Efficient Hybrid Machine Learning Algorithm for text Classification

Dr. M. Durairaj[1]
Assistant Professor
School of Comp. Sci., Engg, & Applications,
Bharathidasan University,
Tiruchirappalli – 620 023

A. Alagu Karthikeyan[2]
Research Scholar
School of Comp. Sci., Engg, & Applications,
Bharathidasan University,
Tiruchirappalli – 620 023
E-mail - alagukarthikeyan.1988@gmail.com

Abstract:- Text Mining and Text Classification are the most important and challenging task. Deriving high quality and relevant information form text is Text Mining and categorizing the text documents is done using the Text Classification. The real challenge in these areas is to address the problems like handling large text corpora, similarity of words in text documents, and association of text documents with a subset of class categories. The feature extraction and classification of such text documents require an efficient machine learning algorithm which performs automatic text classification. The major drawback encountered in text classification and retrieval is determining whether a text is pertinent to the query. This work focuses on text classification by using the data mining techniques. A hybrid algorithm is proposed for classifying the text. The proposed algorithm combines the concepts of KNN, SVM and NB. The results obtained support the proposed hybrid algorithm in text classification.

Keywords: *Big Data, Text Mining, Text Classification, Text Categorization, Tokenization, Feature Selection, Stemming, Classifiers, Data Mining.*

*****

I. INTRODUCTION

Text mining is the process of seeking or extracting the useful information from the textual data. It is an exciting research area as it tries to discover knowledge from unstructured texts. It is also known as Text Data Mining (TDM) and knowledge Discovery in Textual Databases (KDT). KDT plays an increasingly significant role in emerging applications, such as Text Understanding. Text mining process is same as data mining, except, the data mining tools are designed to handle structured data whereas text mining can able to handle unstructured or semi-structured data sets such as emails HTML files and full text documents etc. [1]. Text Mining is used for finding the new, previously unidentified information from different written resources.

Structured data is data that resides in a fixed field within a record or file. This data is contained in relational database and spreadsheets. The unstructured data usually refers to information that does not reside in a traditional row-column database and it is the opposite of structured data. Semi Structured data is the data that is neither raw data, nor typed data in a conventional database system. Text mining is a new area of computer science research that tries to solve the issues that occur in the area of data mining, machine learning, information extraction, natural language processing, information retrieval, knowledge management and classification. Figure 1 gives the overview of text mining process.

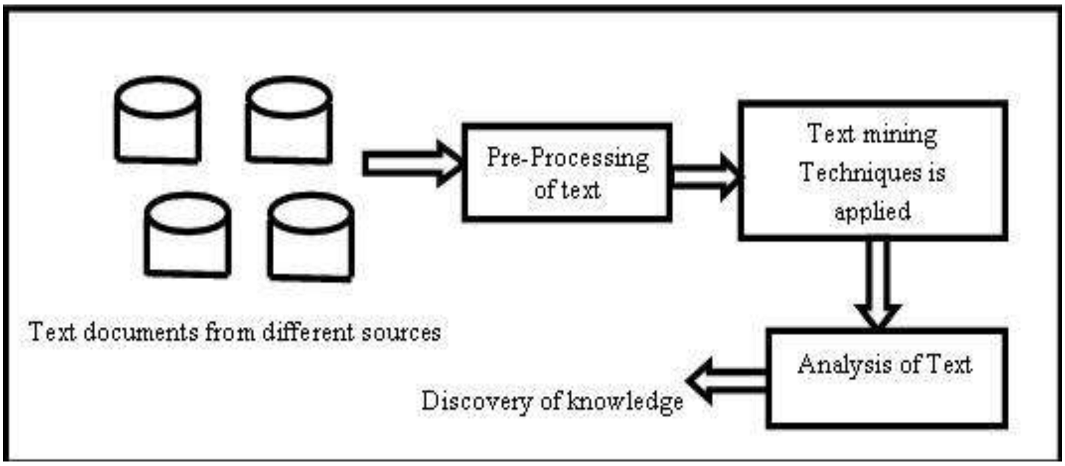
Pre-Processing of text
Text mining Techniques is applied
Text documents from different sources
Analysis of Text
Discovery of knowledge

Figure 1: Overview of the Text Mining process

II. RELATED WORK

For context, we present many research works which are focused on text domains. In [2], the performance of Neural Network based text classification was enhanced by assigning the probabilities obtained from Naïve Bayesian method as initial weights. Fragoudis et al. [3] combined Feature and Instance Selection for classification of text for enhancement. Their method consists of selecting features sequentially that have high precision in predicting the target class. In [4] Naïve Bayesian method was used as a pre-processor for dimensionality reduction followed by the SVM method for text classification. Vandana Korde et al [5] discussed that the text mining studies are obtained more availability of the increasing number of the electronic documents from a variety of sources. Guan and Zhou proposed a training-corpus pruning based approach to enhance the process speed [6]. By using this approach, the size of training corpus can be reduced significantly while classification performance can be kept at a level close to that of without training documents pruning according to their experiments. William B. Cavnar et al [7] says that the N-gram frequency method provides an inexpensive and highly effective way of classifying documents. The authors of [8][9] try to propose an approach to build semi-automatically high-quality training corpuses for better classification performance.

III. BACKGROUND

A supervised machine learner builds a model based on a training set of labelled corpus and employs this model to envisage the classification of unlabelled test texts. The machine learner will pre-process— including, stemming, term weighting, and tokenization which will be completed. The widely used machine learning algorithms, employed classification are k-Nearest Neighbour, Support Vector Machine classifiers and Naive Bayes.

(i) *K-Nearest Neighbor:* The k-nearest neighbor algorithm (k-NN) is used to test the degree of similarity between documents and k training data. This method is an instant-based learning algorithm that categorized items based on closest feature space in the training set [10]. The key element of this method is the availability of a similarity measure for identifying neighbors of a particular document. This method is non parametric, effective and easy for implementation.

(ii) *Naive Bayes Algorithm:* Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes‟ Theorem with strong independence assumptions. The more expressive term for the underlying probability model would be independent feature model. This independence hypothesis of features make the features order is irrelevant and as a result that the presence of one feature does not affect other features in classification tasks which makes the computation of Bayesian classification approach more efficient. Naive Bayes classifiers can be trained powerfully by requiring a small amount of training data to estimate the parameters necessary for classification.

(iii) *Support Vector Machine:* SVMs are a generally applicable tool for machine learning. Suppose we are given with training examples xi, and the target values yi{-1,1}. SVM searches for a separating hyperplane, which separates positive and negative examples from each other with maximal margin, in other words, the distance of the decision surface and the closest example is maximal [11]. The equation of a hyperplane is:

$$w^T x + b = 0$$

Text documents are not agreeable to being interpreted by a classifier or by a classifier-building algorithm. Figure 2 shows the steps involved in machine learning process.

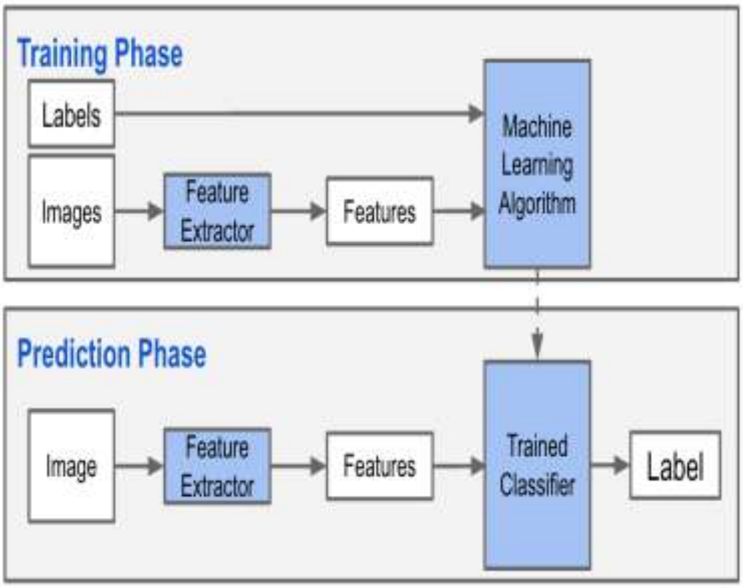

Figure 2: Steps in machine learning classification

The documents can be classified into unsupervised, supervised and semi-supervised methods which are predefined categorizes. The text mining is vital to the process of gaining insights. In the context of machine learning approach, the important resource of pre classified documents. The machine learning approach is easier. In fact, it is easier to manually classify a set of documents than to build and tune a set of rules, since it is easier to characterize a concept extensionally.

IV. PROPOSED APPROACH

The efficient strategy for text classification involves pre-processing of documents, after then feature extraction of labelled corpus (machine learning classifier for training data), next comes the model selection and classifier. These steps are presented in Figure 3.

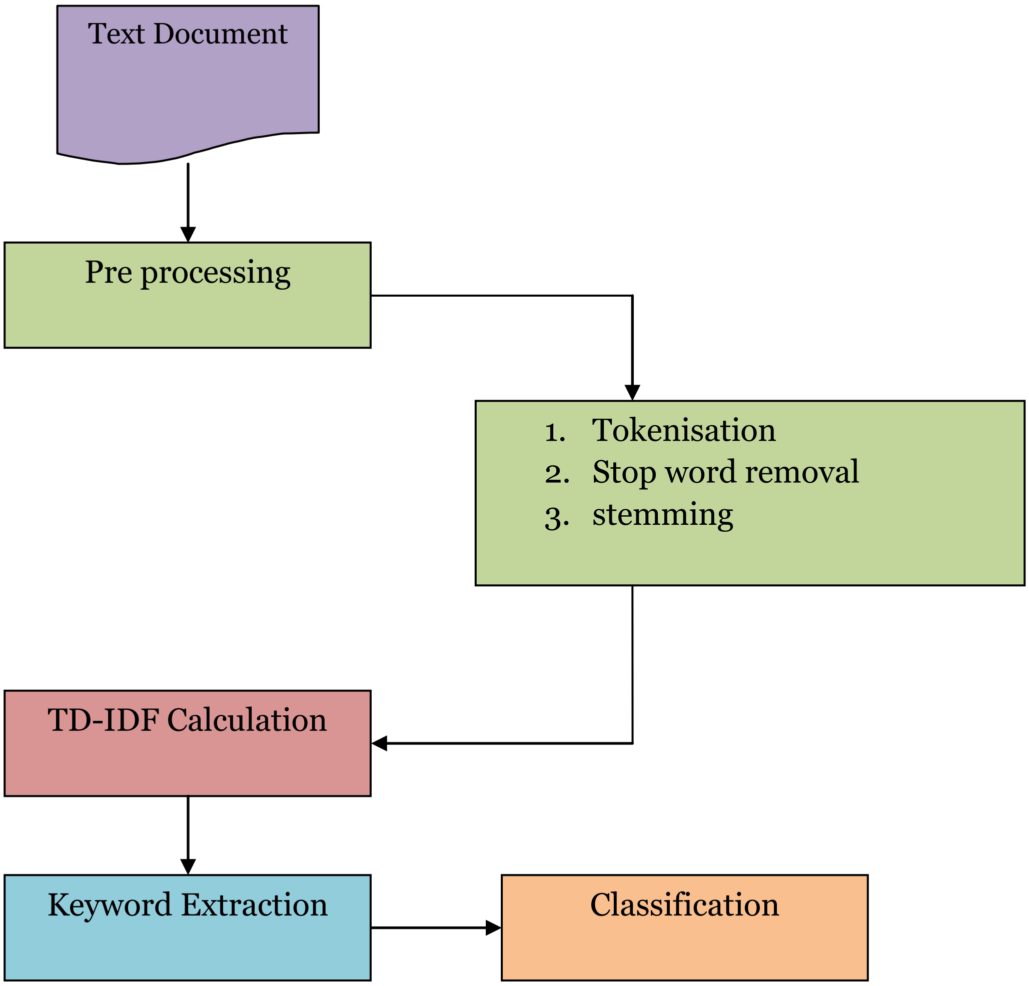

Figure 3: Logical structure of the Proposed Approach

A. PREPROCESSING

Tokenization

Tokenization is the first pre-processing stride of any natural language processing and corpus generation [12]. It is the process of replacing the meaningful sentence in to individual words with space as the delimiter and it retain all the valuable information's. Each individual word is known as tokens. These tokens are the key elements of the Natural Language Processing. In our experiment, tokenization is one of the pre-processing steps of corpus generation

Stop Words Elimination

Stop words are a part of natural language that does not have so much meaning in a retrieval system. The reason that stop-words should be removed from a text is that they make the text look heavier and less important for analysts. Removing stop words reduces the dimensionality of term space. The most common words are in text documents are prepositions, articles, and pro-nouns etc that does not provide the meaning of the documents. These words are treated as stop words. Example for stop words: the, in, a, an, with, etc. Stop words are eliminated from documents because those words are not considered as keywords in text mining applications.

Stemming

Stemming techniques are used to find out the root/stem of a word. Stemming converts words to their stems which incorporates a great deal of language-dependent linguistic knowledge. For example, the words, connection, connects, connected, connecting all can be stemmed to the word 'connect'

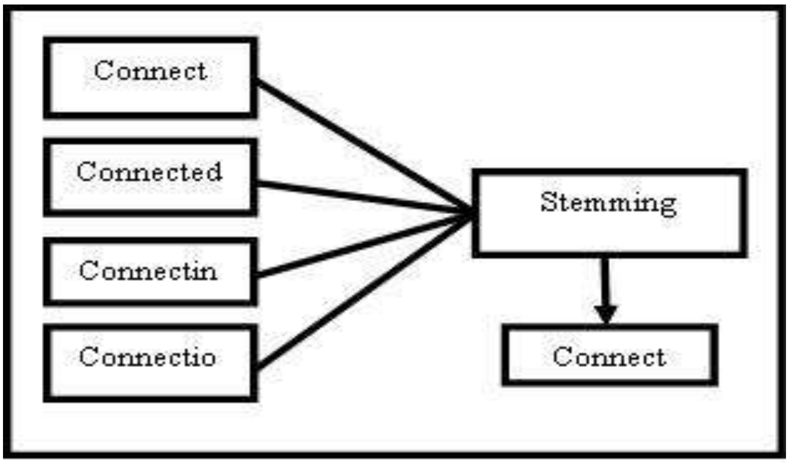

Figure 4 Stemming Process

In stemming, translation of morphological forms of a word to its stem is done assuming each one is semantically related. There are two points are considered while using a stemmer:

- Words that do not have the same meaning should be kept separate
- Morphological forms of a word are assumed to have the same base meaning and hence it should be mapped to the same stem

These two rules are good and sufficient in text mining or language processing applications. Stemming is usually considered as a recall-enhancing device. For languages with relatively simple morphology, the power of stemming is less than for those with a more complex morphology. Most of the stemming experiments done so far are in English and other west European languages.

B. FEATURE SELECTION

Feature selection is a process commonly used in Machine Learning field to reduce the dimensionality of the feature space. The subset of the features available in the data is keywords are selected out. The selected features receive the highest scores according to a function that measures the importance of the feature for text classification task [13]. The functions used to measure the importance are quite significant. Simple and effective function is the term frequency of a term that is only the terms that occur in the highest numbers in a document are retained.

Term Frequency-Inverse Document Frequency

Term Frequency–Inverse Document Frequency (tf-idf) is a numerical statistic which reveals that a word is how important to a document in a collection. Tf-idf is often used as a weighting factor in information retrieval and text mining. The value of tf-idf increases proportionally to the number of times a word appears in the document, but is counteracting by the frequency of the word in the corpus [14] . This can helps to control the fact that some words are generally more common than others. Tf–idf can be successfully used for stop-words filtering in various subject fields including text summarization and classification. Tf–idf is the product of two statistics which are term frequency and inverse document frequency. To further distinguish them, the number of times each term occurs in each document is counted and sums them all together.

Term Frequency- Term Frequency (TF) is defined as number of times a term occurs in a document.

$$tf\,(t,d)=0.5+\frac{0.5*\,f\,(t,d)}{\max imum\ occurence\ of\ words} \qquad ....(1)$$

Inverse Document Frequency- An Inverse Document Frequency (IDF) is a statistical weight used for measuring the importance of a term in a text document collection. IDF feature is incorporated which reduces the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely.

$$idf(t,d) = \log \frac{|D|}{no.of\ documents\,term\ t\,appears} \qquad ....(2)$$

Then Term Frequency - Inverse document frequency [TF-IDF] is calculated for each word using the formula,

$$tfidf(t,d,D) = tf(t,d) * idf(t,d) \qquad ....(3)$$

In this equation (1) and (2) f denotes the frequency of the occurrence of term *t* in document *d.* In equation (3) TFIDF is calculated for each terms in the document by using Term Frequency (tf $_{t,d}$) and Inverse Document Frequency (idf $_{t,d}$).

C. CLASSIFICATION

Text classification is one of the main applications of machine learning. The task is to assign unlabeled new text document to predefined category. The processing of text classification involves two main problems, first problem is the extraction of feature terms that become effective keywords in the training phase and then the second is actual classification of the document using these feature terms in the test phase. Before classifying documents, pre processing has done. In pre processing stop words are removed and the words are stemmed. Then the term frequency is calculated for each term in a document and also TF-IDF is calculated. The document is classified by using the proposed hybrid algorithm which combines the features of KNN, SVM and NB. The pseudocode and working of the proposed hybrid algorithm is given below.

The distance of the training data and unknown sample is calculated. Then the linear function is build using the SVM classifier. The set I is computed which has the indices for the k small distances. The majority label is then returned and the vocabulary is extracted from I. Then Naïve Bayes is applied to calculate the *$P(X_j)$ and $P(X_k|X_i)$.* The output will be the classified document.

```
Classify (X,Y,x)
        X-> training data X      Y->class label of X    x-> unknown sample

for (i=1 to m) do
      Compute distance d(X,x)
End for
      Build Linear Function  $f(x) = \langle w.x \rangle + b$
      Compute set I containing indices for the k smallest distance $d(X_i,x)$
              $y_i = \begin{cases} 1 \ \ if \ < w.x_i > + b \geq 0 \\ -1 \ \ if < w.x_i > + b < 0 \end{cases}$
      Return majority label for {$y_i$ where i $\varepsilon$ I}
      From I extract vocabulary
     Calculate $P(X_j)$ and $P(X_k|X_i)$

For each $X_i$ in I do
            $P(X_i) = \frac{|docs_j|}{|total\ document|}$
                              $docs_j = subset\ of\ documents\ for\ which\ the\,t\arg et\ classify$
End for
For Each word $X_K$ in vocabulary
         $P(X_k \mid X_j) = \frac{X_k + \alpha}{X_j + \alpha |vocabulary|}$        $\alpha = tuning\,/\,smoothing\ parameter$

End for
```

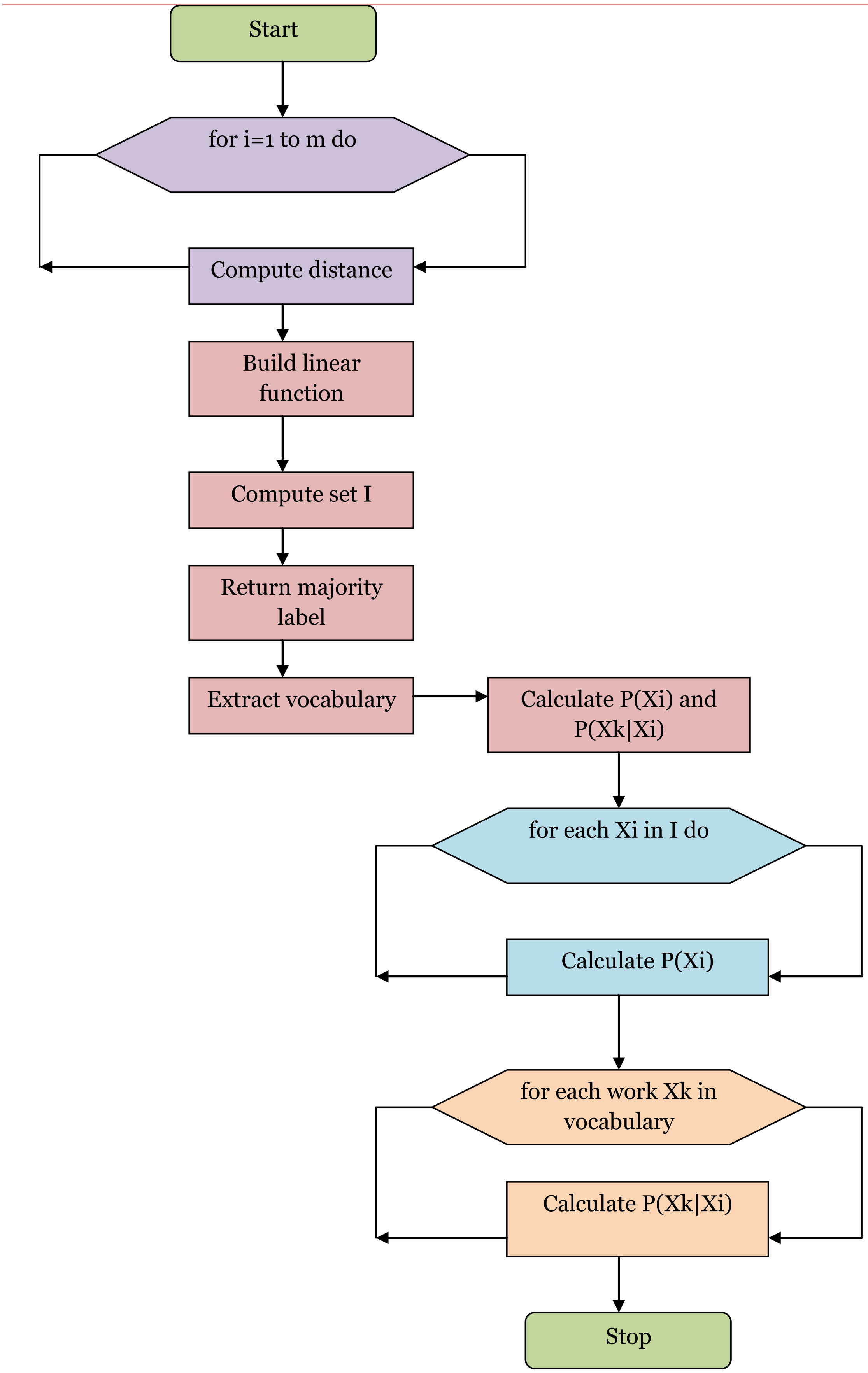

Figure 5: Logical overview of the proposed hybrid algorithm

V. EXPERIMENTAL RESULTS:

Precision and Recall: Precision and Recall are the parameters used for evaluating the performance of text mining.

$$Precision = \frac{True\,Positive}{True\,Positive + False\,Positive} \text{ ....(4)}$$

$$Recall = \frac{True\,Positive}{True\,Positive + False\,Negative} \text{ ....(5)}$$

F-Measure: F-Measure is the balance between Precision and Recall.

$$F - measure = \frac{2 * recall * precision}{precision + recall} \text{ ....(6)}$$

Accuracy: Accuracy is measurement for classification performance.

$$Accuracy = \frac{True\,Positive + True\,Negative}{True\,Positive + False\,Positive + True\,Negative + False\,Negative} \text{ ......(7)}$$

The proposed approach has classified the document with more classification accuracy. Table 1 and Figure 6 compare the performance of the existing algorithm with proposed algorithm in text classification. The results proves that the proposed algorithm out performs the existing in terms of classification accuracy

Table 1: Classification Accuracy

No.	Techniques used	Accuracy (%)
1.	SVM	70.01
2.	NB	75.51
3.	K-NN	75.02
4.	PROPOSED APPROACH	83.25

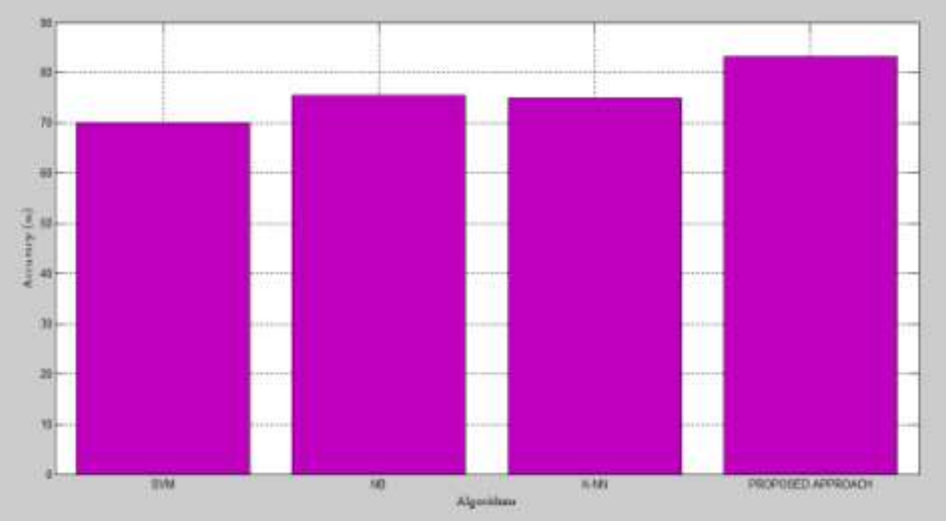

Figure 6: Comparison of Classification Accuracy

The algorithms are also compared in terms of RMSE. The proposed algorithm shows less value in RMSE when compared with the existing work. Table 2 and Figure 7 depicts the results obtained.

Table 2: RMSE

No.	Techniques used	RMSE
1.	SVM	0.54
2.	NB	0.41
3.	K-NN	0.54
4.	PROPOSED APPROACH	0.36

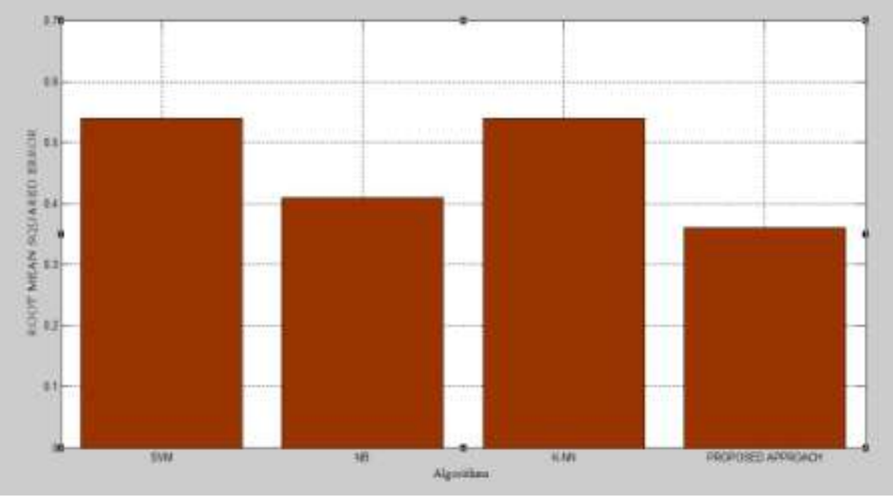

Figure 7: Comparison of RMSE

The metrics like Precision, Recall and F-Measure is also calculated to validate the performance of the algorithms. The promising results obtained with proposed algorithm are shown in Table 3 and Figure 8.

Table 3: Comparison of Algorithm in terms of metrics

No.	Techniques used	Precision	Recall	F-Measure
1.	SVM	0.71	0.71	0.71
2.	NB	0.50	0.71	0.59
3.	K-NN	0.51	0.77	0.61
4.	PROPOSED APPROACH	0.43	0.63	0.51

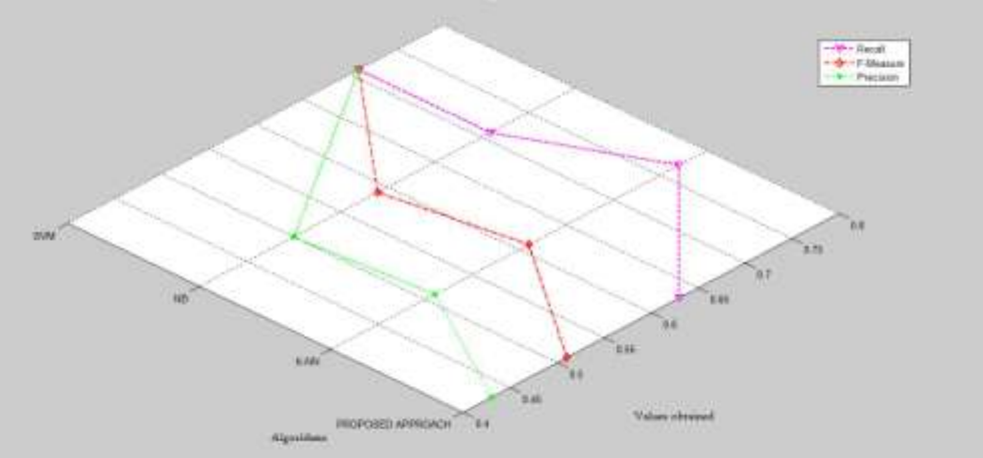

Figure 8: Comparison of Algorithm in terms of metrics

VI. CONCLUSION AND FUTURE WORK

We have presented a hybrid approach for text classification that combines a machine learning algorithm. Classifications of text are the key to the ability of processing, retrieval of queries and efficient management of information. Text classifications are entrained by humans who read the texts and gain insights into the content. The major drawback encountered in text classification and retrieval is determining whether a text is pertinent to the query. The proposed hybrid approach achieves an accuracy of 83.25%. The future direction of this work is to increase the volume of the data set to the maximum and testing the algorithm and suggest an algorithm to further improvise the performance.

REFERENCES

[1] Arjun Srinivas Nayak, Ananthu P Kanive, Naveen Chandavekan and Dr. Balasubramani R, “ Survey on Pre processing techniques for Text Mining”, *International Journal of Engineering and Computer Science,* Vol 5 (6), pp 16875-16879, 2016.

[2] Goyal R. D. 2007. Knowledge based neural network for text classification. In proceedings of the IEEE international conference on Granular Computing, pp. 542 – 547.

[3] Fragoudis et al. Balahur A., and Montoyo A.. 2008. A feature dependent method for opinion mining and classification. In proceedings of the IEEE international conference on Natural Language Processing and Knowledge Engineering, pp. 1-7.

[4] Isa D., Lee L. H., Kallimani V. P., and RajKumar R. 2008. Text document pre-processing with the Bayes formula for classification using the support vector machine.

[5] Vandana Korde et al Text classification and classifiers:” International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.2, March 2012”.

[6] Guan J., Zhou S., “Pruning Training Corpus toSpeedup Text Classification”, DEXA 2002, pp. 831-840.

[7] William B. Cavnar and John M. Trenkle""N-Gram-Based Text Categorization""vol.5 IJCSS 2010.

[8] Shuigeng Zhou, Jihong Guan, Evaluation and Construction of Training Corpuses for Text Classification: A Preliminary Study, Lecture Notes in Computer Science, Volume 2553, Jan 2002, Page 97-108.

[9] Jones K. S. 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, Vol. 28, No. 1, pp. 11-21.

[10] Menaka S and Radha N, “Text Classification using Keyword Extraction Technique”, *International Journal of Advanced Research in Computer Science and Software Engineering,* Vol 3 (12), pp 734-740, 2013.

[11] Jincy B Chrystal and Stephy Joseph, “Text Mining and Classification of product reviews using Structured Support Vector Machine”, Proceedings of CSCP, pp 21-31, 2015.

[12] Yoginee R surkar and Mohd S.W, “A Review on Feature Selection and Document Classification using Support Vector Machine”, *International Journal of Engineering Research and Technology,* Vol 3(2), pp 933-937, 2014.

[13] Jana Novovicova and Antonin Malik, “Information-Theoretic Feature Selection Algorithms for Text Classification”, *Proceedings of International Joint Conference on Neural Networks, Canada,* pp 3272-3277, 2010.

[14] Adamkani J and Nirmala K, “A Content Filtering Scheme for Social Sites”, *Indian Journal of Science and Technology,* Vol 8 (33), pp 1-8, 2015.