International Journal of Computer Science and Informatics

Volume 4 | Issue 2 Article 7

October 2014

INCREASING ACCURACY OF K-NEAREST NEIGHBOR CLASSIFIER FOR TEXT CLASSIFICATION

FALGUNI N. PATEL
Computer Engineering Department, Gujarat Technical University, Sardar Vallabhbhai Patel Institute Of Technology, Vasad, Dist-Anand, Gujarat, falgunin.patel@gmail.com

INCREASING ACCURACY OF K-NEAREST NEIGHBOR CLASSIFIER FOR TEXT CLASSIFICATION

FALGUNI N. PATEL[1], NEHA R. SONI[2]

[1,2]Computer Engineering Department, Gujarat Technical University, Sardar Vallabhbhai Patel Institute Of Technology, Vasad, Dist-Anand, Gujarat

Abstract- k - Nearest Neighbor Rule is a well-known technique for text classification. The reason behind this is its simplicity, effectiveness, easily modifiable. In this paper, we briefly discuss text classification, k-NN algorithm and analyse the sensitivity problem of k value. To overcome this problem, we introduced inverse cosine distance weighted voting function for text classification. Therefore, Accuracy of text classification is increased even if any large value for k is chosen, as compared to simple k Nearest Neighbor classifier. The proposed weighted function is proved as more effective when any application has large text dataset with some dominating categories, using experimental results.

Keywords- *Text Classification, k- Nearest Neighbor Weighted voting, Dominating class.*

1.0 INTRODUCTION

Text Classification is a method for classifying text documents into pre-defined categories. Therefore, it is also known as supervised learning technique. Text classification has many applications like Business Feedbacks Classification, Cinema Blogs Classification, Mobile SMS Classification, Paper Filtering, product opinion classification, e-mail filtering, research paper classification, finding answers to similar questions that have been answered before say in some legal court case, classifying news by subject or newsgroup etc[1]. Text classification makes use of many classifiers like Rocchio's algorithm, Naïve Bayes, Decision rule, Support vector machine and neural network etc. [2][3]. Among all, k-NN classifier is proved better than others for large textual dataset[4]. Though better than other, it suffers from k value sensitivity effect for large dataset. It means that if k value is selected as too small then more sparseness, ambiguous and unlabelled data belong to local estimation and misclassification is done. On the other way, if k valueis taken as too large then outlier data from other dominating class belong to category assignment estimation and result performance goes down. Here, dominating class in text dataset are those classes or categories under which more documents are classified. In this paper, we presented simple k-NN method and k value selection problem. To address this problem we had suggested a new inverse cosine distance weight function and shown the improvement in accuracy through experiments on Reuter-21578 and 20-Newspapers text dataset.

II. TEXT CLASSIFICATION

Text classification is a supervised learning algorithm whose approach is to classify a text documents into pre-defined categories [5]. Classification has basic four types as per applications – Binary Hard, Multi-class single label, Multi-class Multi label hard and multi-class multi label soft classification [6].

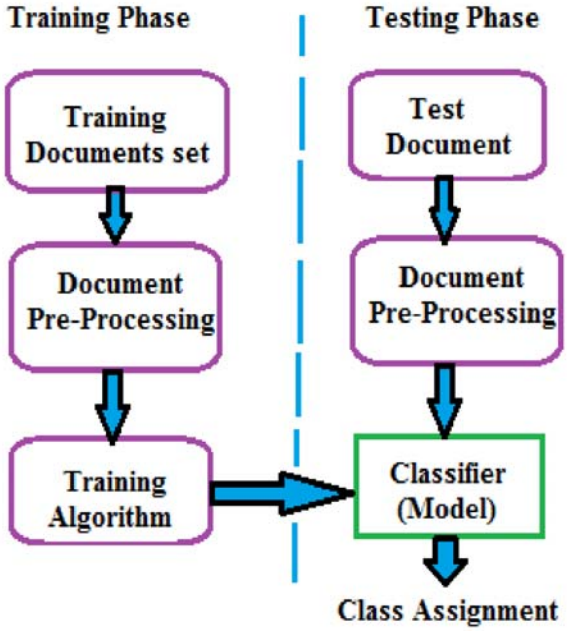

Figure -1 General model of text classification

Multi-class means multi categories are allowed like sport, jokes, and business in news data. Multi –label means document belong to more than one category like any documents belong to business and sport categories. Hard classification means document has Boolean membership into classes and soft means fuzzy membership into all classes. In this paper, our approach is regarding multi-class single label classification. For classification, general model is represented in figure -1. There are two phases - training and testing phase. In training phase, training documents that are pre-labelled are used to build a model using its feature vectors which are generated

by pre-processing. Actual class assignment process performed in test phase. Whenever a new test document come for classification, then feature vector of test document is applied in model and class label assigned. In this paper, we had selected tf-idf feature vector text representation because it works better in k-NN classifier for text.

A. Text Representation

For classification, documents are needed to represent in an acceptable form of classifier. Therefore, Documents are going to convert into simple acceptable compact format like binary frequency, term frequency, Inverse frequency and tf-idf etc. [22]. format. Binary frequency means word is belong to document or not (0 or 1). Term frequency is includes term occurrences in document. Inverse document frequency covers document frequency in which the term is repeated. Combination of this two, tf-idf works better for k-NN classifier for text data [7][8][9]. Next, Dimension reduction method is applied. Terms which are more frequently used in all documents and less used in particular document are not important for classification. They all are removed. And this final feature vector used for classification [10][11].

B. k – Nearest Neighbor Classifier

k-NN classifier is one of the most conceptually simple technique in all text classification[12][13]. To determine class of test document following steps of k-NN are applied. For k-NN, we assume below terms[14],

i) T={x1,x2,…,xN} is training set, having class-labelled point (xi,cj).

ii) Training vectors xi are vectors in the m-dimensional feature space, i=1,2…N, and Classes cj, j=1, 2…Nc are their corresponding class labels where N> Nc.

iii) Test document is X′ and its class C′ is unknown.

Steps:

(1) Calculate distance or Cosine Similarity between test point x′ and all training points T set.it is d(x′, xi) .

(2) Arrange them in sorted order.

(3) Select top k Training set T′, = {x1□□, x2□□…xk□□}.

(4) Calculate sum for documents have same category as per majority voting. Whose category voting is high that category assign to test data point.

As discussed in steps, this simple k-NN is also popular as majority voting k-NN. For class assignment decision, voting of top k documents are used which are most near to test documents. The category, which has highest voting, then that category label is assigned to test document. This simple k-NN classifier has problem of selection of k value for large text dataset having dominating classes [15]. Dominating class means class having large number of documents compared to other classes with less number of documents. If k value is selected too large, outliers from dominating class will be included to estimation and accuracy of k-NN is degraded. On the other hand, if k value is chosen too small then ambiguous, noisy or mislabelled documents comes into local estimation boundary for class estimation of test document. Large text dataset need to cover more training documents for estimation of category. That means in such cases large value for k is required to choose and simple k nearest neighbor leads to misclassification.

The samet is explained with example below. As seen in figure -2, diamond red color shape represent class 1 and circle green color shape represent class 2. Test document in circle shape with blue color is need to assigned class label. If k=2 is selected then as per majority voting, class of test document is 2 that is right answer but if we increase k value and select k=6 then answer is class 1 that is misclassification. That shows that dominating class 1 is degrading accuracy of k-NN classifier for higher k value selection in large text data. We can improve accuracy of k-NN classifier by applying weight.

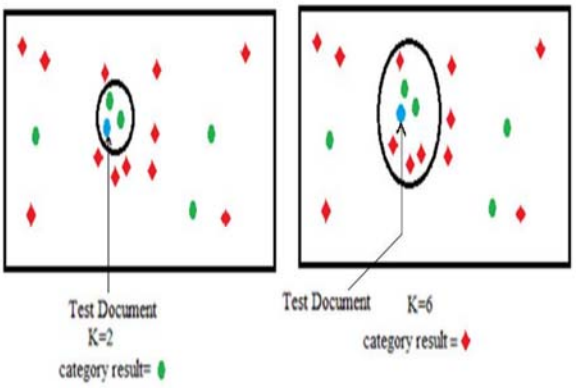

Figure – 2 Selection of k value and sensitivity problem

C. Proposed Scheme

In large text data having dominating classes, large k value selection will be major problem. Therefore, we proposed function which is inverse of cosine distance. Cosine distance is opposite of cosine similarity so cosine similarity is one minus cosine distance and its range between 0 to1. Cosine distance of document is low means highest similarity document so high weight it has and same as less similar documents get low weight value. Weight value ranges between 0 to1.

After calculating weight for each k documents, as per our proposed scheme, weighted sum of documents which belong to same category is calculated.. Category with highest weight that category label assigned to test document. Therefore, misclassification ratio is down in case of k value is increasing. With our proposed scheme- k-NN

algorithm steps for text classification shown below:
(1)Calculate Cosine Similarity between test document x′ and all training documents in T set. it is S(x′, xi).
(2) Arrange them in Descending sorted order.
(3)Select top k Training set T′, = {$x1^{NN}$, $x2^{NN}$…xk^{NN}}.
(4) Calculate weight for each document. Give a high weight to most near document and less weight to far documents. Calculate sum of weight that have same class. Weight function Wi = $1/$(1- CSi), where i =1,2,3…k top k - documents. CS is a Cosine Similarity between test document and training documents [16]. Here, 1- cosine similarity is cosine distance [17].
(5) Assign a category to test document that have high weighted sum.

III. EVALUATION AND RESULT

As final step of classification process, result evaluation performed with different measures. We first discuss basic knowledge about classification measures and then after shown charts of actual result which performed on Reuters -21578 and 20 – Newspaper group text dataset.

A. Evaluation
The evaluation of text classification concentrated on two issues – computationally efficient and classification effectiveness. Computationally efficiency is not a major point because of enough processing capability and memory is available nowadays. For effectiveness, our proposed scheme is proved better than simple k-NN for text data. The measures used for Text classification are precision, recall, F1 Measure and accuracy. This all measure are based or calculated from confusion matrix as shown in figure-3. Confusion matrix consists of four parts : TP, true positive means documents are actually related to current class and by performing algorithm resulted or belong to

	Actual (Expected Class)	
Predicted Class (Observa -tion)	**tp** (Correct Result)	**fp** (Unexpected Result)
	fn (Missing Result)	**tn** (Correct ansence of Result)

Figure – 3 Confusion matrix

that class. FP, false positive means documents which are not related or other class resultant into current class give unexpected result. FN, false negative means related documents resultant into other class or missing result and TN, true negative means unrelated documents not resultant into other class. Here, Actual class means related/expected class before execution and predicted class after execution of algorithm.

Precision is the probability of relevant documents is retrieved from all retrieved documents. Recall is the probability of relevant documents is retrieved from all relevant documents. F_β Measure is harmonic combination of Precision and Recall. If β =1 then F_1 measure where precision and recall are evenly weighted. If β=0.5 then $F_{0.5}$ measure where precision more weighted than recall. If β=2 then F_2 measure where recall more weighted than precision. Accuracy is commonly used as a measure for categorization techniques. Accuracy values, however, are much less reluctant to variations in the number of correct decisions than precision and recall [18].

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

$$A_i = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}$$

B. Experimental Comparison
This section includes results, which show that our proposed method with weight function gives better performance than simple k-NN algorithm. The results are based on following two standard benchmark large text dataset.

The Reuter-21578 newspaper dataset consist of 21578 documents and 65 categories [19]. For our algorithm evaluation, we selected top 5 categories: Earn, Acq, Crude, Trade and Money. The dataset is pre-processed as explained above. For the experiment, we selected top 1400 attributes among 61188 attributes after dimension reduction technique. Some sample dataset are randomly chosen for testing

like 500,700,1000,1250,1413 training documents. As all results are not possible to present, only with 700 training documents results are shown here. For testing, from all categories two documents means total 10 documents are selected and result are evaluated. Selected top categories has following

number of documents as shown in figure-4 and among them different documents selected as a sampletraining documents. The resultant charts of 700 sample text data are shown in figure -7, 8, 9 with Precision, Recall, F-measure measures.

The 20 Newsgroups dataset include 20 categories, 18774 documents and 61188 attributes [20]. We selected top 5 categories: Atheism, Forsale, Motorcycles, Electronics, and Space etc. The dataset is converted in tf-idf format for further processing. For experiment only top 3000 attributes are used with 10 documents for test and 1055 for training. Total documents of top categories of 20 newsgroups are listed in figure-5. The resultant charts are shown in figure -11, 12, 13 of Precision, Recall and F-measure measures.

Category	Training Documents	Testing Documents	Total Documents
Earn	2673	1040	3713
Acq	1435	620	2055
Crude	223	98	321
Trade	225	73	298
Money	176	69	245

Figure – 4 Top 5 category of Reuter -21578 with Document list

Category	Training Documents	Testing Documents	Total Documents
Atheism	480	318	798
Forsale	582	382	964
Motorcycles	596	397	993
Electronics	591	393	984
Space	593	392	985

Figure – 5 Top 5 category of 20 Newspaper group with Document list

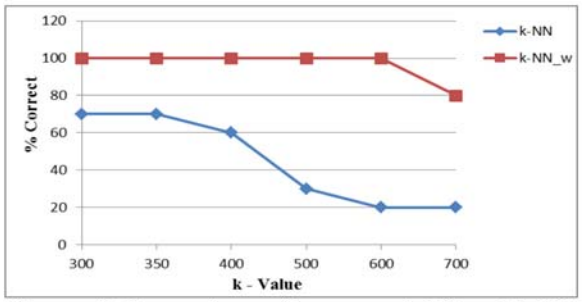

Figure - 6 Comparisons of accuracy k-NN and k-NN weight algorithm for Reuter -21578 news text dataset.

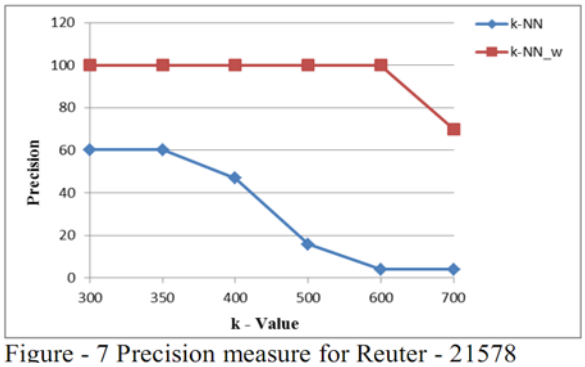

Figure - 7 Precision measure for Reuter - 21578

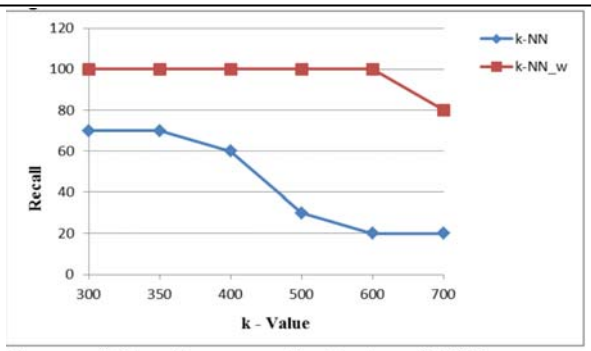

Figure - 8 Recall measure for Reuter - 21578

As we see in chart more large k value choose, Accuracy of simple k-NN algorithm is goes down and in compare to that our proposed work give better accuracy. In general, as per [21], for category estimation required minimum half of the information or training set we choose.

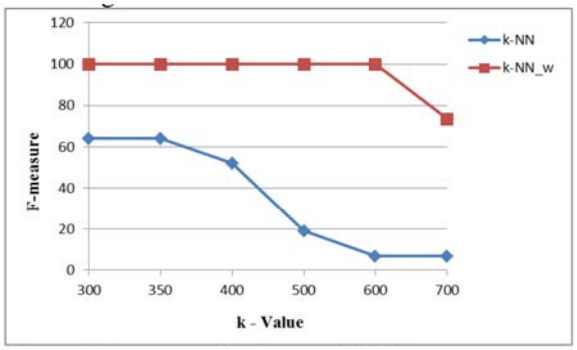

Figure-9 F-measure for Reuter -21578

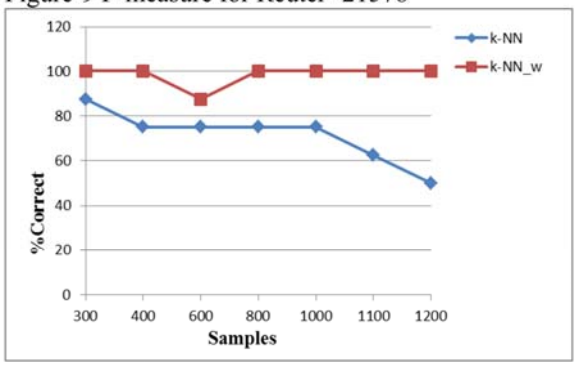

Figure -10 Comparison of Accuracy of k-NN and k-NN with weight algorithm with different sample dataset with fix k=10 value

We are also tested randomly on sample of 300,400,600,800,1000,1100,1200 etc. documents and shown in figure-10 our proposed work give better accuracy than simple k-NN. Here, Following Chart show Precision, Recall and F- measure with 1055 training and 10 text documents of 5 categories of 20-Newsgroup dataset with 3000 attributes. Across all our experimental above, it can be concluded that using our inverse cosine distance weighted function with k-NN not only improves the Classification performance but also overcome sensitivity problem of neighbourhood size k in many practical situations.

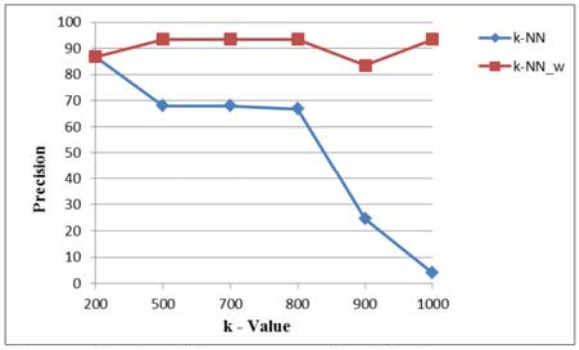

Figure -11 Precision measure for 20-Newsgroup text dataset

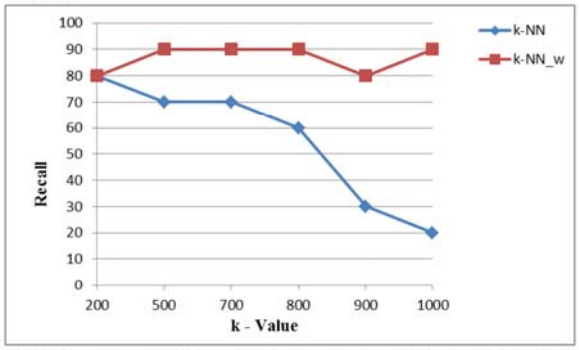

Figure -12 Recall measure for 20-Newsgroup text dataset

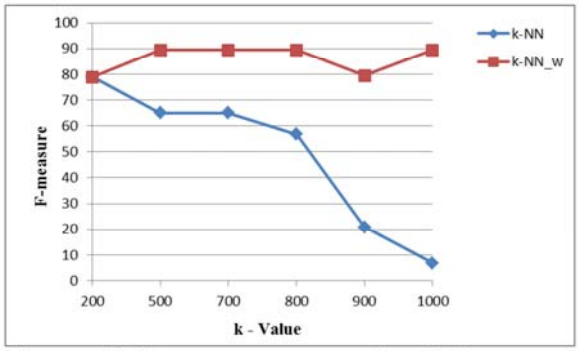

Figure -13 F-measure measure for 20-Newsgroup text dataset

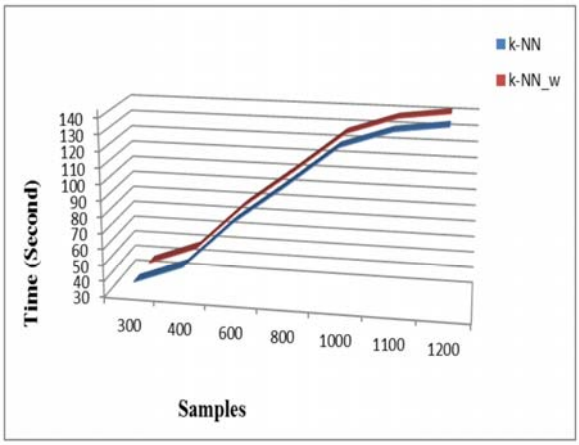

Figure -14 Comparison of Time complexity of k-NN and k-NN with weight algorithm with different sample of Reuter-21578 dataset with fix k=10 value

As shown in figure-14 time line chart show that as weight calculation add a little time in simple k-NN result that is negligible. K-NN take O(NL) time where N is number of training documents set and L is number of test documents set for classification. For k-NN_w algorithm time complexity is O(NL)+O(Kd), where Kd is top k documents whose weight is calculated.

IV. CONCLUSION

In this paper, we proposed a new inverse cosine distance weight function for k-NN classifier for text dataset. This weight function improves accuracy of algorithm for large text dataset which have some dominating class and thus leads to misclassification of new upcoming test documents. The experimental results of Reuter – 21578 and 20-Newsgroups test dataset shows that k-NN classifier with a proposed weight function yields much better accurate result than simple k-NN.

REFERENCES

[1] Nidhi and Vishal Gupta, "Recent Trends In Text Classification Techniques", International Journal of Computer Application, ISSN: 0975-8887, Volume 35, Issue no 6, December 2011.

[2] Aigars Mahinovs, Ashutosh Tiwari, "Text Classification Method Review" Decision Engineering Report Series, Cranfield University, ISBN 978-1-86194-128-2, April 2007.

[3] Vandana Korde, C Namrata Mahender, "Text Classification And Classifiers: A Survey", International Journal of Artificial Intelligence & Applications (IJAIA), DOI: 10.5121/ijaia.2012.3208, Vol.3, No.2, March 2012.

[4] Aditya Chainulgu Karamcheti, "A Comparative Study on Text Categorization", Thesis Report, University Libraries, University of Nevada, Las Vegas, May 2010. [5] Mita K. Dalal, Mukesh A. Zaveri, "Automatic Text Classification: A Technical Review", International Journal of Computer Applications (0975 – 8887), Volume 28– No.2, August 2011.

[6] Mohammed Abdul Wajeed (PhD), Dr.T.Adilakshmi, "Text Classification Using Machine Learning", Journal of Theoretical and Applied Information Technology, 2005 - 2009 JATIT.

[7] P. Sowmya Lakshmi, V. Sushma, T. Manasa, "Different Similarity Measures for Text Classification Using k-NN", IOSR-Journal of Computer Engineering, ISSN: 2278-0661, ISBN: 2278-8727 Volume 5, Issue 6 (Sep-Oct, 2012).

[8] J.Sreemathy, P.S. Balamurugan, "An Efficient Text Classification Using k-NN and Naïve Bayesian", International Journal on Computer Science and Engineering (IJCSE), ISSN: 0975-3397 Vol. 4 No. 03 March 2012.

[9] Pascal Soucy, Guy W. Mineau, "A Simple k-NN Algorithm for Text Categorization", IEEE 0-7695-1119-8/01, 2001.

[10] Wa'el Musa Hadi, Fadi Thabtah, Hussein Abdel jaber, "A Comparative Study using Vector Space Model with k-Nearest Neighbor on Text Categorization Data", Proceedings of the World Congress on Engineering, ISBN:978-988-98671-5-7 Volume I, WCE 2007, July 2 - 4, 2007, London, U.K.

[11] V. Srividhya, R. Anitha, "Evaluating Pre-processing Techniques in Text Categorization", International Journal of

Computer Science and Application Issue 2010, ISSN 0974-0767.

[12] S. Dhanabal, Dr. S. Chandramathi,"A Review of Various k-nearest Neighbor Query Processing Techniques", International Journal of Computer Application (0975-8887), Volume 31-No.7, October 2011.

[13] Nitin Bhatia, Vandana,"Survey of Nearest Neighbor Techniques", (IJCSIS) International Journal of Computer Science and Information Security, Volume 8, No. 2, 2010. http://sites.google.com/site/ijcsis/, ISSN 1947-5500.

[14] Gulen Toker, Oznur Kirmemis, "Text Categorization Using k- Nearest Neighbor Classification", Computer engineering department, Middle East Technical University.

[15] Muhammed Miah, "Improved k-NN Algorithm for Text Classification", Department of Computer Science and Engineering, University of Texas at Arlington, TX, USA.

[16] Sahibsingh A. Dudani, "The Distance-Weighted k-Nearest-Neighbor Rule", IEEE Transactions on Systems, Man, and Cybernetics, April 1976.

[17] Ashok N. Srivastava, Mehran Sahami, "Text Mining: Classification, Clustering and Applications", ISBN: 978-1-4200-5940-3, CRC Press, Taylor and Francis Group, LLC, USA, 2009.

[18] M. Ikonomakis, S. Kotsiantis, V. Tampakas, "Text Classification Using Machine Learning Techniques", Wseas Transactions on Computers, Issue 8, Volume 4, August 2005, pp. 966-974.

[19] http://kdd.ics.uci.edu/databases/reuters21578

[20] http://people.csail.mit.edu/jrennie/20Newsgroups

[21] T. M. Cover, P. E. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory, 13 (1), 1967, 21-27.

[22] Falguni N. Patel, Neha R. Soni, "Text mining:A brief survey", International Journal of Advanced Computer Research, 2(6), 243 - 248.