International Journal of Hybrid Information Technology
Vol.8, No.6 (2015), pp.95-106
http://dx.doi.org/10.14257/ijhit.2015.8.6.10

Automated Classification of Research Papers Using Hybrid Algorithm

Er. Rajvir Kaur[1] and Er. Nishi[2]

[1] *Research Scholar,* [2] *Assistant Professor, Computer Science and Engineering Department*
DAV University, Jalandhar (India)
[1] *rajvir.parmar33@gmail.com,* [2] *nishi.bti.02@gmail.com*

Abstract

A lot of time of the users is consumed in searching appropriate papers related to the desired topic. It takes time to look through the paper also. In this paper, a hybrid method is introduced to classify research papers. This algorithm is designed to classify all research papers at the time of uploading in the repository. Hence it becomes easy to explore appropriate paper on a specific topic in minimum time. A data set has generated with research papers on different topics like natural language processing, machine learning, etc. The proposed algorithm passes the most frequent items fetched from the training data set to k-nearest neighbor method instead of the whole data set, to make clusters. The performance of the proposed method is compared with traditional KNN method which results the accuracy, improved by the factor of 7.46%.

Keywords: *classification, Frequent term mining, KNN, Text mining*

1. Introduction

Data mining is one of the most upcoming fields of research as abundance of data is produced by the rapid growth of networks and information technology, we need a tool to manage such a large amount of data and figure out the hidden and useful information from it. Data mining provides such a tool to the decision makers, which makes easy for them to process such a large data effectively and efficiently and extract information from it on the behalf of which decision can be taken. Data mining is multi-disciplinary field, which has great importance in other fields like machine learning, pattern recognition, neural networks, *etc.* [1].

Text mining is basically employed for extraction of interesting facts and relationships and the discovery of knowledge from large amounts of text. For this purpose, text mining employs techniques and algorithms from disciplines such as data mining, information retrieval, statistics, mathematics, machine learning and natural language processing. Text mining, also known by the name of Text Data Mining, Intelligent Text Analysis or Knowledge-Discovery in the Text (KDT), as it extract interesting and meaningful data and knowledge from text which is unstructured basically. Text mining tools Analyzes clusters of words, words in documents, etc., or documents and find similarities between the words and how they are correlated with other words or variables [2].

Text mining is identical to data mining, yet there are differences between two. The data mining tools are basically to handle structured data from databases, but text mining can work on unstructured or semi-structured data sets such as full-text documents, emails and HTML files etc. The outcome of this, text mining is a key for corporate sector to analyze the large data. Data mining is characterized as the extraction of implicit, previously unknown, and potentially useful information from data. The information is implicit in the input data: it is hidden, unknown whereas in text mining the information to be extracted is

ISSN: 1738-9968 IJHIT

clearly and explicitly stated in the text. It's not hidden at all. In data mining the information extracted must be comprehensible in that it helps to explain the data. On the other hand, in text mining the input is comprehensible.

The clustering is unsupervised learning. It is classification of items into groups such that intra similarities of items within groups increases and inter similarities between items of different groups ceases. The algorithm used for clustering documents in K-Means Algorithm. It helps in implementing semi-supervised learning clusters, to recognize relative text to find using predefined arrangements [1, 2]. It is used to cluster observations into groups of related observations without any prior knowledge of those relationships. The k-means algorithm is one of clustering techniques and is commonly used in medical imaging, biometrics and related fields. The k-means algorithm takes the input, k, and divides a set of n objects into k clusters so that the resulting intracluster correlation is high but the intercluster correlation is low.

2. Related Work

In 2007, Le Wang, *et al.,* [4], proposed simple hybrid algorithm based on k-means and top-k frequent terms for web document clustering. Frequent term sets consisting of n number of top terms, are utilized to produce initial clusters, and further polished by k-means algorithm. The final clusters produced by k-means algorithm while the intelligible description of clustering is provided by n frequent term sets.

In 2012 K. Nithya, *et al.,* [5] has designed mining model which consist of concept analysis and corpus- based concept analysis. It analyzed the terms of sentences for their semantics in sentence, documents and corpus levels and extract feature vectors. Then k-nearest neighbor is applied for classification.

In 2013, E. Alan Calvillo, *et al.,* [6] purposed a better classification of research papers by providing a architecture that works with a knowledge database related topics of databases, operating system and programming. This architecture works using k-means algorithm for clustering of research papers.

In 2013 N. Arunachalam [7] and his team has came with an idea of ontology based text mining approach to cluster research proposals on the basis of similarities in research area. They used ontology which is a knowledge repository containing concepts and terms and relationships between the concepts which makes the task of searching similar pattern of text effective, efficient and interactive.

In 2014 Jadhav Bhushan, *et al.,* [8] focused on the necessity of the new search engine which is based n the fastest reading algorithm and provide best results. They purposed the architecture that worked on knowledge database system containing topics of operating system, database and programming. It searches base keyword of content from knowledge database and uses clustering and text mining.

In 2014 Dr. T. Lalitha, *et al.,* [9] presented a architecture for text mining called DISCOTEX, *i.e.,* Discovery from Text Extraction which used a learned information extraction system to convert text into data which is more structured for mining interesting relationships. It combines the information extraction module with standard rule induction module to improve recall of underlying system.

In 2014 Ruchika Mavis Daniel, *et al.,* [10] presented a model of a hybrid text document clustering approach which comprises of text summarization, fuzzy calculation and traditional k-means algorithm, for text search process. The hybrid approach is more efficient for various real time applications.

3. Problem Outline

Thousands of research papers are published in different journals, websites, magazines like Springer, IEEE (International Electrical and Electronic Engineering), Elsevier and many more international and national recognized journals. The information which the

user's needs is not only present on the specialized research journals but also on the websites. The search engine locates this information using snowball effect which also displays the content which may not be useful to user. Hence it takes hours, while searching for papers for specific topics. The main objective is to minimize the time spent on searches that is to reduce the response time for locating scientific articles. This can be accomplished if the research papers are properly updated in the repository, when the user demands for specific paper it would be easy to locate it in minimum time.

4. Contribution

This paper uses the concept of frequent term sets along with the text mining techniques, classification and clustering algorithm. The main objective of this work is the better classification of papers in repositories so that when the user searches for the specific topic, he gets the desired results in minimum time. The architecture model of the proposed system is shown in Figure 1. The input documents are the research papers which are on different topics. The feature extraction component in the architecture refers to the frequent terms set. Frequent terms set comprises of the most frequent terms in the data set. Then clustering algorithm is applied. This whole process is the training phase of the model. During the testing phase the features are extracted from the research papers and are passed to the classifier which assigns the label to the research paper, *i.e.*, the category to which the paper belongs.

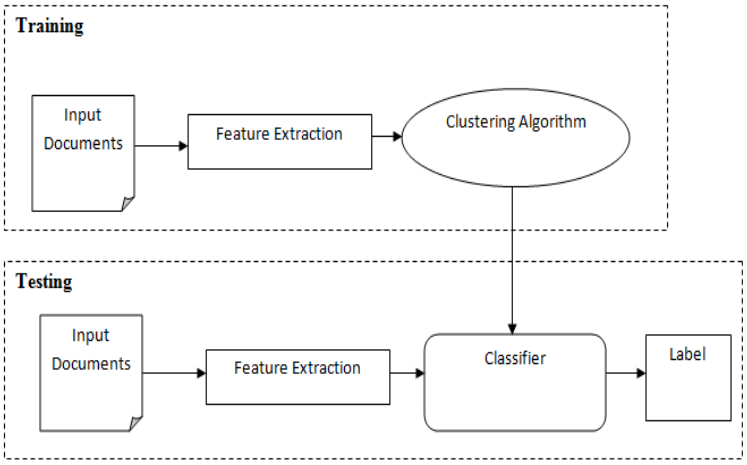

Figure 1. The Architecture Model for Classification

The proposed algorithm is named as hybrid algorithm which is the combination of two algorithms, frequent terms mining and K means algorithm. The frequent terms mining algorithm mines the frequent terms set form the data set. The frequent terms data set is then passed to the k-means algorithm for the initial clustering.

The hybrid algorithm works:

a. Prepare the data set *i.e.*, preprocessing.
b. Mine the frequent terms from the dataset.
c. Apply KNN algorithm.
   i. Choose the number of clusters, k.
   ii. Randomly generate k clusters and determine the cluster centers (centroids), where a cluster's centroid is the mean of all points in the cluster.
   iii. Repeat the following until no object moves (*i.e.*, no object changes its cluster)
      - Determine the Euclidean distance of each object to all centroids.

- Assign each point to the nearest centroid.
- Re-compute the new cluster centroids.

Thus, according to the K Means algorithm assigns each point to a cluster whose center (also called centroid) is nearest. The centroid of a cluster is the average of all the points in the cluster based on the Euclidian distance measure. Thus, in each loop of step 3 above, the algorithm aims at minimizing the following function for k clusters and n data points.

$$J = \sum_{j=1}^{j=k} \sum_{i=1}^{i=n} \left\| x_i - c_j \right\|^2 \quad (1)$$

Where ||xi-cj|| is a chosen distance measure (*e.g.*, Euclidean measure) between data point xi from cluster cj.

The proposed methodology used for classification of research papers so as to upload them in the cluster to which they belong is represented in the flowchart shown in Figure 2.

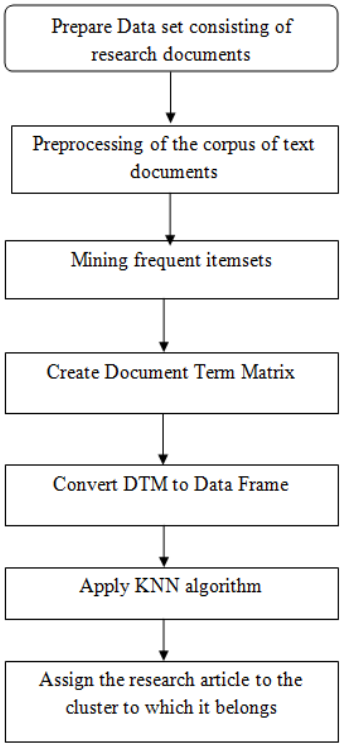

Figure 2. The Proposed Approach

For the effective and efficient classification of research papers, the frequent term set is combined to the clustering algorithm i.e. KNN algorithm.

The proposed approach is:

4.1. Prepare Dataset

As per topic is research papers classification using text mining, dataset consist of research papers on different topics like artificial intelligence, computer networks, data mining, software engineering, machine learning, digital image processing, cryptography, cloud computing, *etc*.

4.2. Preprocessing

Preprocessing is preparing data for text mining by removing the terms that appear too often and support no information for the task. During preprocessing numbers, punctuations, whitespace and stopwords of English like is, am, are, *etc.*, are removed from the corpus. The transformations are applied sequentially to remove unwanted characters

from the text. Stemming is also performed for the reduction of words into their root. Example agree, agrees, agreeing can be reduced to its base form or stem agree.

Table 1. Data Preprocessing

Research papers topics(Labels)	Document/term before preprocessing	Document/term after preprocessing
Ai(AI)	76/4097168	76/2945528
Cloud Computing(CC)	108/6030872	108/4467232
Cryptography (CRYPTO)	167/21245000	167/14912576
DIP(DIP)	117/4466944	117/3327480
Data mining(DM)	180/9176096	180/6560856
Network Security(NS)	74/4753848	74/3424672
Software engineering (SOFT)	107/9796512	107/66218816
Stenography (STEGNO)	101/7316232	101/5248112
VLSI(VLSI)	80/4176224	80/3077944
Wireless Networks(WN)	120/9636568	120/6614144

4.3. Mining Frequent Term Set

After preprocessing the most frequent items are mined. Then these frequent term sets are used for partitions, where documents can be clustered within the partition. Top–f frequent words are taken from each document. The mined frequent terms are arranged in descending order based on their support level for every length of term sets.

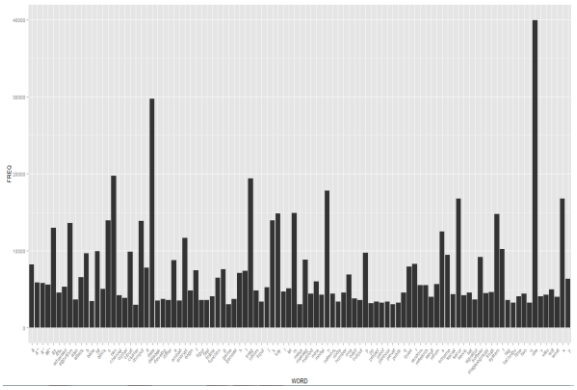

Figure 3. Terms-frequency Graph

4.4. Create Document Term Matrix

A document term matrix is simply a matrix with documents as the rows and terms as the columns and a count of the frequency of words as the cells of the matrix. Document term matrix (DTM) is created by considering the frequent terms mined in previous step. It reduces the number of terms being processed and hence reduces the response time and increases the speed.

4.5. Convert DTM To Data Frame

Then the data frame is constructed using the DTM. Data frame is a Table or 2-D array-like structure, in which each row represents one case and each column have measurements on one variable or attribute. A data frame is a collection of column vectors. Partition the dataset into two halves: training data and testing data. The training data has labels and are used to train the model. The testing data is passed to model to check whether it correctly classify research papers. The class labels of the testing data are known in advance.

Table 2. Training and Testing Data

Research papers topics	Training Data (no. of documents)	Testing Data(no. of documents)
AI	53	23
CC	76	32
CRYPTO	117	50
DIP	84	33
DM	132	48
NS	53	21
SOFT	75	32
STEGNO	95	33
VLSI	59	21
WN	97	23

4.6. Apply KNN Algorithm

Apply KNN (k-nearest neighbor) algorithm for classification of research papers. The initial clustering is done using the frequent terms set. Compare the predicted results with actual results.

5. Application of the Proposed Algorithm

The proposed hybrid algorithm serves as the first stage for the fully automated classification of the documents as shown in Figure 4. If the primary classification system yields better result than the overall output will be improved of fully automated classification system [15].

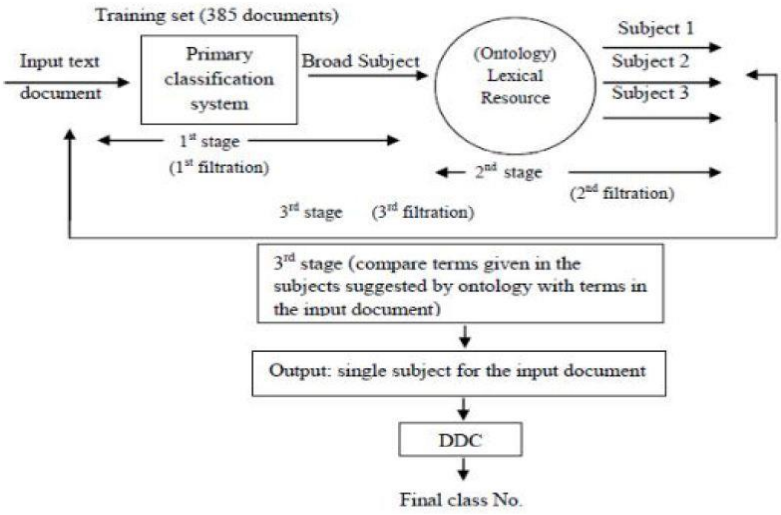

Figure 4. Fully Automated Classification System

6. Experimental Results

We have practiced two algorithms, *i.e.*, KNN algorithm and hybrid Algorithm, for the classification of research papers. The algorithms are implemented on the data sets given in Table 2. We implemented the both algorithms on a window 7 PC with an Intel® Core (TM) i5-2430M CPU @ 2.4GHz processor, running R 3.1.2 version and RStudio. R is a programming language mostly used by statisticians for data mining and statistics. Table 3 and Table 4 shows the confusion matrix generated by KNN algorithm and hybrid algorithm respectively.

Table 3. Confusion Matrix of KNN Algorithm

	Predicted class(KNN algorithm)									
Actual class	***AI***	***CC***	***CRYPTO***	***DIP***	***DM***	***NS***	***SOFT***	***STEGNO***	***VLSI***	***WIRELESS***
AI	15	0	3	0	3	0	1	0	0	1
CC	0	25	4	0	2	1	0	0	0	0
CRYPTO	0	2	38	0	0	3	1	4	2	0
DIP	0	0	3	30	0	0	0	0	0	0
DM	1	0	0	0	45	1	1	0	0	0
NS	0	2	1	0	0	17	0	0	0	1
SOFT	0	0	1	0	0	0	31	0	0	0
STEGNO	0	0	3	0	0	0	0	30	0	0
VLSI	1	0	0	1	1	0	0	0	17	1
WIRELESS	2	0	0	0	0	0	0	0	1	20

Table 4. Confusion Matrix of Hybrid Algorithm

	Prediction class(Hybrid Algorithm)									
Actual class	***AI***	***CC***	***CRYPTO***	***DIP***	***DM***	***NS***	***SOFT***	***STEGNO***	***VLSI***	***WIRELESS***
AI	*21*	*1*	*1*	*0*	*0*	*0*	*0*	*0*	*0*	*0*
CC	*1*	*29*	*0*	*0*	*1*	*1*	*0*	*0*	*0*	*0*
CRYPTO	*0*	*2*	*41*	*0*	*0*	*5*	*1*	*1*	*0*	*0*
DIP	*1*	*0*	*1*	*30*	*0*	*0*	*0*	*0*	*1*	*0*
DM	*0*	*0*	*0*	*0*	*47*	*1*	*0*	*0*	*0*	*0*
NS	*0*	*2*	*1*	*0*	*0*	*18*	*0*	*0*	*0*	*0*
SOFT	*0*	*0*	*0*	*0*	*0*	*0*	*32*	*0*	*0*	*0*
STEGNO	*0*	*0*	*4*	*0*	*0*	*0*	*0*	*28*	*0*	*1*
VLSI	*1*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*19*	*1*

WIRELESS	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*23*

The Table 5 and Table 6 show the computed results of KNN Algorithm and Hybrid algorithm. The experimental results include the recall, precision, F-Score, specificity negative predicted value prevalence, detection Rate, detection Prevalence and balanced accuracy of all classes, *i.e.*, Cloud Computing, data mining, wireless networks, stenography, DIP.

Table 5. Computed Results of Knn Algorithm

Class	Recall	Precision	F-Score	Specficity	Neg Pred Value	Prevalance	Detection Rate	Detection Prevalence	Balanced Accuracy
AI	0.789	0.652	0.7143	0.9864	0.97306	0.07278	0.0475	0.06013	0.8193
CC	0.862	0.781	0.8197	0.9859	0.976	0.10127	0.0791	0.09177	0.8836
CRYPTO	0.717	0.760	0.7379	0.9436	0.954	0.1582	0.1203	0.1677	0.8518
DIP	0.968	0.909	0.9375	0.9965	0.989	0.10443	0.0949	0.09810	0.9529
DM	0.882	0.938	0.9091	0.9776	0.988	0.1519	0.1424	0.1614	0.9576
NS	0.773	0.809	0.7907	0.9831	.9863	0.06646	0.0538	0.06962	0.8963
SOFT	0.913	0.968	0.9394	0.9894	0.997	0.1013	0.0981	0.1076	0.9791
STEGNO	0.882	0.909	0.8955	0.9859	0.989	0.10443	0.0949	0.10759	0.9475
VLSI	0.850	0.809	0.8293	0.9898	0.986	0.06646	0.0538	0.06329	0.8997
WIRELESS	0.869	0.869	0.8696	0.9898	0.989	0.07278	0.0633	0.07278	0.9296

Table 6. Computed Results of Hybrid Algorithm

Class	Recall	Precision	F-Score	Specficity	Neg Pred Value	Prevalance	Detection Rate	Detection Prevalence	Balanced Accuracy
AI	0.875	0.913	0.8936	0.9897	0.993	0.07278	0.0665	0.07595	0.9514
CC	0.853	0.906	0.8787	0.9823	0.989	0.10127	0.0918	0.10759	0.9443
CRYPTO	0.854	0.820	0.8367	0.9737	0.966	0.1582	0.1297	0.1519	0.8968
DIP	1.000	0.909	0.9523	1.0000	0.989	0.10443	0.0949	0.09494	0.9546
DM	0.979	0.979	0.9792	0.9963	0.996	0.1519	0.1487	0.1519	0.9877
NS	0.720	0.857	0.7826	0.9762	0.989	.06646	0.0569	0.07911	0.9167
SOFT	0.969	1.000	0.9846	0.9965	1.000	0.1013	0.1013	0.1044	0.9982
STEGNO	0.966	0.848	0.9032	0.9964	0.982	0.10443	0.0886	0.09177	0.9225
VLSI	0.950	0.905	0.9268	0.9966	0.993	0.06646	0.0601	0.06329	0.9507

WIRELESS	0.920	1.000	0.9583	0.9931	1.000	0.07278	0.0728	0.07911	0.9966

The recall of Network Security is low as compared to other classes. The reason behind it is, it becomes difficult to distinguish whether the paper belongs to cryptography or network security as both the disciplines are interrelated. The network security is implemented with the help of cryptographic techniques.

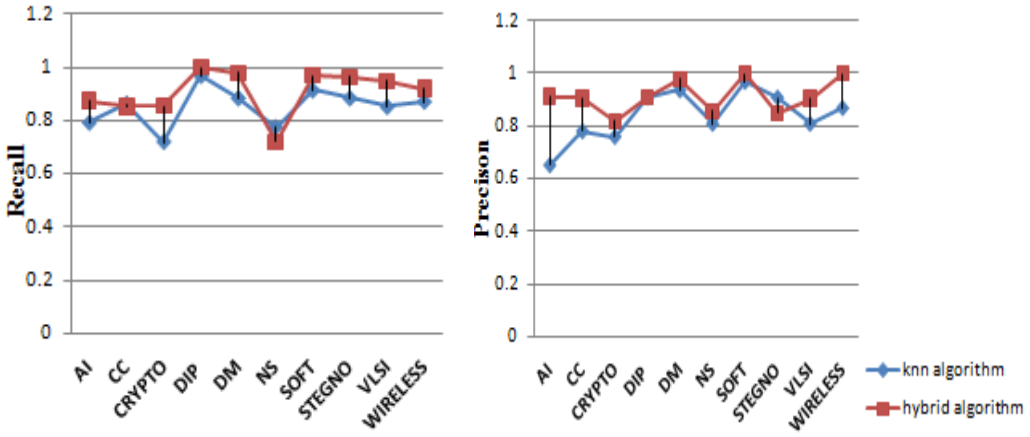

Figure 5. Comparing KNN and Hybrid Algorithm (on Left) Recall, (on Right) Precision

The Figure 5 shows the line graphs of the recall and precision of KNN and the hybrid algorithm against all the classes, *i.e.*, the ai, cc, crypto, *etc*.

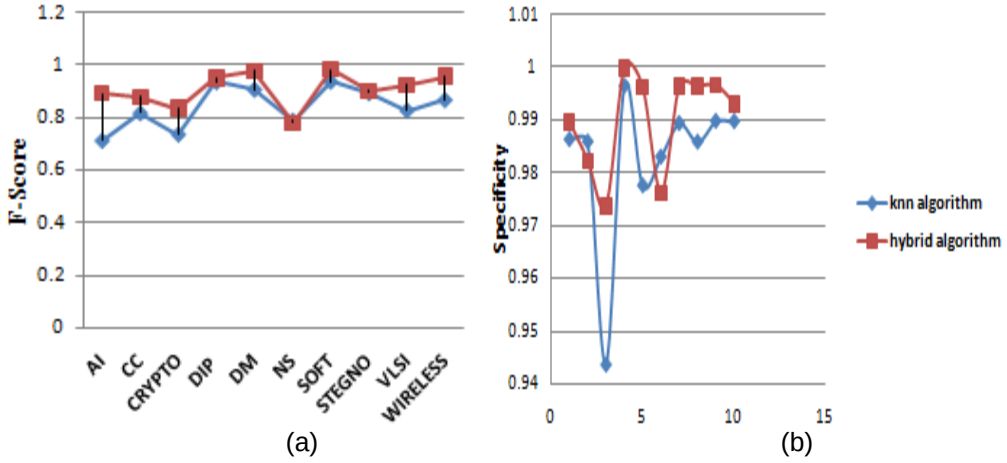

Figure 6. Comparsion of KNN and Hybrid against Parameter (a) F-score (b) Specificity

The Figure 6 shows the line graph of the F-Score and the scatter plots of Specificity of both algorithms, KNN algorithm and hybrid algorithm. The graphs show the results of hrbrid algorithms are far better than the KNN algorithm. The Figure 6 shows the scatter plot of the KNN algorithm and the hybrid algorithm against the non pred value and the detection rate parameters. The results show the improvement is obtained when the hybrid algorithm is applied to the dataset.

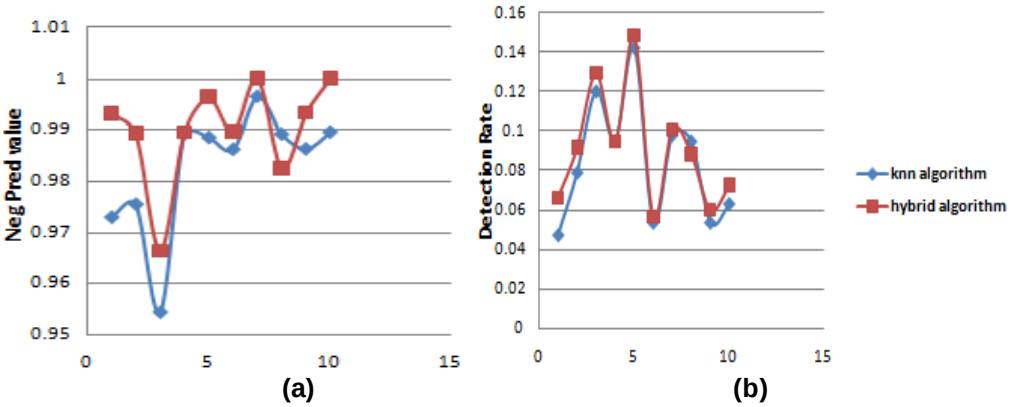

Figure 7. Algorithms Results against Parameters (a) Non Pred Value, (b) Detection Rate

The Figure 7 shows the detection prevalence and balanced accuracy of algorithms. From the Figure it is clearly concluded that the hybrid algorithm yield better results. The classification of documents becomes easy and efficient using the hybrid algorithm.

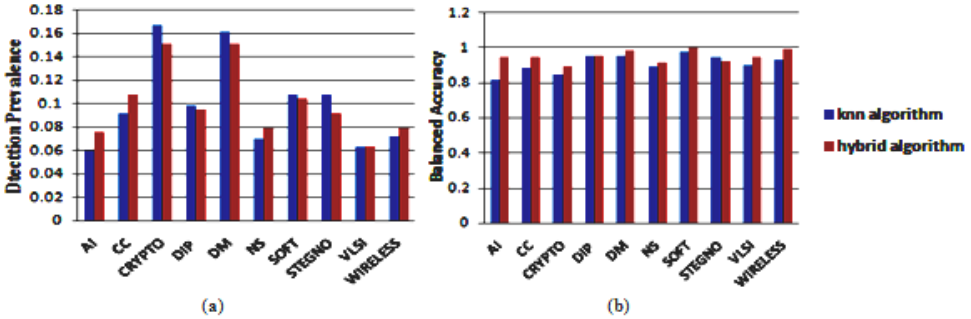

Figure 8. KNN and Hybrid Algorithm Results (a) Detection Prevalence, (b) Balanced Accuracy

We can see that the hybrid algorithm has better results than the KNN algorithm. The overall accuracy of the traditional KNN algorithm and the proposed algorithm called Hybrid algorithm is 84.8101% and 91.139% respectively. As the hybrid algorithm yields the better and efficient results, we can say that it classifies the document more efficiently and effectively than the KNN algorithm.

6. Conclusion

We proposed the hybrid algorithm which is combination of frequent item set extraction and KNN algorithm for the classification of academic research papers. The Top k frequent terms used for initial clustering. We compared the results of the KNN algorithm with the proposed hybrid algorithm. The proposed hybrid algorithm works more effectively and efficiently. As for the future research work, we would like to use predefined ontology to improve the performance and efficiency of the proposed algorithm for the proper classification of the research documents.

References

[1] J. Han, M. Kamber and J. Pei, "Data mining, Southeast Asia edition: Concepts and techniques", Morgan kaufmann, **(2006)**.
[2] V. Gupta and G. S. Lehal, "A survey of text mining techniques and applications", Journal of emerging technologies in web intelligence, vol. 1, no. 1, **(2009)**, pp. 60-76.

[3] W. Le, T. Li, J. Yan and H. Weihong, "A hybrid algorithm for web document clustering based on frequent term sets and k-means", Advances in Web and Network Technologies, and Information Management. Springer Berlin Heidelberg, **(2007)**, pp. 198-203.
[4] K. Nithya, P. C. D. Kalaivaani and R. Thangarajan, "An enhanced data mining model for text classification", Computing, Communication and Applications (ICCCA), International Conference on IEEE, **(2012)**.
[5] E. A. Calvillo, P. Alejandro, M. Jaime, P. Julio and F. T. Jesualdo, "Searching research papers using clustering and text mining", Electronics, Communications and Computing (CONIELECOMP), 2013 International Conference on IEEE, **(2013)**.
[6] N. Arunachalam, E. Sathya, B. S. Hismath and M. M. Uma, "An Ontology Based Text Mining Framework for R&D Project Selection", International Journal of Computer Science & Information Technology (IJCSIT), vol. 5, **(2013)**.
[7] G. B. Jadhav, U. P. Warke, P. S. Kuchekar, J. Bhushan G1, W. Pushkar U2, K. Shivaji P3, K. N. V. N. Kadam, "Searching Research Papers Using Clustering and Text Mining", International Journal of Emerging Technology and Advanced Engineering, vol. 4, Issue 4, **(2014)** April, pp. 788-791.
[8] T. Lalitha and S. Meenakshi, "Text Mining Algorithm Discotex (Dis-Covery from Text Extraction) With Information Extraction", Journal of Theoretical & Applied Information Technology, vol. 64, no. 2, **(2014)**.
[9] R. V. Daniel, and A. K. Shukla, "Improving Text Search Process using Text Document Clustering Approach", pp. 2319-7064.
[10] F. Beil, M. Ester and X. Xu, "Frequent term-based text clustering", Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, **(2002)**.
[11] F. M. Kwale, "A Critical Review of K Means Text Clustering Algorithms", International Journal of Advanced Research in Computer Science, vol. 4, no. 9, **(2013)**.
[12] V. Gupta and G. S. Lehal, "A survey of text mining techniques and applications", Journal of emerging technologies in web intelligence, vol. 1, no. 1, **(2009)**, pp. 60-76.
[13] R. S. Segall, Q. Zhang and M. Cao, "Web-Based Text Mining of Hotel Customer Comments Using SAS Text Miner and Megaputer Polyanalyst", SWDSI, **(2009)**, pp. 141-152.
[14] J. Szymański, "Towards automatic classification of wikipedia content", Intelligent Data Engineering and Automated Learning–IDEAL 2010. Springer Berlin Heidelberg, **(2010)**, pp. 102-109.
[15] C. M. Wijewickrema and R. Gamage, "An ontology based fully automatic document classification system using an existing semi-automatic system", **(2013)**.

Authors

Rajvir Kaur, Persuing, M. Tech CSE Department DAV University, Jalandhar Punjab. She completed B. Tech from SBBSIET Jalandhar, PTU Jalandhar. She was having many projects on Java, PHP and R. Many papers published in international journals.

Er. Nishi, She is working as assistant professor in the Department of Computer Science and Engineering at DAV University, Jalandhar Punjab (INDIA). She has done her M.Tech in Computer Science and Engineering from Punjab Technical University, Jalandhar (INDIA). She has also served NIT Jalandhar, as an assistant professor in CSE Department and was a member of an anti ragging squad. Her research areas include Digital Image Processing and Data Mining. She has various research publications in various international journals and has also presented several papers in various national and international conferences.