International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 09, September-2015

A Simple Approach for Scientific Document Categorization

Arlina D'cunha
Computer Department
St. Francis Institute of Technology
Mumbai, India

Dr. A. K. Sen
St. Francis Institute of Technology
Mumbai, India

***Abstract*— Classification is the alignment of data or items in predefined labeled groups based on resemblances. Exponential progression amount of scientific documents leads to uncontrollable physical classification. Feature extraction is the crucial condition of automatic document classification. TF-IDF (term frequency-inverse document frequency) is frequently used to represent the text feature weight. This paper proposes a new yet simple feature weighting scheme by modifying TF-IDF formula. The experimental results show that the modified method improves the accuracy and other parameters.**

Keywords—Classification;Scientific document;tf-idf

I. INTRODUCTION

Text mining is a flourishing new area that efforts to glean meaningful information from natural language text. It may be roughly characterized as the process of analyzing text to extract information that is useful for specific purposes. Text is amorphous, ambiguous and rough to deal with algorithmically. Yet it is the most common vehicle for the formal exchange of information. The area of text mining usually deals with texts whose function is the communication of realistic information or opinions. Information Retrieval (IR) is the science of searching for data in documents, documents itself or metadata which define documents and classification i.e. assemblage of information in predefined labelled classes based on likenesses leads to good IR. Classification of scientific documents is a task done by professional libraries where the standard for classifying documents is subject to several features and attributes.

A. Overview of Scientific Document

Superlatively, scientific documents should be reasonable to nonscientist individuals who may be involved in scientific issues, or may be in a position to backing scientific tasks. Expansion of the scientific document has been inspirational, both in diversity of content and in the complexity with which this content is discussed. Nevertheless, at origin these documents are often crude activities. Authors who have facts and figures vital to the growth of the human race are often displeased by boundaries of time and linguistic in their efforts to be perceived. Scientific Documents that detail investigational work are often arranged chronologically in five sections: first, Introduction; then Tools and Techniques, Results, and Discussion and lastly, Conclusion.

The Introduction section explains the motivation for the effort presented and makes readers for the organization of the paper.

The Tools and Techniques section offers suitable structures for other scientists to reproduce the experiments presented in the paper.

The Results and Discussion sections present and converse the research results, respectively. They are frequently combined into one section, but readers can hardly make logic of results alone without additional clarification.

The Conclusion section presents the consequence of the work by concluding the findings at a higher level of abstraction and by linking these findings to the motivation stated in the Introduction.

B. Document Classification Overview

Document (or text) classification runs in two phases: the training phase and the testing (or classification) phase.

Through training, a feature extractor is used to transform every document to a feature set, which capture the simple details about each document that should be used to classify it. Feature collections and labels are served to the classification model to produce a model. During testing, the same feature extractor is used to alter non-classified documents to feature sets followed by the model to assign tags to input documents.

C. TF-IDF for Feature Extraction [1][2]

TF can be calculated as per equation (1) and equation (2) gives IDF

$$TF_{t,d} = \frac{t_d}{T_d} \quad (1)$$

Where t_d is number of times term t appears in a document d and T_d is total number of terms in the document d.

$$IDF_t = \log \frac{N}{df_t} \quad (2)$$

The feature vector of document d from collection D with n different terms is denoted as follows:

$$d_d = [w_{1,d}, w_{2,d} \ldots\ldots w_{n,d}] \quad (3)$$

$$w_{t,d} = TF_{t,d} * IDF_t \quad (4)$$

$W_{t,d}$ is the weight of term *t* of document *d*.

Efficient information retrieval needs efficient classification. Classification of scientific documents is the grouping of information or documents in predefined labeled categories based on similarities. Exponential development of scientific document collection leads to unmanageable manual classification. Thus automatic classification of scientific documents into categories is an increasingly important task. Feature extraction is the central prerequisite of automatic document classification. TF-IDF (term frequency-inverse document frequency) is commonly used to express the text feature weight. This research proposes a new feature weighting method by modifying TF-IDF formula.

II. LITERATURE SURVEY

Literature [3] gives a brief overview of scientific document classification. This paper undergoes every phase of the methodology in order to be classified and instantiated in ontology that models knowledge matters. Once the ontology is populated it can be used to performed implications and obtained hidden knowledge from the papers.

A. *Weight Computation*

The Vector Space Model (VSM) proposed by Salton [1] is a common method for document representation in classification where each document is represented as a vector of features. Each feature is associated with a weight. Typically these features are simple words in document. The feature weight can be just a Boolean value indicating the presence or absence of the word in document, its existence number in document or it can be calculated by a formula like the well-known TF-IDF [1] [2] method which treats a document as a "bag of terms" [4].

B. *Classification Algorithms*

After extracting all features from document and calculating their weight document vector is constructed to feed into classifier model. The Naïve Bayes Classifier [5][6] is the simplest probabilistic classifier used to classify the text documents. It severe assumption that each feature word is independent of other feature words in a document [7]. The idea is to use the joint probabilities of words and categories to estimate the class of a given document. Given an unknown document sample *D*, Naive Bayesian classification will classify *D* as the class with the highest posterior probability i.e., Bayesian classification assigns the unknown sample to the class *Ci* if and only if $P(Ci|D) > p(Cj|D)$, where $1 \le j \le m$, $j \ne i$, the class *Ci* is called as the maximum posteriori assumption when $P(Ci|D)$ is the largest. $P(Ci|D)$ is according to Bayesian theorem [8].

Instead of using frequency selection method for assigning the features generated in the training stage to their correct category, FRAM[9] assigns the features that are generated from the new given document to their categories based on the Frequency Ratio (FR) of the features that are sorted in the training stage. Assigning the features by using FRAM involves combining it with the classification process. Thus the time for the training stage will be reduced by excluding the feature selection task.

The traditional centroid-based method [10], can be observed as a specialization of Rocchio method [11] and used in numerous works on text classification [12].

III. PROPOSED TECHNIQUE

A. *Scientific Document Hierarchy Construction*

We considered a scientific document as a hierarchy in which the nodes are tagged by structural labels like title, keywords, abstract, etc. The bottom-most node contains the document text.Fig.3 illustrates one such example of scientific document hierarchy.

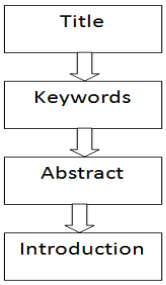

Fig.1. An example of scientific document hierarchy

B. *Feature Extraction*

After constructing the aggregated tree, we need to extract the terms in each node. We need to apply a series of preprocessing. Pre-processing may involve text-extraction, stop-word removal, tokenization, etc.

C. *Weight Computation*

Our assumption is that a term which appears in two different structural levels should have different importance. For example the word “Engineer” appears in the document title and in paragraph composes two different features and the weight of the first feature is more significant than the second. For calculating the weight of features we consider a modification of traditional tf*idf on structural element level instead of document level. So, the weight of feature will be calculated as per equation (5)

$$w_{t,e,d} = TF * IDF * ED \quad (5)$$

$$ED = \log \frac{L_d}{l_{d,e}} \quad (6)$$

ED is element depth to judge the significance of the structural element *e*. Where L_d is the depth of document hierarchy, and $l_{d,e}$ is the depth of the node e in the document .

As per our assumption, content information has least importance. Thus even if we ignore certain numbers of terms from the content and consider only *T* terms from the content for weight computation, it should not affect accurate classification prediction. Terms from title, keywords and abstract should affect the classification prediction.

After weight computation, the document vector feed into classifier model.

IV. EXPERIMENTS AND EVALUATION

A. Dataset

We considered nine categories as shown in Table 1. Including 904 scientific documents from various open access journals [13][14][15][16][17] as training dataset, another 304 documents to test the system. Pretreatment involves text extraction, tokenization and remove the stop words.

Table 1: Classification Categories under consideration

Artificial Intelligence	Database System and Data Mining	Computer Security and Cryptography
Internet, Web Services and Cloud Computing	Distributed System	Antenna
Image and Video Processing	Networking	Human-Machine Interaction and Virtual Reality

B. Model Evaluation

The performance evaluation of the classifier usually used evaluation indicator which are some quantitative index which used to evaluate the performance of classification in the testing process. The well-known use of performance evaluation indicators in the text classification contained Recall, Precision, F-Score, Specificity and Accuracy. The higher of this evaluation indicator value, the better performance of the classification model is. Formulas are as follows:

- $$Precision = \frac{True\_Positive}{True\_Positive + False\_Positive} \quad (7)$$
- $$Recall = \frac{True\_Positive}{True\_Positive + False\_Negative} \quad (8)$$
- $$Specificity = \frac{True\_Negative}{True\_Negative + False\_Positive} \quad (9)$$
- $$F-Score = \frac{2True\_Positive}{2True\_Positive + False\_Positive + False\_Negative} \quad (10)$$
- $$Accuracy = \frac{True\_Positive + True\_Negative}{N} \quad (11)$$

Where,

$N = True\_Positive + False\_Positive + False\_Negative + True\_Negative$

Other parameter used to evaluate the performance of the system is *Execution Time*.

C. Analysis of Results

The experiment included comparison of FRAM, Naive-Bayes and Centroid classification algorithms with TF-IDF and improved TF-IDF.

As we have considered certain predefined number of terms from content information, execution time of overall execution can be saved which is illustrated in table 2.

Table 2: Average Execution time (in ms) for 304 documents

	Existing TF-IDF	Improved TF-IDF	Time saved (in %)
FRAM	38.14	12.45	67.77
Naïve-Bayes	100.68	26.29	73.89
Centroid	5262.05	5069.92	3.65

As per our previous discussion, terms occurring in title, keywords and abstract have high impact on the weight in document vector, these terms play important role in classification. Thus overall performance of the system has been improved as shown in following comparative graphs.

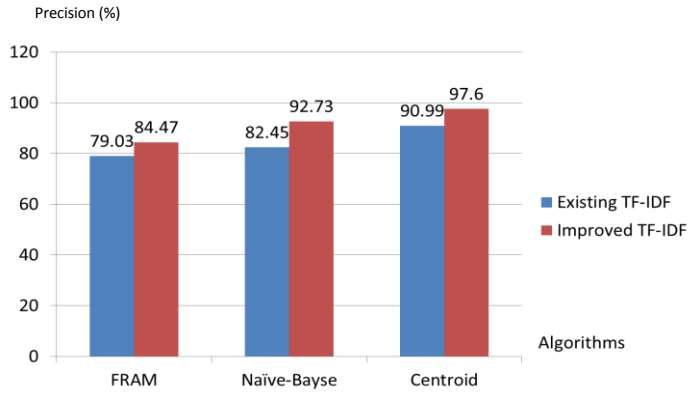

Fig 2: Average Precision comparison for different algorithms

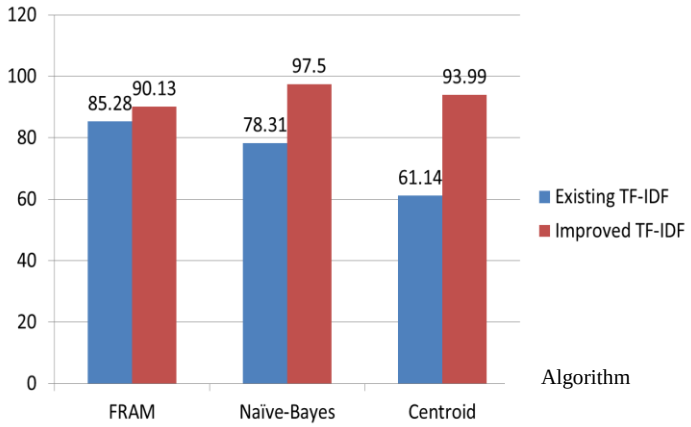

Fig 3: Average Recall comparison for different algorithms

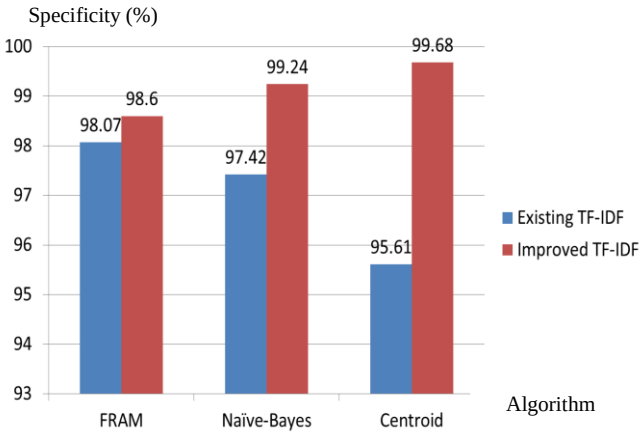

Fig 4: Average Specificity comparison for different algorithms

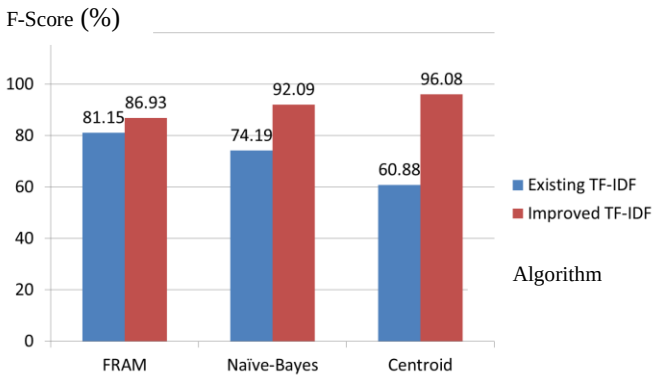

Fig 5: Average Specificity comparison for different algorithms

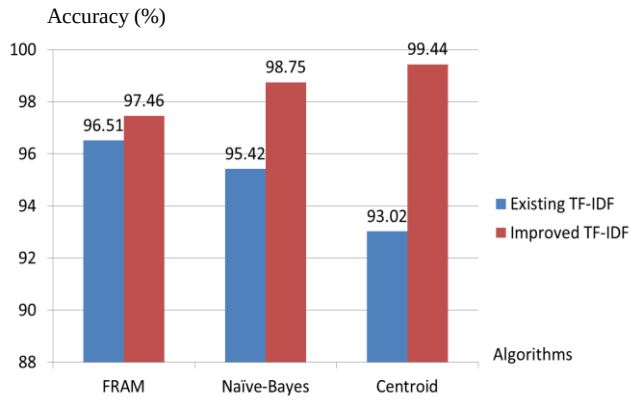

Fig 6: Average Accuracy comparison for different algorithms

The results show that using improved TF-IDF approach not only provides a more graceful and simpler solution to the classification problem, but also results in considerable performance gain in terms of classification.

V. CONCLUSION

Automatic document classification is a machine learning task that automatically assigns a given document to a set of pre-defined categories based on the features extracted from its textual content. Our proposed work involves modification of TF-IDF and its effects on three classification algorithms for scientific documents. Experiments were conducted to test Execution Time, Precision, Recall, Specificity, F- Score and Accuracy. Experimental results proved that the parameters tested were improved compared to the existing system.

ACKNOWLEGEMENT

We are grateful to Ms. Vincy Joseph and Ms.Anuradha S., Associate Professors at St. Francis Institute of Technology for their insightful comments and suggestions. We would also like to thank Mr. Abhitesh Das, Technical architect, CACTUS Communications, for his valuable guidance.

REFERENCES

[1] Salton G., "Search and retrieval experiments in real-time information retrieval" C. University, Ed., 1968, pp. 1082-1093.
[2] Salton, G., Buckley, C., "Term weighting approaches in automatic text retrieval" Information Processing and Management: an International Journal, 1988, Vol. 24, Issue 5, pp. 513-523.
[3] Juan C. Rendón-Miranda, Julia Y. Arana-Llanes, Juan G. González-Serna and Nimrod González-Franco, "Automatic classification of scientific papers in PDF for populating ontologies". 2014 International Conference on Computational Science and Computational Intelligence, Vol. 2, pp. 319-320.
[4] Yanjun Li, Congnan Luo, and Soon M. Chung, "Text Clustering with Feature Selection by Using Statistical Data," IEEE Transactions on Knowledge and Data Engineering, May 2008,Vol. 20, Issue 5, pp 641 - 652.
[5] Jingnian Chen, Houkuan Huang, Shengfeng Tian and Youli Qu, "Feature selection for text classification with Naïve Bayes", Expert Systems with Applications: An International Journal, Elsevier, 2009, Vol. 36, Issue 3.
[6] Hu, YJ. Zhou, X. L., Ling, L., Wang, X.L, "A Bayes Text Classification Method Based on Vector Space Model", Computer & Digital Engineering,2004, Vol.6, Issue 32, pp.28-30.
[7] Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaebg, "Some Effective Techniques for Naive Bayes Text Classification," IEEE Transactions on Knowledge and Data Engineering, 2006, Vol. 18, Issue 11, pp 1457 -1466.
[8] David McAllester, "Some PAC-Bayesian Theorems", Proceedings of the Eleventh Annual Conference In Computational Learning Theory,1998.
[9] Suzuki M. and Hirasawa S., "Text Categorization Based on the Ratio of Word Frequency in Each Categories," in Proceedings of IEEE International Conference on Systems Man and Cybernetics, 2007, pp. 3535-3540.
[10] Han, E.-H., Karypis, G., "Centroid-based document classification: analysis and experimental results", Principles of Data Mining and Knowledge Discovery, pp. 424–431, 2000.
[11] Rocchio J.J., Jr., "Relevance feedback in information retrieval", G. Salton (Ed.), The SMART REtrieval System: Experiments in Automatic Document Processing, Prentice-Hall, Englewood Cliffs, NJ, 1971, pp. 313–323.
[12] V. Lertnattee, T. Theeramunkong, "Improving centroid-based text classification using term distribution-based weighting and feature selection", Proceedings of INTECH-01, 2nd International Conference on Intelligent Technologies, Bangkok, Thailand, 2001, pp. 349–355.
[13] www.waset.org
[14] http://www.airccse.org/
[15] http://aisel.aisnet.org/journal/sipij/
[16] http://www.scirp.org/journal/ojapr/
[17] http://www.sersc.org/