Proceedings of the Pakistan Academy of Sciences 50 (2): 115–126 (2013)
Copyright © Pakistan Academy of Sciences
ISSN: 0377 - 2969 (print), 2306 - 1448 (online)

Pakistan Academy of Sciences

Research Article

Classification of Scientific Publications using Swarm Intelligence

Tariq Ali *, Sohail Asghar, Naseer Ahmed Sajid and Munir Ahmad

Department of Computer Science,
Mohammad Ali Jinnah University, Islamabad, Pakistan

Abstract: Document classification is an important task in data mining. Currently, identifying category (i.e., topic) of a scientific publication is a manual task. The Association for Computing Machinery Computing Classification System (ACM CCS) is most wildly used multi-level taxonomy for scientific document classification. Correct classification becomes difficult with an increase in number of levels as well as in number of categories. Domain overlapping aggravates this problem as a publication may belong to multiple domains. Thus manual classification to taxonomy becomes more difficult. Most of the existing text classification schemes are based on the Term Frequency and Inverse Document Frequency (TF-IDF) technique. Similar approaches become computationally inefficient for large datasets. Most of the techniques for text classification are not experimentally validated on scientific publication datasets. Also, multi-level and multi-class classification is missing in most of the existing schemes for document classification. The proposed approach is based on metadata (i.e., structural representation), in which only the title and keywords are considered. We reduced the features set by dropping some of the metadata, like abstract section of the scientific publication that diversifies the result accuracy. The proposed solution was inspired from the well-known evolutionary Particle Swarm Optimization (PSO). The proposed technique results in overall 84.71% accuracy on Journal of Universal Computer Science (J.UCS) dataset.

Keywords: Topic identification, category identification, document classification, multi-class, multi-level

1. INTRODUCTION

Classification is an important task in data mining [1]. Automated text categorization is becoming more important with the advent of digital libraries and with a rapid increase in the number of documents on the web. The research community is producing a large number of scientific documents. These documents are then searchable over the internet using search engines, digital libraries and citation indexes. There is a need to classify this huge amount of documents into a hierarchy or taxonomy [2]. Similarly, the document's relatedness to a node in an existing taxonomy will assist in searching the user-relevant information in an efficient way. Accuracy of information retrieval basically depends on accurate classification of the documents [3]. Besides information retrieval, accurate classification helps in analysing trends, finding expertise, and the relevant document recommender system.

Classification is a two level approach in which first level generates a model from the training set of instances and the second level checks accuracy of the classifier [4]. There are a number of approaches for document classification, such as Decision Tree [5], Naive Bays Classifier [6], Particle Swarm Optimization (PSO) [7], Support Vector Machine (SVM) [8], and Term Frequency and Inverse Document Frequcy (TF-IDF) [9]. We have already reported a detailed survey [10] towards automated text classification in the context of supervised learning.

One important step towards the document classification is the category identification. Currently, authors of scientific publications identify the relevant category or categories (from

Received, December 2012; Accepted, April 2013
*Corresponding author: Tariq Ali: Email: tariqali.1982@gmail.com

onward written as category/ies) to their papers manually. Common categorization used in research community is the Association for Computing Machinery Computing Classification System (ACM CCS) [11].

Manually, category identification for a document is difficult task for new researcher, especially if the work belongs to multiple domains. Due to the diversity in domains and mapping of one domain to another domain, the manual classification task is becoming extremely difficult. This research is an effort to bridge the gap between users towards identifying correct document category, and suggest possible categorization to the author's work automatically.

Accurate categorization can be helpful in relevant information (i.e., document) retrieval. Traditional text classification is usually attained by assigning a document to one class, but in scientific community document can belong to multiple categories. We propose that initially the document may be categorized in the top category, and after matching with the top category it may be further classified to its sub-levels as depecited in Fig. 1. The search space for new document clasification is reduced by considering the sub-levels of the parent category/ies selected at the first level. Similar approach may be adopted for the third level, if it exists for any category selected at first and second level.

Scientific document classification has structural advantage over the unstructured document classification, as structural representation increases accuracy of the classification [10]. Most of the existing text classification schemes are based on the TF-IDF technique [7, 12]. Similar approaches become computationally inefficient for large datasets. Detail survey towards document classification is given by Sebastiani [12]. Similarly, most of the techniques for text classification are not experimentally validated on scientific publication datasets. Also, multi-level and multi-class classification is missing in most of the existing schemes for document classification [7, 13-18].

The proposed technique is based on metadata (i.e., structural representation), in which we considered the *title* and the *keywords*. The *title* of a scientific publication normally reflect theme of the work and and the *keywords* are the representative features of the paper to their category/ies.

Every year large number of documents are added to the web. These documents contain a large number of attributes, on the basis of which accurate classification is becoming extremly difficult. Self-adaptability of evolutionary approaches makes it possible to use it for such a dynamic problem having many features for huge number of documents. A document can act as a particle with regard to its own position and the position of other documents in the taxanomy. Thus, based on their position, we can find the similarity between any two documents.

The proposed solution is inspired from the well known PSO algorithm [19, 20]. Documents in the taxonomy are represented with its local position in a category along with global position with neighbourhood categories documents. Classifying a new Test Document (TD) depends with the similarity measure of all particles in each individual category. At second and third level, the movement of new document in the taxonomy depends on the selection of category/ies at the first level. The movement of TD is inspired with the document's similarity in each individual category. Classification of a scientific publication to a taxonomy is a multi-level classification. The number of documents and the nature of classification (e.g., multi-class and multi-level) makes this problem more complex. Due to these two reasons, PSO stands out as one of the optimum solution. PSO is simple, easy to implement and computationally efficient.

We have implemented our proposed solution and tested it on the Journal of Universal Computer Science (J.UCS) dataset [21], in which *2/3* instances were used for the training set and *1/3* for the test dataset. Furthermore, we assigned some heuristics for the selection of system generated category. We implemented our proposed techniques and concluded overall 84.71% accuracy on the J.UCS dataset.

Rest of the paper is organized in a way that Section 2 presents the problem statement and Section 3 contains related approaches with critical analysis, Section 4 presents the representation of scientific documents, Section 5 contains the proposed technique for the scientific publication classification, Section 6 contains the experimental results of the proposed approach with detailed discussion and analysis and Section 7 concludes the paper and provides future directions.

2. PROBLEM STATEMENT

Document classification assigns a new document (e.g.., a research paper) to a set of previously defined taxonomy. The pre-defined categories can be of any type. Normally, in computer science the most common categorization is the ACM Computing Classification System (ACM CCS 1998) on which we have tested our proposed approach. Formally, the problem can be defined as given in Eq. 1.

$$d_1 : D \in C_{j..k} : C = d_1 : C \;\ldots\ldots Eq\!: 1$$

Where *D* is a set of documents. *C* is a set of predefined categories. The problem is to classify a document *di* of type *D* to category/ies $c_{j..k}$ belonging to category set *C*. The problem is also depicted in Fig. 1. TD is the user provided publication, whereas *A to K* are the main categories of the ACM CCS. Each category is divided into sub-categories which have further sub-categories at third level. Relevant category for new document has to be calculated in the ACM CCS.

3. RELATED WORK

Different approaches for document classification based on Particle Swarm Optimization (PSO) [19, 20], Support Vector Machine (SVM) [8], Bayesian network [6] etc exist in literature. Some of the document classification techniques work on metadata, while others work on the complete text available in the document. Very few Classification techniques are available for the scientific publication category identification as compare to other document classification approaches. Scientific publication contains both metadata and text, which increases its classification accuracy. Some of the approaches towards document classification with reference to the multi-class and multi-level classification are analysed in detail.

Our survey towards the document classification in the context of supervised learning technique is given in [10]. Classification in both structured and unstructured context is analysed. With published literature [13] it is strongly argued that structured documents give more accuracy in classification over unstructured documents. This survey provides theoretical comparison of different techniques, while accuracy and efficiency of classifiers is missing [10].

A PSO based document classification for web documents is given in [7]. In this approach the documents are pre-processed by removing *stopwords* [22] and *wordsstemming* by using porter stemmer [23] algorithm. After preprocessing, the documents were represented as document term frequency matrix. Documents were finally represented as term vectors, using TF-IDF weighting approach [7]. Feature selection was done through entropy weighting scheme [25]. Entropy weighting scheme is done using local weighting of term *k* and global weighting of term *k* as *Ljk* × *Gk*. after feature selection, particle swarm optimization (PSO) was used as a classifier.

Initialization of individual particles is done randomly; the structure for each particle at given iteration is represented [7] as

$$\mathrm{X}_{\mathrm{i}}^{0} = (\, \mathrm{x}_{\mathrm{i1}}^{0}, \ldots, \mathrm{x}_{\mathrm{in}}^{0})$$

Where *0* represents the iteration and n represents the term numbers in document set. The velocity of individual particles [7] is given as, which corresponds to the update quantity of all weighting values.

$$\mathrm{V}_{\mathrm{i}}^{0} = (\, \mathrm{v}_{\mathrm{i1}}^{0}, \ldots, \mathrm{vi}_{\mathrm{in}}^{0})$$

Finally, the effectiveness is measured in terms of precision and recall, as:

F1 = (2 * Precision * Recall) / (Precision + Recall)

Experimental evaluations were performed on two standard text dataset reauter-21578 and TREC-AP. In this approach [7] no weighting mechanisms for structural contents are used. Similarly, classifying a document to multiple categories is missing. Strategies for multileveled classification are also not discussed.

Effectiveness of PSO with respect to different dataset towards classification problem is given in [15]. Ten different datasets with multiple instances, classes and number of parameters composing each instance are taken. PSO accuracy in terms of error rate is compared with other nine classifiers. On three data sets PSO outperformed than all other classifiers. PSO efficiency with increasing number of classes is highlighted, which may be due to implementation or similarity measure used for evaluating fitness function.

Improving the document classification with structural contents and citation based evidence is given in [16]. For classification, both structural (i.e., title and abstract) and citation based information is considered. Different similarity measures for both structural (i.e., bag of words, cosine, and Okapi) and citation based (e.g., Bibliographic coupling, Co-citation, Amsler, and Companion) similarity are used. Genetic programming is used for classification of new document. For prediction of new document, best similarity tree for each class is maintained. Class for the new document, is decided on the based of majority voting from each classifier.

A new approach based on the neighbourhood preserving embedding (NPE) with PSO is given in [17]. NPE preserves the local manifold structure and preserve the most discriminating features for classification. Documents features in the higher domain *X* are reduced to the lower domain *Y* by using the NPE. PSO is used similar to the approach presented in [7] Discriminative features extraction plays an important role in increasing the document classification accuracy. Results of the NPE with PSO has shown better results than LDA PSO, LSI PSO, and LSI-KNN [17].

Bayesian based approach for the classification of conference paper is given in [13]. 400 educational conference papers in four categories (e-learning, Teacher Education, Intelligent Tutoring System, and Cognition issues) are used for constructing the Bayesian network. Only keywords are used for conference paper classification. Compound keywords are parsed into Single keywords which are ranked with respect to frequency and top 7 keywords for each category are considered as input for Bayesian network. Each category shares some common keywords along with some individual keywords. The network has trained with 100 papers for each category. This technique is efficient due to the reason that only keywords are used for classification. Conversely, the misclassification error can increase due to non availibilty of keywords in some documents or due to wrong keywords assigned by the authors.

Text classification using swarm intelligence in terms of automated grouping of PDF documents is given in [18]. The algorithm presented is inspired from the ant colony optimization. Basically classification is used in terms of clustering; PDF documents are converted into text files. Relative frequency of words in a document is calculated which is normalized with the word frequency in all documents to lower the importance of words occurring in all documents. Cosine similarity is used as a measure between the two objects.. For convergence the picking parameter of ant for picking an object was reduced.

Association rule mining approach towards document classification is given in [26]. Associative rule mining discovers relationships among items in a dataset. Documents are represented as transactions. *Stopswords* removal and *Stemming* is used to reduce the transaction size. Initially, rules are generated using the apriori algorithm. Two methodologies are used for the rules generation: one is, rule generation for each category; and the other one is association rule mining for the categories collectively. On the basis of these rules, classifier is developed. Experimental results are presented on the Reuters-21578 text collection [27].

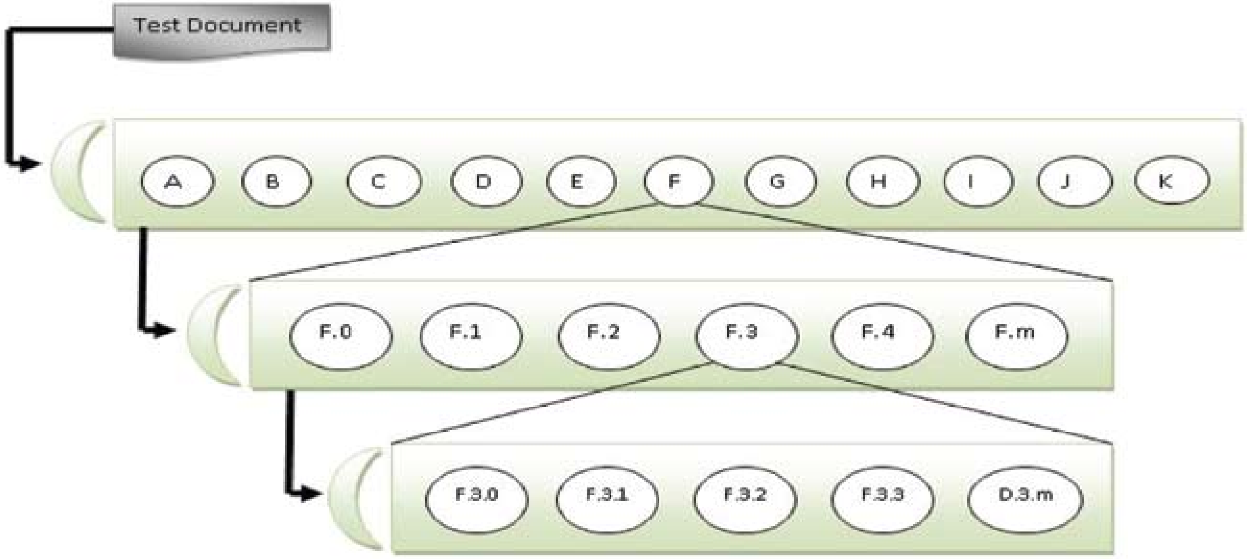
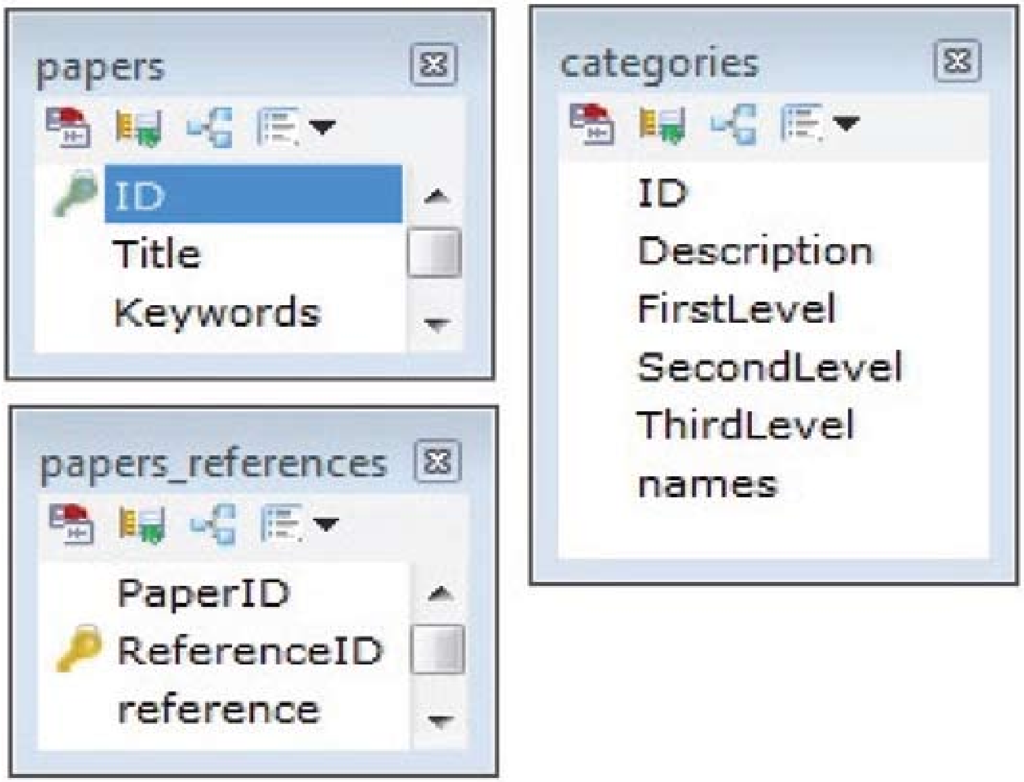


Fig. 1. Classifying new document to ACM CCS.



papers

ID
Title
Keywords

papers_references

PaperID
ReferenceID
reference

categories

ID
Description
FirstLevel
SecondLevel
ThirdLevel
names

Fig. 2. Relevant information of documents.

Linear text classification using category relevance factors (CRF) is given in [14]. CRF is maintained for all documents that belongs to a category. Profile vector for each category is maintained from CRF feature vector. Based on the cosine similarity between papers and categories, the document is classified to the category having maximum similarity.

Structured document representation increases the classification accuracy, as scientific publications are well structured 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 data and text available in the document. Some approaches towards classification rely on either *keywords*, or *abstract* or *fulltext* of the document. Most of the existing schemes are not evaluated for scientific publications datasets. Only two approaches [13, 16] focus on the classification of scientific publications. Similarly, most of the schemes do not consider multi class classification and multi level classification. To overcome these limitations, we have devised an approach that not only uses the relevant information for classification but also deals with multi-class and multi-level classification. Detailed discussion on relevant data selection from scientific publication is presented in the following sections.

4. DOCUMENT REPRESENTATION

For efficient document classification metadata and contents can be used. The common features are *title*, *abstract*, *authors*, *keywords*, and *references*. In most of the cases, *title* contains the theme of the work presented in the paper. *Author's* information helps in identifying previous papers of the authors in the database. *Abstract* summarizes the paper and almost contains the important terms and theme of the paper. *Keywords* are the most weighted terms in assigning a document. This part mostly shows the domain of the paper. *References* can help in finding the cited paper's category. In case of the most papers category of *references*, it is highly **likely** to assign the paper to that category [28]. Among these features, we selected *title* and *keywords*. The selection of these is discussed in detail in the discussion section. The relevant information required for document classification is depicted in the Fig. 2.

The information about a document is first pre-processed. Pre-processing prepares the data for accurate classification. First step in the pre-processing is the removal of unnecessary words from the TD. From this relevant feature (*Title* and *Keywords*) *stopwords* are removed using a list containing 548 *stopwords* [29]. After removing the *stopwords,* these features are stemmed using the well-known porter stemmer algorithm [23]. Based on the pre-processed data, we apply our classification algorithm.

5. PROPOSED TECHNIQUE FOR SCIENTIFIC DOCUMENT CLASSIFICATION

Document classification process is depicted in the proposed framework (Fig. 3). Initially, the dataset is populated with the similar approach.. *Title* and *keywords* are extracted from each of the documents in the training dataset. Extracted *title* and keywords are pre-processed using *stop words removal* and *stemmer* module in the framework. Representation of the dataset with respect to the documents is given in Fig. 2 and Fig. 3.

In Fig. 3, the user issues a TD for category identification. The TD is parsed in the *dataextractor* module, in which relevant data (*Title* and *Keywords*) are retrieved. The data extracted is passed on to the *stopwordremoval* module, which removes the unnecessary words using the list provided in [29]. The remaining text is passed on to the *stemmer* module, which returns the stemmed result to the *matcher* module. For stemming, we used the well-known porter stemmer algorithm [23]. The *matcher* module then predicts the category of the TD using the existing dataset. The category result is returned to the user and dataset is updated with the relevant category/ies information of the TD.

Matcher module is the main module of the proposed framework. The *matcher* classifies an input TD into their relevant category/ies. Proposed solution towards automated category identification is inspired from the well know evolutionary particle swarm optimization. Proposed solution overcomes the two well-known problems (Multi-class and Multi-level) in the solution towards automated category identification.

Initially, the TD is matched with all the documents in each category. Average similarity of each category is computed, among which highly similar (using Eq. 3) categories are selected. At each level, besides identifying the category, we find similarity among the similarity score of each category. The search space is reduced at the second level by assigning the TD to selected categories at first level. Our algorithm recursively reaches to the bottom level to assign the paper to its correct category.

Proposed solution towards the classification of new document is a recursive approach, as depicted in Fig. 4. Initially, the *TD* will be checked with the same number of documents from each category. Global best *gBest* among all the local best *pBest* available categories will be selected. In our case, more than one *gBest* can be selected based on the Eq. 3 among average similarity measures with each category. If the difference is higher than a certain threshold then more than one category can be selected for the new document. After selection at first level, the new document will be matched with all the subcategories of the selected category/ies. The process can be continued till the leaf level of the ACM CCS is achieved or the fitness function is achieved.

Formally, the solution can be formulated as:

$$C = \{\mathrm{A}, \mathrm{B}, \ldots\ldots, \mathrm{K}\}$$

where *C* is the set of categories

Each category contains sub-categories. For example

$$A = \{\mathrm{A}_0, A_1, \ldots\ldots, \mathrm{A}_m\}$$

Each sub-category may itself contain third level sub-categories. Each category contains a set of documents as for example

$$A = \{d_{a1}, d_{a2}, \ldots\ldots, \mathrm{d}_{am}\}$$
$$B = \{d_{b1}, d_{b2}, \ldots\ldots, \mathrm{d}_{bm}\}$$
$$\cdot$$
$$\cdot$$
$$\cdot$$
$$K = \{d_{k1}, d_{k2}, \ldots\ldots, \mathrm{d}_{km}\}$$

Each document contains a set of words among which t_1, t_2,……..t_k are terms belonging to *title* and *keywords*, as a feature vector, as

$$D_{ci} = \{\mathrm{t}_1, t_3, \ldots\ldots, \mathrm{t}_k\}$$

Similarly, the new document is to classify contains terms of *title* and *keywords* as

$$\mathrm{TD} = \{\mathrm{t}_1, t_3, \ldots\ldots, \mathrm{t}_k\}$$

Membership of the TD (similarity) with each category is calculated as

$$\mu_{c_i}(\mathrm{TD}) = \frac{AVG(\sum_{i=1}^{\mathrm{n}} \mathrm{similarity}(TD, \mathrm{d}_{Ci}))}{} \mathrm{n} = x_{ci} \; where \;\; n \; the \; minimum \; number \; of$$

$$documents \; in \; Ci \;\; Eq{:}\,2$$

For multi-topic classification, Eq. 3 is used. In Eq. 3, $max(x_{ci})$ is the maximum average similarity selected at any level in the hierarchy; whereas x_{ci} is the membership of the document in each category using Eq. 2, Ψ is the threshold defined by domain expert which can be the maximum similarity difference between any two categories. We used Levenshtein distance [30] as a similarity measure. Classification at the next level (lower level) will be performed only for the categories selected using equation 3. The document movement in the taxonomy is identified using the Eq 3.

$$D_t \in C_i \text{ where difference } (\max(x_{ci}) - x_{ci}) > \psi \; \ldots. Eq{:}\,3$$

Concept of social interaction is applied in using the PSO. Each particle (category) takes part in classification of the TD category identification in the taxonomy. Position and velocity of TD using PSO is given by the Eq. 2 and Eq. 3. In our velocity equation, we are not using cognitive and

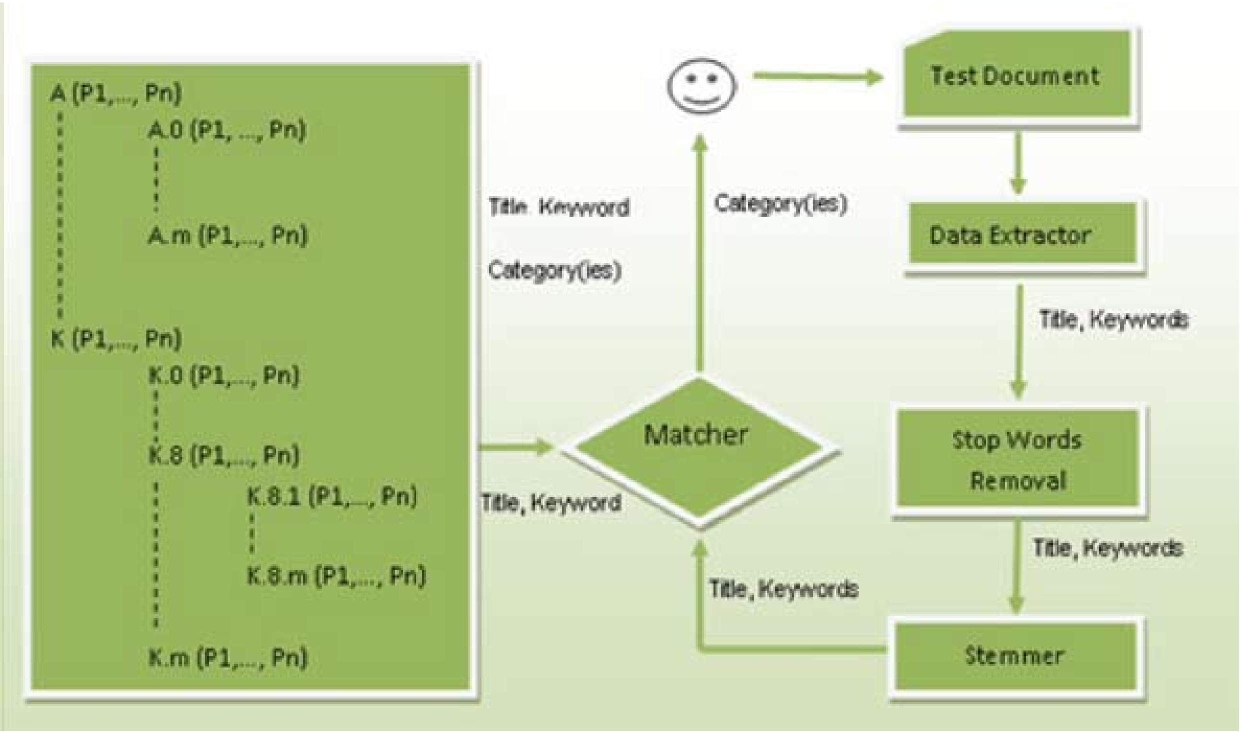


Fig. 3. Framework for classification of test document.

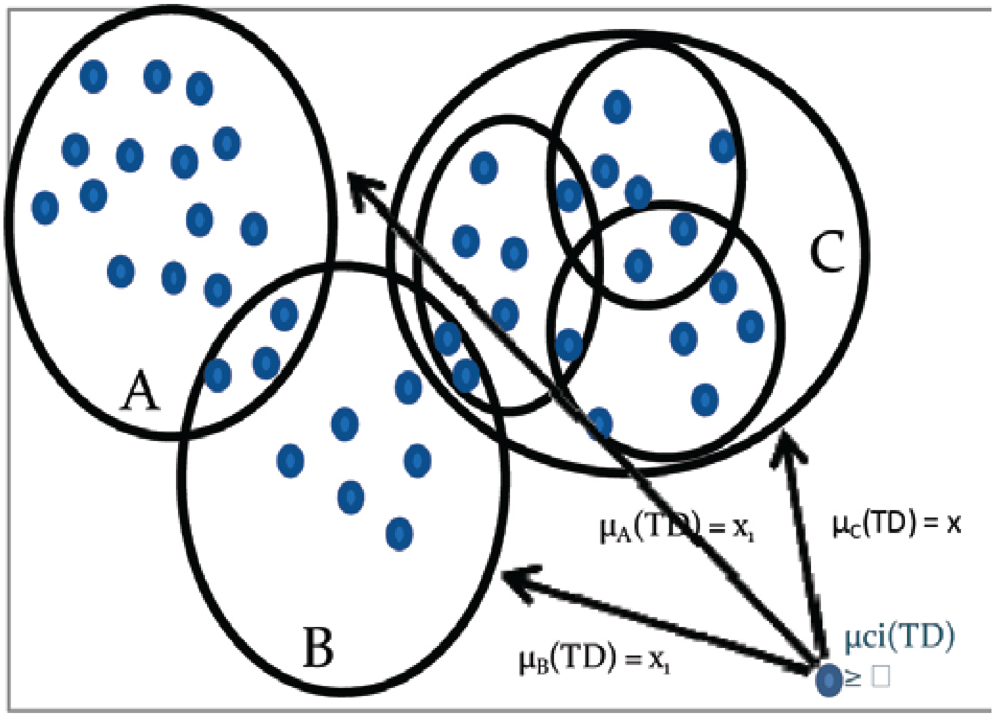


Fig. 4. Classification of test document.

social parameters of PSO, as scientific document classification is hierarchical in nature. And at each level, the document is classified to a category or some of the categories. At lower level, PSO is again applied on the selected categories using Eq. 3.

Fig. 5 explains the proposed algorithm inspired from PSO. This algorithm works as a central module (Matcher) of the framework shown in Fig. 4.Initially the category with minimum numbers of documents is selected from the available categories are selected. *pBest* and *gBest* for each category are calculated using Eq. 2 and Eq. 3 respectively. The resultant category/ies information of the TD is updated in the database.

6. EXPERIMENTAL RESULTS

We have implemented the proposed scheme on the Journal of Universal Computer Science (J.UCS) dataset. Our proposed dataset contains 1460 research publication. We have taken 2/3 documents for training dataset and 1/3 for test dataset. J.UCS has extended the ACM CCS with two more categories with *L* and *M*. Test accuracy result for 30 test documents from each category are shown in Fig. 7. The test sample selected was random in each category as shown in Fig. 7; each column shows the selected input documents in each category. The result of our test documents with each category is given in Fig. 6. The blue bar shows the correctly classified categories whereas the red bar shows the number of incorrect classified documents in each category. Overall accuracy of our approach is more than 84.71%. We have implemented our approach using MySQL as database with PHP.

We performed a set of experiment for the automation of topic identification in ACM CCS. In our first experiment, we used the features provided in the ACM CCS. For top level, we aggregated child features for a parent category. At each level, we stored all of the decedent's features for each category. We used this database for the classification of TD. We matched the extracted TD features with our stored features for each category in the database. After several tests, the classification was not accurate. We changed the similarity measure used for finding the relatedness between the TD features with the features for each category, but the results were not promising. After manual expectation of the extracted features, similarity with the stored features in the database we conclude that the features provided in the ACM CCS are not suitable for the automation of topic identification

Our second experiment was the selection of relevant information from test document to find similarity with individual documents in each category. Initially, we selected *title*, *keywords* and *abstract*, after a set of experiments we concluded that abstract diversify our classification results. When we tested the similar approach by excluding the *abstract*, the results were satisfying. *Abstract* contains a lot of text, which diversify the classification results. Another reason to exclude the *abstract* from text classification is the similarity measure which we used for our experimentations. After analyzing the result from our second experimentation for each individual category we observed that the categories having larger number of documents as compared to other categories, their results were comparably poor with the categories having less number of documents. The classification of a category having less number of documents was remarkable.

In our third set of experimentation, we selected the same number of document from each category. This time the classification for each category was relevant, closer to the results with other categories. Detail result for each category is shown in Fig. 7. The instance (*x y --- value*) in a cell represents *x* for original category, *y* for the system identified category and *value* representing the maximum average similarity. In some categories (*D, G, and K*) the misclassification error is relatively high as compared to the remaining categories.

One major problem in this experimentation is the error rate in training dataset. As previously used techniques for category identification were manual, in some of the cases in training documents we noticed that the assigned categories

```
For each category
       Initialize each category with minimum number of documents in a category among all
       the categories C
End

Do
          For each category
              Calculate similarity (Levenshtein distance) of TD with each document
               calculate the average similarity for each category as new pBest using Eq. 2
  End

Choose the category with the best average similarity value as gBest using Eq. 3

        For each category
            Calculate TD velocity (relevant categories for next level) using Eq3
            Update category information of the new TD in database
        End
While leaf level is not attained
```

Fig. 5. Classification of new document algorithm.

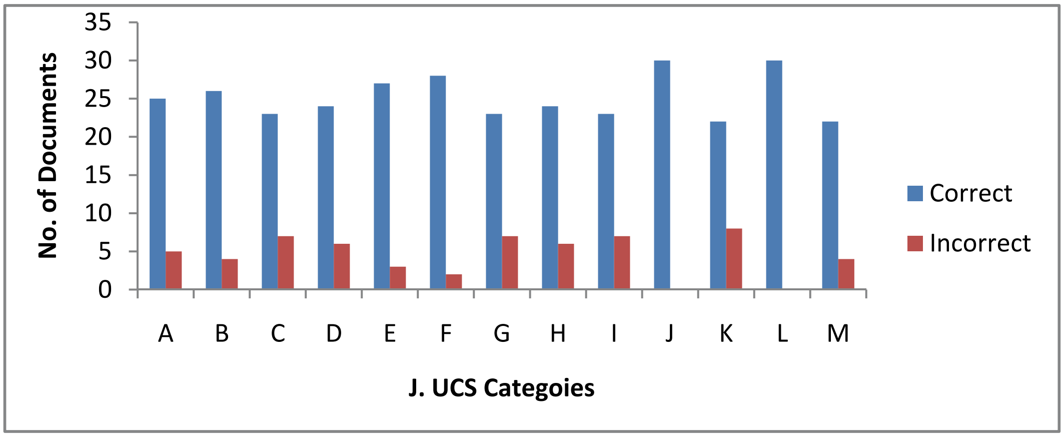
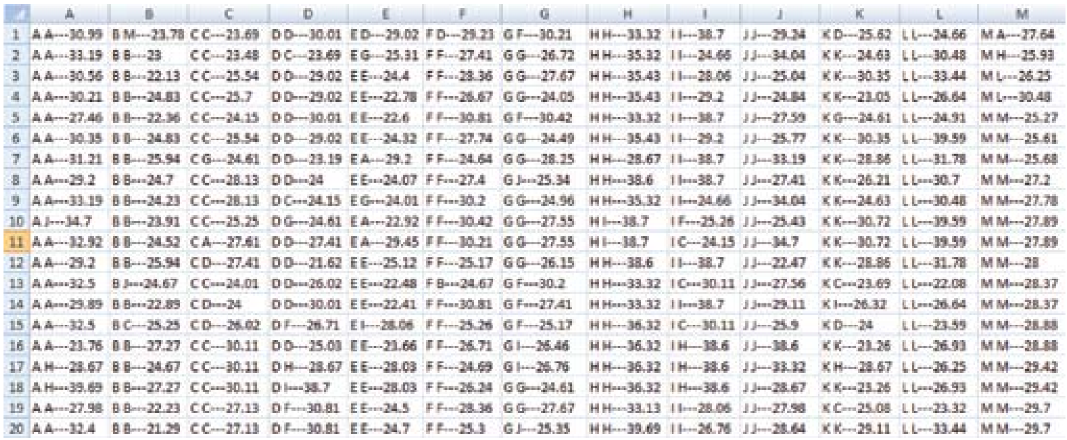


Fig. 6. Result of test documents classification with each category.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	A A---30.99	B M---23.78	C C---23.69	D D---30.01	E D---29.02	F D---29.23	G F---30.21	H H---33.32	I I---38.7	J J---29.34	K D---25.62	L L---24.66	M A---27.64
2	A A---33.19	B B---23	C C---23.48	D C---23.69	E G---25.31	F F---27.41	G G---26.72	H H---35.32	I I---24.66	J J---34.04	K K---24.63	L L---30.48	M H---25.93
3	A A---30.56	B B---22.13	C C---25.54	D D---29.02	E E---24.4	F F---28.36	G G---27.67	H H---35.43	I I---28.06	J J---25.04	K K---30.35	L L---33.44	M L---26.25
4	A A---30.21	B B---24.83	C C---25.7	D D---29.02	E E---22.78	F F---26.67	G G---24.05	H H---35.43	I I---29.2	J J---24.84	K K---23.05	L L---26.64	M L---30.48
5	A A---27.46	B B---22.36	C C---24.15	D D---30.01	E E---22.6	F F---30.81	G F---30.42	H H---33.32	I I---38.7	J J---27.59	K G---24.61	L L---24.91	M M---25.27
6	A A---30.35	B B---24.83	C C---25.54	D D---29.02	E E---24.32	F F---27.74	G G---24.49	H H---35.43	I I---29.2	J J---25.77	K K---30.35	L L---39.59	M M---25.61
7	A A---31.21	B B---25.94	C G---24.61	D D---23.19	E A---29.2	F F---24.64	G G---28.25	H H---28.67	I I---38.7	J J---33.19	K K---28.86	L L---31.78	M M---25.68
8	A A---29.2	B B---24.7	C C---28.13	D D---24	E E---24.07	F F---27.4	G J---25.34	H H---38.6	I I---38.7	J J---27.41	K K---26.21	L L---30.7	M M---27.2
9	A A---33.19	B B---24.23	C C---28.13	D C---24.15	E G---24.01	F F---30.2	G G---24.96	H H---35.32	I I---24.66	J J---34.04	K K---24.63	L L---30.48	M M---27.78
10	A J---34.7	B B---23.91	C C---25.25	D G---24.61	E A---22.92	F F---30.42	G G---27.55	H I---38.7	I F---25.26	J J---25.43	K K---30.72	L L---39.59	M M---27.89
11	A A---32.92	B B---24.52	C A---27.61	D D---27.41	E A---29.45	F F---30.21	G G---27.55	H I---38.7	I C---24.15	J J---34.7	K K---30.72	L L---39.59	M M---27.89
12	A A---29.2	B B---25.94	C D---27.41	D D---21.62	E E---25.12	F F---25.17	G G---26.15	H H---38.6	I I---38.7	J J---22.47	K K---28.86	L L---31.78	M M---28
13	A A---32.5	B J---24.67	C C---24.01	D D---26.02	E E---22.48	F B---24.67	G F---30.2	H H---33.32	I C---30.11	J J---27.56	K C---23.69	L L---22.08	M M---28.37
14	A A---29.89	B B---22.89	C D---24	D D---30.01	E E---22.41	F F---30.81	G F---27.41	H H---33.32	I I---38.7	J J---29.11	K I---26.32	L L---26.64	M M---28.37
15	A A---32.5	B C---25.25	C D---26.02	D F---26.71	E I---28.06	F F---25.26	G F---25.17	H H---36.32	I C---30.11	J J---25.9	K D---24	L L---23.59	M M---28.88
16	A A---23.76	B B---27.27	C C---30.11	D D---25.03	E E---23.66	F F---26.71	G I---26.46	H H---36.32	I H---38.6	J J---38.6	K K---23.26	L L---26.93	M M---28.88
17	A H---28.67	B B---24.67	C C---30.11	D H---28.67	E E---28.03	F F---24.69	G I---26.76	H H---36.32	I H---38.6	J J---33.32	K H---28.67	L L---26.25	M M---29.42
18	A H---39.69	B B---27.27	C C---30.11	D I---38.7	E E---28.03	F F---26.24	G G---24.61	H H---36.32	I H---38.6	J J---28.67	K K---23.26	L L---26.93	M M---29.42
19	A A---27.98	B B---22.23	C C---27.13	D F---30.81	E E---24.5	F F---28.36	G G---27.67	H H---33.13	I I---28.06	J J---27.98	K C---25.08	L L---23.32	M M---29.7
20	A A---32.4	B B---21.29	C C---27.13	D F---30.81	E E---24.7	F F---25.3	G J---25.35	H H---39.69	I I---26.76	J J---28.64	K K---29.11	L L---33.44	M M---29.7

Fig. 7. Detail result of random selected document in each category along with their results.

to documents are not relevant, while in some cases it was observed that a document was assigned to some extra categories, beside their main categories. Result of proposed technique can be improved by removing the error rate from the training dataset.

Our experimental results are better than the Bayesian approach presented in [13] with 83.75% classification accuracy tested on four categories. Similarly, the proposed approach accuracy 84.71% is better than Bayesian network learned from data with 76.25% accuracy and naïve Bayesian classifier with 82.5% accuracy respectively. Majority best evidence, majority Genetic programming approach and SVM results are compared with the proposed approach. Majority best evidence, Majority GP and SVM [26, 31] having performance accuracy of 53.60%, 57.74% and 57.74%respectively. The compared results with the approaches for scientific publication classification are given in Table 1. In our experiments, we have included all of the categories provided in J.UCS and provided results for each category. The other advantage of the proposed approach is to overcome the multi-class and multi-level classification of scientific publication to the taxonomy.

Table 1. Comparison of different approaches.

Approaches	Number of Categories	Average Accuracy
Bayesian Approach	4	83.75%
Bayesian Network learned from Data	4	76.25%
Naïve Bayesian	4	82.50%
Majority Best Evidence	11	53.60%
Majority GP	11	60.81%
SVM	11	57.74%
Proposed Approach	13	84.71.%

7. SUMMARY

Classification is, in general, a central problem in different domains. The multi-class classification is different to the ordinary classification problem. Similarly, classifying the document at different levels is also an important issue to many classification problems. Therefore, in this paper we have proposed a solution for both multi-class classification and multi-level classification. Our classification technique is an enhanced from of PSO in the context of document classification. Efficiency is achieved by reducing the size of features set and by considering the minimum number of documents in all categories. This solution can be applied for different domain where an instance can belong to multiple categories along with multi-level classification. We have implemented and tested our technique which provides better results as compared to existing techniques. This work will help authors in selecting the correct category for papers. Correct classification can, in terms, be quite useful for document retrieval and analysis.

8. REFERENCES

1. Fay, R. P. Syad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Ed). *Advances in Knowledge Discovery and Data Mining.* American Association of Artificial Intelligence, Massachusetts Institute of Technology, AAAI/MIT Press (1996).
2. Koller, D. & M. Sahami. Hierarchically classifying documents using very few words. In: *Proceedings of ICML-97, 14th International Conference on Machine learning*, Nashville, USA, 170-178 (1997)
3. Baeza-Yates, R. & B. Ribeiro-Neto. Modern Information Retrieval. Addison Wesleym, New York, USA p. 463 (1999).
4. Han, J. & M. Kamber. Data Mining: Concepts and Techniques. *Morgan Kaufmann Publishers*, San Francisco, California, USA (2001).
5. Gerstl, P. M. Hertweck, & B. Kuhn. Text mining: grundlagen, verfahren und anwendungen. *Praxis der Wirtschafts informatik- Business Intelligence* 39: 222 38-48 (2001).
6. Kononenko, I. Comparison of inductive and naïve bayesian learning approaches to automatic knowledge acquisition, In: Current Trends in Knowledge Acquisition. IOS Press, Amsterdam, The Netherlands (1990).
7. Wang, Z., Q. Zhang, & D. Zhang. A PSO-based web document classification algorithm. In: *Proc. IEEE Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, Qingdao, China, p. 659-664 (2007).
8. Cortes, C. & V. Vapnik. Support vector networks. *Machine Learning* 20: 273-297 (1995).

9. Salton, G. Developments in automatic text retrieval. *Science* 253: 974-980 (1990).
10. Azeem, S, & S. Asghar. Evaluation of structured and un-structured document classification techniques. *Proceedings of the 2009 International Conference on Data Mining (DMIN'09),* Las Vegas, Nevada, USA, p. 448-457 (2009).
11. Coulter, A. Computing Classification System 1998: Current Status and Future Maintenance. *Report of the CCS Update Committee, Computing Reviews*, New York, USA, p. 1-5 (1998).
12. Sebastiani, F. Machine learning in automated text categorization. *Technical Report IEI-B4-31*, Consiglio Nazionaledelle Ricerche, Pisa, Itlay (1999).
13. Kok-Chin, K. & T. Choo-Yee. A Bayesian Approach to Classify Conference Papers. In: *Proc.5th Mexican International Conference on Artificial Intelligence*, Apizaco, Mexico, p. 1027-1036 (2006).
14. Zhi-Hong, D. S. Tang, D. Yang, M. Zhang, X. Wu & M. Yang. A Linear Text Classification Algorithm Based on Category Relevance Factors. In: *Proceedings of the 5th International Conference on Asian Digital Libraries: People, Knowledge and Technology*, London, UK 2555: 88-98 (2002).
15. De Falco, I. A. D. Cioppa, & E. Tarantino. Evaluation of particle swarm optimization effectiveness in Classification. *Lecture Notes in Computer Science* 3849: 164–171 (2006).
16. Baoping, Z. M. Andre, Goncalves, W. Fan, Y. Chen, E. Calado & M. Cristo. Combining structural and citation-based evidence for text classification. In: *Proceedings of the thirteenth ACM international conference on Information and knowledge management (CIKM '04).* New York, USA, p. 162-163 (2004).
17. Ziqiang, W. & S. Xia. Document classification algorithm based on NPE and PSO. In: *E-Business and Information System Security,* Wuhan, China, p. 1-4 (2009).
18. Vizine, A. de-Castro, L. Gudwin, R. Text Document Classification using Swarm Intelligence. In: *Proceedings of 2005 IEEE International Conference on Integration Of Knowledge Intensive Multi-agent Systems,* Waltham, Massachusetts. p. 18-21 (2005).
19. Everhart, R. & J. Kennedy. A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro machine and Human Science*, Nagoya, Japan, p. 39-43 (1995).
20. Kennedy, J. The particle swarm: Social adaptation of knowledge. In: *Proceedings of IEEE International Conference on Evolutionary Computation*, Indianapolis, IN, USA, p. 303-308 (1997).
21. Afzal, T. H., W. Maurer & N. Balke. Kulathuramaiyer, Improving Citation Mining. In: *Proc. International Conference on Networked Digital Technologies,* Ostrava, Czech Republic, p. 116-121 (2009).
22. Wang, Z. Improving on latent semantic indexing for chemistry portal. *Master of Engineering dissetation, Institute of Process Engineerin*g, Chinese Academy of Sciences, Beijing, China (2003).
23. Porter, M. An algorithm for suffix stripping. *Readings in Information Retrieval, Morgan Kaufmann Publishers,* San Francisco, California, USA, p. 313-316 (1997).
24. Guerrero-Bote, V. F. Moya-Anegon & V. Herrero-Solana. Document organization using Kohonen's algorithm. *Information Processing and Management* 38(1) 79-89 (2002).
25. Dumais, T. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments and Computers* 23(2): 229-236 (1991).
26. Zaiane O. & M. Antonie. Classifying text documents by associating terms with text categories. In: *Proceedings of the 13th Australasian Database Conference,* Melbourne, Australia, p. 215-222 (2002).
27. The Reuters-21578 Text Categorization Test Collection. http://www.research.att.com/~lewis/reuters21578.htmlretirved (2009).
28. N. A. Sajid, T. Ali, T. Afzal, M. Ahmad, & M. Qadir. Exploiting reference section to classify paper's topics. In: *The International Conference on Management of Emergent Digital EcoSystems (MEDES' 11)*, San Francisco, California, USA (2011).
29. Rolling, L. Indexing consistency, quality and efficiency. *Information Processing and Management* 17(2): 69-76 (1981).
30. Levenshtein, I. Binary Codes capable of correcting deletions, insertions, and reversals. *Cybernetics and Control Theory* 10(8): 707–710 (1966).
31. Senthamarai K, & N. Ramaraj. Similarity based technique for Text Document Classification. *International Journal of SoftComputing* 3(1): 58-62 (2008).