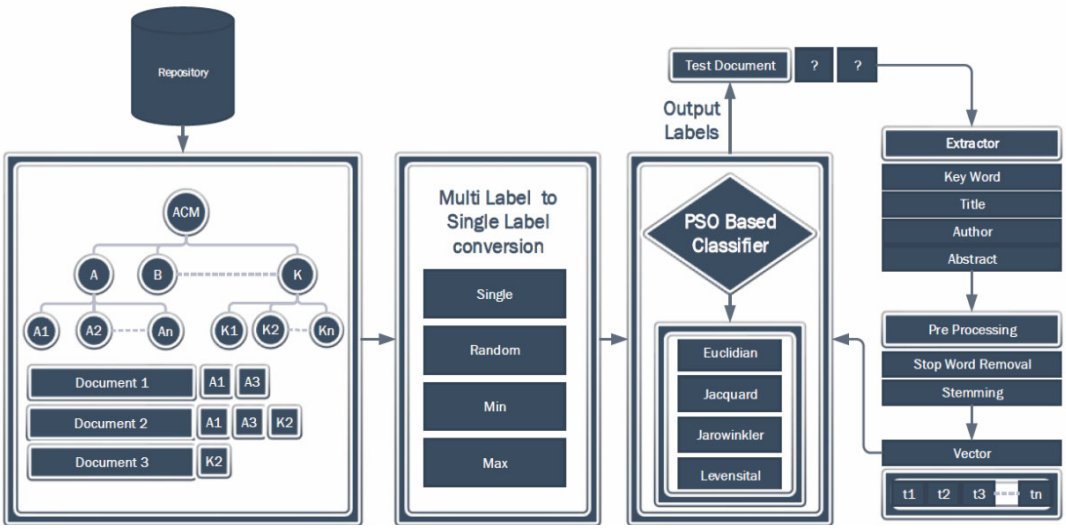


**Figure 1.** Proposed framework for classification of test document

The classification scheme is the central model of our framework; different similarity measures for our classifier have been evaluated. The category of the TD using the training dataset is predicted by the classifier. The resultant category of TD is returned to the user and the relevant category/ies information is updated in the dataset. The well know evolutionary particle swarm optimization inspires the proposed solution towards automated category identification. Detail discussion about major components is given in following subsections.

### 3.1 Multi-label to Single-label Conversion

There are two ways to deal with MLC problem: problem transformation and algorithm adaptation. Problem transformation method converts a MLC into single-label classification problem. On the other hand, the algorithm adaptation method is all about changing

the algorithm for MLC problem. Our focus is on problem transformation method.

Binary relevance (BR) train one binary classifier for every label [33-35]. For each class it associates true or false with every instance, as represented in Figure 2. In our problem domain there are a large number of instances. This technique expands the dataset excessively, so we cast off this method because of its complexity which is O (n. m), where n is the number of instances and m is the total number of labels. We have thirteen labels in J.UCS dataset and 11 labels in the ACM dataset [11]. By adopting this approach, 13 and 11 instances will replace each record in J.UCS and ACM dataset respectively. As a result, the 1,460 instances in J.UCS dataset increases to 18980 instances and 6,116 instances to 94,7276 instances in the ACM dataset. To predict the output, m binary classifiers are being used after transformation using BR.

| PID | Title | Keyword | Class |
|---|---|---|---|
| 1 | Integration Communities Process Oriented Structures | Cooperative knowledge generation knowledge | H |
| 3 | Small Groups Learning Synchronously Online Workplace | Professional training, workplace learning computer | H |
| 4 | Using Weblogs Knowledge Sharing Learning Information | Experience based Information System wiki, weblog | H, J, K |
| 5 | Modelling Implementing Pre built Information Spaces | Modelling method introduction method context | H, J |

| PID | H |
|---|---|
| 1 | True |
| 3 | True |
| 4 | True |
| 5 | True |

| PID | J |
|---|---|
| 1 | False |
| 3 | False |
| 4 | True |
| 5 | True |

| PID | K |
|---|---|
| 1 | False |
| 3 | False |
| 4 | True |
| 5 | False |

**Figure 2.** Binary relevance transformation of multi-label for single-label

Single-label learning is one of the simplest techniques that overlooks all the multi-label instances from the dataset [33]. On the contrary, the information on multi-labels is lost. We have applied this technique to our dataset as reducing the dataset size do not affect the classifier accuracy.

Another approach is to adapt the multi-label instances with most frequent label (max), or less frequent label (min) or a random selection (Rand). In these techniques the information loss is not as big in terms of multi-labels for each instance as shown in Figure 3. The max label selection yields better results as compared to less frequent label selection.

| PID | Title | Keyword | Class |
|---|---|---|---|
| 1 | Integration Communities Process Oriented Structures | Cooperative knowledge generation knowledge community | H |
| 3 | Small Groups Learning Synchronously Online Workplace | Professional training, workplace learning computers | H |
| 4 | Using Weblogs Knowledge Sharing Learning Information | Experience based Information System wiki, weblog | H, J, K |
| 5 | Modeling Implementing Pre built Information Spaces | Modeling method introduction, method context awareness | H, J |

| Max | |
|---|---|
| PID | Class |
| 1 | H |
| 3 | H |
| 4 | H |
| 5 | H |

| Min | |
|---|---|
| PID | Class |
| 1 | H |
| 3 | H |
| 4 | K |
| 5 | J |

| Ran | |
|---|---|
| PID | Class |
| 1 | H |
| 3 | H |
| 4 | J |
| 5 | J |

| Single | |
|---|---|
| PID | Class |
| 1 | H |
| 3 | H |

**Figure 3.** Multi-label transformation to single-label using Max, Min, and Rand and Single-label selection

There are also many other methods like weighting technique. It assigns weight to each label of an instance. As the multi Naïve Bayes classier does not use weights, so this technique was ignored. Pairwise comparison is also considered one of the important methods. It learns one single-label classifier H: X→P (L), where P is Powerset of L [36]. In this procedure different combination of class labels is considered. In our

domain as there are a large number of labels due to which this procedure is not applicable. As a result of this method, each category contains very few records which affect the classifier results too much. In our experiments we have considered the techniques in which information loss is low and which can be applied to our scientific document datasets.

### 3.2 Proposed Algorithm Using Transformed Single-label Dataset

A large number of documents are added to the web and their ratio is increasing with each passing day. These documents include a large number of features. It is one of the important reasons due to which accurate classification is becoming very difficult to discover. Self-adaptability of evolutionary approaches makes it possible to use it for such a dynamic problem having many features of a large number of documents.

A document can be assumed as a particle regarding to its own position and the position of other documents in the taxonomy. Thus, based on their arrangements, one can find the correspondence between any two documents. The proposed solution is stimulated from the well known PSO algorithm which provides local and global optimum with high convergence rate [37]. Documents in the taxonomy are often characterized by their local positions in the categories, along with the global positions, with fellow category documents.

The new Test Document (TD) classification depends upon the category wise measurement of resembling features of all particles. The category selection at the first level of the taxonomy determines the new document association on the second and third levels. The document's similarity with each category stimulate the movement of TD. Scientific publications can be classified into multi-level taxonomy, which becomes more complex with the total number of documents and nature of classification. These two points inspire PSO as optimum solution. PSO is preferred due to its simple and easy implementation and computationally proficient in nature [6, 37]. In our work, we modify PSO by incorporating the following features in order to improve its accuracy for multi-level hierarchical categorization. During pbest calculation, different similarity measures are examined in order to produce better results.

In our proposed solution, we use a recursive technique for new document classification as shown in Figure 4. For matching TD, the total number of documents is selected randomly from each category. The procedure is that the pBest of all available categories are used for selecting multiple gBest. The new document is then matched with the subcategories among the selected ones. This process is continued until and unless the bottom (leaf) level of ACM CSS is reached.

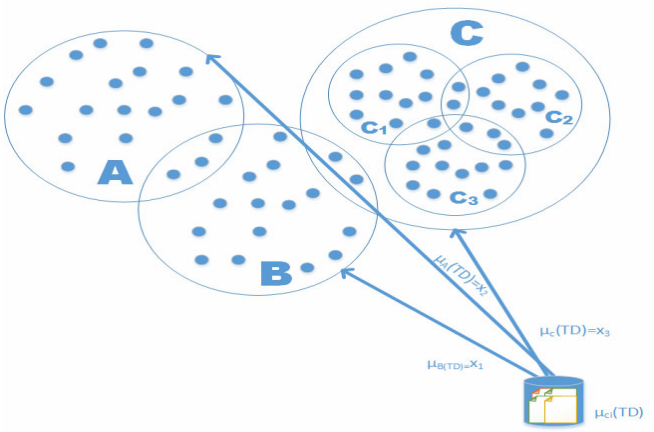


**Figure 4.** Classification of TD

Formally, the solution can be formulated as:

$$C = \sum_{i=A}^{K} C_i \tag{3}$$

$$C_i = \sum_{j=0}^{m} C_{ij} \tag{4}$$

Where C is the set of all categories, each category contains sub-categories as shown in Equation 3 and Equation 4. Each sub-category may itself contain third level sub-categories. Each category has a set of documents modelled as

$$C_{ij} = \sum_{k=1}^{n} D_k \tag{6}$$

Each document contains a set of features $\{X_1, X_2, \ldots X_n\} \in D_k$ and set of labels $\{C_1, C_2, \ldots, C_q\} \in C$, as a feature vector modelled as.

$$D_k = \sum_{p=1}^{r} X_p \wedge \sum_{q=1}^{s} C_q \tag{7}$$

Similarly, the test document TD contain feature only for which the label set is to be predicted, as modelled in equation 8.

$$TD = \sum_{p=1}^{r} X_p \tag{8}$$

The taxonomy consists of a set of classess defined over a partial order set $(C, \prec)$, where C is finite set of nodes in the taxonomy, that are defined for a particular domain, $\prec$ represents the IS-A relationship between parent and child class. IS-A relationship is asymmetric, anti-reflexive and transitive. In the taxonomy, R represents the root node. Other properties of the taxonomy are given as

- $\forall$ ci, cj $\in$ C, if ci $\prec$ cj then cj $\prec$ ci.
- $\forall$ ci $\in$ C, ci $\prec$ ci.
- $\forall$ ci, cj, ck $\in$ C, ci $\prec$ cj and c j $\prec$ ck imply ci $\prec$ ck.

In the proposed algorithm (Figure 5) features from different segments of the research papers are retrieved for classification. The dataset is divided into training and testing set. Each document from the test set is checked in order to assign a label. Training set T is formulated by randomly selected documents from each category. The test document TD is matched with each document and the similarity score is stored in a list $A_k$. Different similarity measures are used to find the score between the test document and every training document. The similarity measure of test document with each train document in the list $A_l$ is then sorted. From the sorted list top k similarities score is chosen, from which most frequent label can be assigned as label of the documents.

```
Input: Test Set TS
       Train Set TR
       Taxonomy C
Output:  predicated set of label for TS
    Initialization:
        For each category Ci in C
            Initialize train set TR with random number of documents Di from each category Ci
        Select similarity measure
    Classification:
        do
            for j=1; j<=sizeof(TR); j++
                A[j] = sim_measure(TS, TR[j])      //similarity with each train document
            for m=1; m<=sizeof(A); m++             //sort by descending order of similarity measures
                for n=1; n<=sizeof(A); n++
                    if(A[m] < A[n])
                        swap (A[m] ↔ A[n])
            for p=1;  p<=k; p++                    //pick top k similar documents
                C(labelof(A[p]))+=1;               //count label for each selected document
                Labelof(TS) ← max(Ci| i=1 to |Ci|) //frequent label is predicted class
            While(Ci≺Cj)
```

**Figure 5.** Proposed hierarchical PSO based algorithm

The proposed algorithm will examine the complete taxonomy from root to a leaf node, by selecting a label at each level. In our approach, same algorithm will be used at each level of the taxonomy to further classify a document in the deeper levels of the taxonomy. Document similarity plays an important role in finding the closely related documents in the proposed algorithm. Conventional measures merely consider the overlap between words in the documents. Moreover, they show negligence to the semantic relationship and between the contents of a document. We espoused a similarity measure in order to look for the content's relationship of a research document.

Fundamentally, text classification algorithms are reliant on the similarity measure in order to find proximity among documents. Our proposed algorithm, for example finds the K nearest neighbors of the test document. Neighbors are then computed, based on text similarity measures. Mis-classification can occur if true related documents are not selected as the neighbor.

On the other hand, the naïve Bayes algorithm computes the prior and conditional probabilities on the basis of a condition in which a class is assigned to a document. The naïve Bayes algorithm does not regard the semantic relationship between words and exact word matching. It yields low to low classification accuracy as evident from experimental results.

Euclidean distance is mostly used for the geometric shape similarities in image processing. Let $X_i$ and $Y_j$ be two documents represented as term vectors. The Euclidean measure is defined as the root of the sum of all square differences between the respective features of $X_i$ and $Y_j$ as modeled in Equation 9.

$$D_{EUC}(X,Y)=\sqrt{\sum_{j=1}^{J}(X_j-Y_j)^2} \tag{9}$$

Jaccard coefficient is figured as the ratio of common terms between the two documents by the terms available in any of the document but not in both the documents. The similarity of the two documents is 1 if both have the same terms, and zero if they are completely different given in equation 10.

$$D_J(A,B)=1-J(A,B)=\frac{|A\cup B|-|A\cap B|}{|A\cup B|} \tag{10}$$

The binary Jaccard coefficient measures the ratio of common features of two documents with the total number of features in both documents. Usually, it is used in market basket analysis applications.

$$d_j=\begin{cases}0 & if\ m=0\\ \frac{1}{3}\left(\frac{m}{|s_1|}+\frac{m}{|s_2|}+\frac{m-1}{m}\right) & otherwise\end{cases} \tag{11}$$

Levenshtein Distance is often used for similarity checks, given its simple nature in equation 12.

$$lev_{(i,j)} = \begin{cases} \max(i,j) \\ \min \begin{cases} lev_{a,b}(i-1,j)+1 & if \min(i,j)=0 \\ lev_{a,b}(i,j-1)+1 & otherwise \\ lev_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} \end{cases} \quad \textbf{(12)}$$

## 4 Results and Discussion

The proposed algorithm is implemented in Java (Netbeans IDE 8.0) with 2.30 GHz processor and 8 GB RAM. The workbench of MySQL was used for the datasets. The employed algorithm besides the four conversion techniques is tested on two real datasets. Our first dataset is the publications in J. UCS [10] and the second dataset is an ACM [11]. The aspects of both the dataset are given in below Table 2. Accuracy over the different datasets is computed as in Equation 13.

**Table 2.** Features of J. UCS and ACM dataset

| Features Dataset | Number of papers | Number of classes | Total number of labels | Label cardinality |
|---|---|---|---|---|
| J. UCS | 1460 | 13 | 3044 | 2.08 |
| ACM | 86116 | 11 | 137679 | 1.60 |

$$Accuracy = \frac{\sum_{i=A}^{K} (C_i = C_i')}{n} \quad (13)$$

A to K are categories, n is the total number of test documents, $C_i$ and $C_i'$ are the sets of actual and predicted categories.

We implemented the above four similarity measures along with the proposed algorithm. The results were obtained on J. UCS dataset using the four transformation techniques from multi-label to a single-label. The dataset contains 1400 instances in which data with missing labels, title or keywords were sorted out. The remaining 1017 instances with multi-labels were transformed into Max, Min, Rand and Single dataset. The proposed algorithm was then used on these four datasets whose result is shown in Table 3. The Jaccard similarity measure outperformed as compared to the other three techniques. The Max dataset encloses 764 training and 253 instances, the accuracy of Jaccard, Levenshtein, Jaro Winkler and Euclidean was 77.86, 73.51, 47.82 and 49.8 respectively. The correspondence of Euclidean among these measures is very low for text data. On Min dataset the accuracy of these measures is lightly low as compared to Max dataset. However, Euclidean accuracy is improved from 49.8 to 53.56 percent.

**Table 3.** Result of different similarity measure of proposed algorithm on transforming datasets

| Similarity | | | Jaccard | | Levenshtein | | Jaro Winkler | | Euclidean | |
|---|---|---|---|---|---|---|---|---|---|---|
| Data Set | Train Set | Test Set | Correct % | Incorrect % | Correct % | Incorrect % | Correct % | Incorrect % | Correct % | Incorrect % |
| Max | 764 | 253 | 77.865 | 22.134 | 73.51 | 26.48 | 47.82 | 52.17 | 49.8 | 50.19 |
| Min | 764 | 253 | 77.47 | 22.529 | 71.54 | 28.45 | 46.24 | 53.73 | 53.96 | 47.04 |
| Ran | 764 | 208 | 83.173 | 16.826 | 73.07 | 26.92 | 44.71 | 55.28 | 53.84 | 46.15 |
| Single | 551 | 170 | 79.411 | 20.588 | 68.23 | 31.76 | 42.35 | 57.64 | 40 | 60 |

In Rand dataset, a single-label is assigned from multi-label available for an instance. Likewise, the Rand dataset includes 764 training and 208 instances. The accuracy of Jaccard, Levenshtein, Jaro Winkler and Euclidean was 83.17, 73.07, 44.71 and 53.84 respectively. The precision of Jaccard is highest i.e., 83.17 on Rand dataset among all the sets. Moreover, the association of Jaro Winkler and Euclidean is also better as compared to the preceding run on other datasets. On the other hand, the accuracy with Jaro Winkler is decreased as well. The Min dataset depicts that the accuracy of Jaccard is higher than the other similar measures for the second time. The accuracy of Jaccard, Levenshtein, Jaro Winkler and Euclidean on this dataset is 79.41, 68.23, 42.35 and 40 percent correspondingly. Among all the runs, the accuracy of Jaccard is high. On the average, the accuracy of the proposed algorithm with Jaccard measure is 79.47.

We compared the accuracy of the proposed algorithm with a multinomial naïve Bayes [38], ZeroR [39], SMO [40], Kstar [41] and J48 algorithms for the transformed dataset in Weka. The accuracy of proposed algorithm using Jaccard similarity measure is high as compared to all the algorithms. Detail accuracy comparison of the algorithms is given in Table 4 on text data, the multi-nomial naïve bayes and the proposed algorithm give better accuracy.

On Max dataset the Multinomial naïve Bayes algorithm gave 62.0619 percent accuracy while the accuracy of proposed algorithm is 78.79 percent. Nevertheless, a single dataset having 449 train instances and 141 test instances shows the accuracy of proposed algorithm that was higher than the other algorithms. From the experimentation, an interesting patten can be observed that most of the classifiers have better accuracy on max and a single dataset. The reason in max dataset, most frequent label is assigned, which increases the overall accuracy. In single-label

**Table 4.** Accuracy percentage comparison on J.UCS dataset

| Dataset | #Train | #Test | Multinomial Naïve Bayes | ZeroR | SMO | Kstar | J48 | HDC Proposed |
|---|---|---|---|---|---|---|---|---|
| Max | 864 | 283 | 63.06 | 34.84 | 59.93 | 39.72 | 46.34 | 78.79 |
| Single | 449 | 141 | 68.02 | 20.40 | 61.90 | 30.61 | 53.06 | 75.88 |
| Ran | 864 | 283 | 55.74 | 23.24 | 54.70 | 33.79 | 40.76 | 71.73 |
| Min | 864 | 283 | 42.16 | 16.02 | 37.63 | 26.13 | 28.57 | 60.77 |

dataset, instances having single label are considered only which reduces the dataset. On small dataset the classifier produces better results.

The average accuracy of multinomial naïve Bayes and HDC proposed algorithm on all dataset is 57.24% and 71.79%. Overall, 15 percent accuracy improvement is achieved on these datasets. In this run, the accuracy of all algorithms was reported low due to the sparseness in the dataset. We repeated the experiment by removing the categories with fewer instances.

The experiment was reported on instances from five categories (H, I, D, F and K) for J. UCS dataset as reported in Table 5. In this run, the accuracy of all the algorithms increased and the accuracy of proposed algorithm is still better than all other algorithms. The overall accuracy of each classifier improved as regards to the results shown in Table 4. The average accuracy of the multinomial dataset on Max, Single, Rand and Min dataset is 71.38. The average accuracy of HDC proposed is 79.48. The accuracy of proposed algorithm is 8% higher than the multinomial naïve Bayes algorithm. The phenomenon of removing the categories with less number of document improved the average accuracy of multinomial naïve Bayes from 57.24% to 71.39% with an improvement of 14 percent. In the same way, the average accuracy of the proposed algorithm improved from 71.79% to 79.48%. From the accuracy difference it can be inferred that the HDC proposed solution is less affected by the sparseness in the datasets. The best accuracy of each classifier remained the same as highlighted, as bold for each classifier. On this run, the best accuracy of each classifier was on max and a single dataset.

**Table 5.** Accuracy percentage comparison on 5 categories of J. UCS dataset

| Dataset | # Train | # Test | Multinomial Naïve Bayes | ZeroR | SMO | Kstar | J 48 | HDC Proposed |
|---|---|---|---|---|---|---|---|---|
| Max | 764 | 253 | 70.47 | 41.73 | 67.32 | 44.88 | 59.44 | 77.86 |
| Single | 551 | 170 | 78.51 | 33.88 | 76.03 | 47.10 | 59.50 | 79.41 |
| Ran | 764 | 253 | 71.25 | 31.88 | 66.53 | 45.27 | 58.66 | 83.17 |
| Min | 764 | 253 | 65.32 | 30.80 | 65.23 | 45.27 | 58.66 | 77.47 |

We repeated the experiment on a bigger dataset of ACM documents [11]. Papers from three categories (H, D and I) were deemed for categorization and taxonomy. The dataset was transformed into Max, Min, Rand and Single dataset. The result is reported in Table 6. Max set contains train 28533 number of documents and the test set contains 9457 documents. The accuracy of proposed algorithm is low as compared to multinomial naive Bayes. The result was only compared with multinomial naïve bayes as the Weka tool does not support large dataset. The rationale for this is a random selection of train documents. The proposed algorithm for the larger dataset is efficient as compared to the multi nominal naive Bayes.

**Table 6.** Result comparison on ACM dataset

| Dataset | # Train | # Test | Multinomial Naïve Bayes | HDC Proposed |
|---|---|---|---|---|
| Max | 28533 | 9457 | 64.31 | 65.84 |
| Min | 28533 | 9457 | 60.62 | 60.78 |
| Ran | 28533 | 9457 | 60.02 | 60.18 |
| Single | 22312 | 7382 | 64.70 | 64.92 |

## 5 Conclusion

Label identification for scientific document is an important research area. At present, the authoring of scientific documents using labels to their document is used manually as well.

The systematic documents can be fit into numerous groups. Presently, we have formulated a solution by making over multi-label dataset to single-label dataset. The proposed solution gives overall 15 percent better accuracy than a multinomial naïve Bayes algorithm. The predicament in our algorithm is scalability, which can be inspected, and be balanced and evaluated in terms of competence with the current state of the art algorithms. We have also evaluated different similarity measure for text data. According to the experimental results, Jaccard text similarity measure gives better accuracy as compared to other measure. The maximum label selection for transforming multi-label set to single-label set also gave improved results as compared to other adaptation procedures. In future the algorithms can be empirically validated for time complexity.

## References


[1] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, *Advances in Knowledge Discovery and Data Mining*, American Association for Artificial Intelligence, 1996.

[2] D. Koller, M. Sahami, Hierarchically Classifying Documents Using Very Few Words, *In Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97)*, Nashville, TN, 1997, pp. 170-178.

[3] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, ACM Press, 1999.

[4] J. Han, M. Kamber, J. Pei, *Data Mining: Concepts and Techniques*, Elsevier, 2011.

[5] G. Peter, H. Matthias, K. birgit, Text Mining: Grundlagen, Verfahren und Anwendungen, *HMD-Praxis der Wirtschaftsinformatik*, Vol. 38, No. 222, pp. 38-48, December, 2001.

[6] Z. Wang, Q. Zhang, D. Zhang, A Pso-based Web Document Classification Algorithm, *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, Qingdao, China, 2007, pp. 659-664.

[7] C. Cortes, V. Vapnik, Support-vector Networks, *Machine Learning*, Vol. 20, No. 3, pp. 273-297, September, 1995.

[8] G. Salton, Developments in Automatic Text Retrieval, *Science*, Vol. 253, No. 5023, pp. 974-980, August, 1991.

[9] N. Coulter, J. French, E. Glinert, T. Horton, N. Mead, R. Rada, A. Ralston, C. Rodkin, B. Rous, A. Tucker, P. Wegner, E. Weiss, C. Wierzbicki, Computing Classification System 1998: Current Status and Future Maintenance, Report of the CCS Update Committee, *Computing Reviews*, Vol. 39, No. 1, pp. 1-5, January, 1998.

[10] M. T. Afzal, W.-T. Balke, H. Maurer, N. Kulathuramaiyer, Improving Citation Mining, *First International Conference on Networked Digital Technologies*, Ostrava, Czech Republic, 2009, pp. 116-121.

[11] A. P. Santos, F. Rodrigues, Multi-label Hierarchical Text Classification using the ACM Taxonomy, *14th Portuguese Conference on Artificial Intelligence*, Aveiro, Portugal, 2009, pp. 553-564.

[12] G. Tsoumakas, I. Katakis, Multi-label Classification: An Overview, *International Journal of Data Warehousing and Mining*, Vol. 3, No. 3, pp. 1-13, July, 2007.

[13] J. H. Zaragoza, L. E. Sucar, E. F. Morales, C. Bielza, P. Larranaga, Bayesian Chain Classifiers for Multidimensional Classification, *The Twenty-Second International Joint Conference on Artificial Intelligence-Volume Three*, Catalonia, Spain, 2011, pp. 2192-2197.

[14] J. Read, L. Martino, D. Luengo, Efficient Monte Carlo Optimization for Multi-label Classifier Chains, *International Conference on Acoustics, Speech and Signal Processing*, Vancouver, Canada, 2013, pp. 3457-3461.

[15] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier Chains for Multi-label Classification, *Machine Learning*, Vol. 85, No. 3, pp. 333-359, December, 2011.

[16] G. Tsoumakas, I. Vlahavas, Random k-labelsets: An Ensemble Method for Multilabel Classification, *18th European Conference on Machine Learning*, Warsaw, Poland, 2007, pp. 406-417.

[17] G. Tsoumakas, I. Katakis, I. Vlahavas, Random k-labelsets for Multilabel Classification, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 23, No. 7, pp. 1079-1089, July, 2011.

[18] G. Tsoumakas, I. Katakis, I. Vlahavas, Mining Multi-label Data. in: O. Maimon, L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook*, Springer US, 2009, pp. 667-685.

[19] J. Read, L. Martino, P. M. Olmos, D. Luengo, Scalable Multi-output Label Prediction: From Classifier Chains to Classifier Trellises, *Pattern Recognition*, Vol.48, No. 6, pp. 2096-2109, June, 2015.

[20] J. Read, P. Reutemann, Bernhard, B. Pfahringer, G. Holmes, A Multi-label/Multi-target Extension to Weka, *Journal of Machine Learning Researc*, Vol. 17, No. 21, pp. 1-5, February, 2016.

[21] B. Klimt, Y. Yang, The Enron Corpus: A New Dataset for Email Classification Research, *European Conference on Machine Learning*, Pisa, Italy, 2004, pp. 217-226.

[22] J. Lehečka, J. Svec, Improving Multi-label Document Classification of Czech News Articles, *International Conference on Text, Speech, and Dialogue*, Pilsen, Czech Republic, 2015, pp. 307-315.

[23] M. Hrala, P. Kral, Multi-label Document Classification in Czech, *International Conference on Text, Speech and Dialogue*, Pilsen, Czech Republic, 2013, pp. 343-351.

[24] T. Ali, S. Asghar, N. A. Sajid, M. Ahmad, Classification of Scientific Publications using Swarm Intelligence, *Pakistan Academy of Sciences*, Vol. 50, No. 2, 115-126, June, 2013.

[25] N. A. Sajid, M. T. Afzal, M. A. Qadir, Multi-label Classification of Computer Science Documents Using Fuzzy Logic, *Journal of the National Science Foundation of Sri Lanka*, Vol. 44, No. 2, pp. 155-165, June, 2016.

[26] N. A. Sajid, T. Ali, M. T. Afzal, M. Ahmad, M. A. Qadir, Exploiting Reference Section to Classify Paper's Topics, *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, San Francisco, CA, 2011, pp. 220-225.

[27] S. Vogrinčič, Z. Bosnic, Ontology-based Multi-label Classification of Economic Articles, *Computer Science and Information Systems*, Vol. 8, No. 1, pp. 101-119, January, 2011.

[28] Q. Wu, H. Liu, X. Yan, Multi-label Classification Algorithm Research Based on Swarm Intelligence, *Cluster Computing*, Vol. 19, No. 4, pp. 2075-2085, December, 2016.

[29] Q. Liang, Z. Wang, Y. Fan, C. Liu, X. Yan, C. Hu, H. Yao, Multi-label Classification Based on Particle Swarm Algorithm, *IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks*, Dalian, China, 2013, pp. 421-424.

[30] K. Khor, C. Ting, A Bayesian Approach to Classify Conference Papers, *5th Mexican International Conference on Artificial Intelligence*, Apizaco, Mexico, 2006, pp. 1027-1036.

[31] B. Zhang, M. Andre, Goncalves, W. Fan, Y. Chen, E. A. Fox,

P. Calado, M. Cristo, Combining Structural and Citation-Based Evidence for Text Classification, *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, Washington, DC, 2004, pp. 162-163.

[32] M. F. Porter, An Algorithm for Suffix Stripping, *Program*, Vol. 14, No. 3, pp. 130-137, July, 1980.

[33] M. R. Boutell, J. Luo, X. Shen, C. M. Brown, Learning Multi-label Scene Classification, *Pattern Recognition*, Vol. 37, No. 9, pp. 1757-1771, September, 2004.

[34] B. Lauser, A. Hotho, Automatic Multi-label Subject Indexing in a Multilingual Environment, *7th European International Conference on Theory and Practice of Digital Libraries*, Trondheim, Norway, 2003, pp. 140-151.

[35] T. Li, M. Ogihara, Detecting Emotion in Music, *Proceedings of the International Symposium on Music Information Retrieval*, Washington, DC, 2003, pp. 239-240.

[36] S. Diplaris, G. Tsoumakas, P. A. Mitkas, I. Vlahavas, Protein Classification with Multiple Algorithms, *10th Panhellenic Conference on Informatics*, Volas, Greece, 2005, pp. 448-456.

[37] R. Eberhart, J. Kennedy, A New Optimizer Using Particle Swarm Theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science MHS'95*, Nagoya, Japan, 1995, pp. 39-43.

[38] A. M. Kibriya, E. Frank, B. Pfahringer, G. Holmes, Multinomial Naive Bayes for Text Categorization Revisited, *Australian Joint Conference on Artificial Intelligence*, Cairns, Australia, 2004, pp. 488-499.

[39] P. Domingos, M. Pazzani, On the Optimality of the Simple Bayesian Classifier under Zero-one Loss, *Machine Learning*, Vol. 29, No. 2-3, pp. 103-130, November, 1997.

[40] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, K. R. K. Murthy, Improvements to Platt's SMO Algorithm for SVM Classifier Design, *Neural Computation*, Vol. 13, No. 3, pp. 637-649, March, 2001.

[41] D. W. Aha, D. Kibler, M. K. Albert, Instance-based Learning Algorithms, *Machine Learning*, Vol. 6, No. 1, pp. 37-66, January, 1991.

## Biographies

**Tariq Ali** received his MS degree from the Department of Computer Sciences, Muhammad Ali Jinnah University, Islamabad, Pakistan in 2009. Currently, Mr. Ali is working as a lecturer in UIIT, PMAS Arid agriculture university and a Ph.D. (CS) scholar at Abasyn University, Islamabad, Pakistan.

**Sohail Asghar** is working as a *Professor of Computer Science* at COMSATS Institute of Information Technology Islamabad. In 1994, he graduated with honors in Computer Science from the University of Wales, United Kingdom. He received his Ph.D. from Faculty of Information Technology at Monash University, Melbourne Australia in 2006. Dr. Sohail has taught and researched in Data Mining and is a member of ACS, and IEEE. http://ww3.comsats.edu.pk/faculty/FacultyDetails.aspx?Uid=4564