Feature Selection for Effective Text Classification using Semantic Information

Rajul Jain
PG Student
Department of Computer Engineering,
Maharashtra Institute of Technology Pune,
Pune, Maharashtra, India

Nitin Pise
Associate Professor
Department of Computer Engineering,
Maharashtra Institute of Technology Pune,
Pune, Maharashtra, India

ABSTRACT
Text categorization is the task of assigning text or documents into pre-specified classes or categories. For an improved classification of documents text-based learning needs to understand the context, like humans can decide the relevance of a text through the context associated with it, thus it is required to incorporate the context information with the text in machine learning for better classification accuracy. This can be achieved by using semantic information like part-of-speech tagging associated with the text. Thus the aim of this experimentation is to utilize this semantic information to select features which may provide better classification results. Different datasets are constructed with each different collection of features to gain an understanding about what is the best representation for text data depending on different types of classifiers.

General Terms
Text Classification

Keywords
Context, POS tagging, semantic information, text categorization

1. INTRODUCTION
The rebellious expansion of the internet has led to a great deal of interest in developing useful and efficient tools and software to assist users in searching the Web.

Most of the information content available on the internet is in the form of text data hence it is imperative to deal with text data. Text mining generally refers to the process of extracting interesting and non-trivial information and knowledge from unstructured text. Text categorization is a crucial research field within text mining. The crucial objective of text categorization is to recognize, understand and organize the volumes of text data or documents. The main issues are the complexity of natural languages and the extremely high dimensionality of the feature space of documents that convolute this classification problem. Thus machine learning has a dual role: Firstly we need an efficient data representation to store and process the massive amount of data, as well as an efficient learning algorithm to solve the problem. Secondly, the accuracy and efficiency of the learning model should be high to classify unseen documents. The momentous advantages of this approach over the knowledge engineering approach (consisting of manual definition of a classifier by domain experts) are a very good efficacy, significant savings in terms of expert manpower, and the possibility of easy generalization (i.e. easy portability to different domains) [1].

The process of text categorization can be broadly understood through the steps shown in Figure 1.The document set first needs to be converted to a representation suitable for classification which requires a sequence of steps that have been discussed in detail in the literature survey.

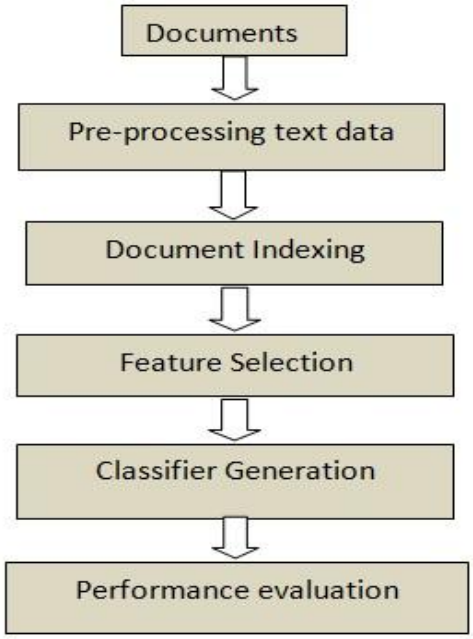

Figure 1: The process of text categorization

After this step the classifier can be trained and hence evaluated later for unseen data samples. Thus the main issues are concerning three different problems, viz. data representation, classifier training and classifier performance evaluation. These tasks actually form the main phases of the life cycle of a text classification system and are discussed briefly ahead.

2. LITERATURE SURVEY
A number of experiments have been performed to tackle the issues in text categorization. Here we can throw some light upon the subtasks involved in the process of text categorization along with the experiments done by many of the researchers:

2.1 Document Preprocessing
A document by itself is just a collection of words and hence needs to be first preprocessed and converted into a form where it is usable as a dataset by a classifier generating algorithm. Hence a document or text is usually represented by an array of words called the feature set. So a document can be presented by a binary vector, assigning the value 1 if the document

contains the feature-word or 0 if the word does not appear in the document. The basic steps that are part of the pre-processing stage are:

2.1.1 Tokenizing the text

The task of converting a text into tokens (called words or terms), which are then usable as features for classifier development.

2.1.2 Stop words removal

Not all of the words presented in a document can be used in order to train the classifier. There are futile words such as auxiliary verbs, articles and conjunctions, which are not useful for the classification process, such words are called stopwords. There exist many lists of such words, which are removed as a part of the pre-processing task.

2.1.3 Stemming Words

In order to reduce the size of the initial feature set, it is required to remove misspelled or words with the same stem. A stemmer, which performs stemming actually removes words with the same stem and keeps the stem or the most common of them as feature. For example, the words "train", "training", "trainer" and "trains" can be replaced with "train". The Porter's Stemming Algorithm is the most commonly used algorithm for word stemming in English [2].

2.1.4 Part-of-Speech Tagging

The process of assigning a part-of-speech or lexical class marker to each word in a corpus is called part-of-speech tagging**.** Words in a natural language tend to somehow behave alike e.g. appear in similar contexts, perform similar functions in sentences or undergo similar transformations, thus words may belong to certain classes. There are 9 traditional word classes for part-of-speech like noun, verb, adjective, preposition, adverb, article, interjection, pronoun, conjunction etc.

The traditional activities of stop words removal and stemming were the first most approaches for reducing the total number of words which will be used as features, whereas the process of part-of-speech tagging helps to identify features with the help of semantic information.

2.2 Document indexing

Document indexing denotes the activity of mapping a document d_j into a compact representation of its content that can be directly interpreted

(i) by a classifier building algorithm (during the training phase) and

(ii) by a classifier, once it has been built (during the testing phase).

The document indexing procedure needs to be uniformly applied to training, validation and test documents. The choice of a representation for text depends on what one regards as the meaningful constituent of text (the problem of lexical semantics) and the meaningful natural language rules for the combination of these entities (the problem of compositional semantics). An indexing method is characterized by a definition of what a term is and a method to compute term weights [3].

2.3 Data Representation Models

After converting an unstructured data into a structured data, we need to have an effective document representation model to build an efficient classification system. There are a number of representation techniques that have evolved over through research work done by various researchers in diverse domains. The various data representation models that have been proposed: Bag of Word (BoW) or Vector Space Model (VSM), term weighting approach [4][5], n-grams and n-multigrams approach[6], n-gram graph model[7], keywords or key-phrases approach, Latent Semantic Indexing (LSI)[8], Concise Semantic Analysis (CSA)[9], Rich Data Representation (RDR)[10].

The major drawbacks of the earliest, most popular and simplest VSM model are: high dimensionality of the representation, loss of semantic relationship that exist among the terms in a document and loss of correlation with adjacent words. This lead to different approaches proposed to incorporate semantic information to text representation some of them used a different approach of associating context information with words while others took the aid of background knowledge bases such as WordNet and ODP2. There are other research works done which try to utilize both syntactic as well as semantic information [11] to enhance the text categorization performance further. There are still other representation methods, one of which is an extension of the vector model adjusting the calculation of the tf*idf by considering the structural element instead of whole document is proposed in [12]. A remarkable improvement in precision, recall and F1-measures with the consideration of content and structure of the documents has been shown in the classification progress. A comparison of the Part of Speech (POS) Tagging and the use of WordNet features: synonyms, hypernyms, hyponyms, meronyms and topics have been performed with respect to a single classifier in [13]. To eliminate the ambiguity, a disambiguation method is proposed that gains better results, especially in Micro-F measure. A fusion of rule based approach and context association has been proposed in [14]. Apriori algorithm is used to find frequent words and frequent pattern of combination of words to identify context of terms and also help in enhancing classification efficiency. The relationship among words is used to derive the context of the words and hence the context of the document itself.

2.4 Dimensionality Reduction

A dimensionality reduction phase is often applied so as to reduce the size of the document representations. This has both the effect of reducing overfitting (i.e. the tendency of the classifier to better classify the data it has been trained on than the new unobserved data), and to make the problem more manageable for the learning method, since most of the learning algorithms are not easily scalable to large problem sizes. Dimensionality reduction is often performed through two types of approaches:

2.4.1 Feature Selection

The number of features representing the documents can be reduced by keeping only those which are most effective for the classification process and eliminating most of the features which are either irrelevant for classification or dependent on other features. The goal is to reduce the curse of dimensionality to yield improved classification accuracy and also the time consumption due to unnecessary processing. The methods for feature subset selection for text document classification task employ an evaluation function that is applied to each single word also known as terms. Tally of individual words can be performed using some of the measures like: document frequency, information gain, term strength, mutual information, χ^2 (Chi Square) statistics and many other such measures [15].The one thing that is common to all of these feature-scoring methods is that they wrap up by

ranking the features by their autonomously determined scores, and then select the best scoring features. Since there is no distinct metric that performs constantly better than all others, researchers often combine two metrics in order to benefit from both metrics [2].A few newly designed feature selection measures have also been proposed in [16] and have shown remarkable improvement in the classification performance.

2.4.2 Feature Transformation

A set of "artificial" terms is generated from the original term set in such a way that the newly generated features are both fewer in count and stochastically more independent of each other than the original ones and also provide a better classification parameter. Principal Component Analysis (PCA) is a well known method for feature transformation [2]. Another method based on PCA, which further reduces the size of representation is named Latent Semantic Indexing (LSI)[17], its origin has been in information retrieval community. Another approach called Linear Discriminant Analysis (LDA) has become a kind of popular dimension reduction method for pattern recognition [18]. Among the other efforts was to enhance efficiency of text categorization through summarization, to reduce both the dimensionality and the time take to process the data [19], [20].

2.5 Classifier Construction

A number of methods have been studied in the literature and utilized for document classification like decision tree classifiers[17], Naïve Bayes classifier[17], Rocchio's algorithm[21], Winnow algorithm[21], Sleeping Experts algorithm [21], k-nearest neighbour classifier[17], Support Vector Machines [22][23] and neural networks [24][25] etc. A few experiments also suggest enhancements on existing traditional algorithms like k-NN [10][26]. There are a number of other classifiers that have been used for experimentation, like centroid based classifiers and associative classifiers and a few others that have been discussed in [27][28], a few approaches involving use of hybrid techniques[29]. Apart from traditional classifiers there have been various experiments with the concept of combining classifiers [30] to form ensembles as a new direction for the improvement in performance of classification through individual classifiers. The employment of different base learner processes and/or different combination schemes [31] leads to different ensemble methods. There are many effective ensemble methods out of which three most popular methods are bagging, boosting, and stacking (or generalization)[32].

2.6 Classifier Evaluation

In text categorization research, effectiveness is considered the most important criterion, since it is the most reliable one when it comes to experimentally comparing different learners or different text categorization methodologies, given that efficiency depends on too volatile parameters (e.g. different software and/or hardware platforms). In text categorization applications, however, all three parameters are important, and one must carefully look for a trade-off among them, depending on the application constraints. There are a number of factors to evaluate the performance of the learned classifier e.g. the training time, the testing time, the accuracy, precision, recall etc. There are a number of basic measures like sensitivity, specificity, recall, precision. Apart from these basic measures other measures can be derived using the relationship among these basic measures. Some of the more popularly used measures that are widely used for estimating the performance of the classification system are: Accuracy, F1- measure, $F_ß$ –Measure [33], Macro-average measure and Micro- average measure etc. [34]. The F1-measure balances recall and precision in a way that it gives them equal weight. Its score is maximized when the values of recall and precision are equal or close; otherwise, the smaller of recall and precision dominates the value of Fl. For evaluating performance average across categories, there are two conventional methods, namely macro-averaging and micro-averaging. Micro-average performance scores give equal weight to every document, and are therefore considered a per-document average, while macro-average performance scores give equal weight to every category (or class), regardless of its frequency, and is therefore a per-category average[23].

$$\text{Macro-average F1} = 1/N \left(\sum^{N}_{i=1} F_i\right) \quad (1)$$

Where F_i is defines as:

$$F_i = \frac{2 \times \text{Precision}_i \times \text{Recall}_i}{(\text{Precision}_i + \text{Recall}_i)} \quad (2)$$

Where F_i is F-measure for class 'i' when total number of classes is N and Precision_i & Recall_i are precision and recall for each class 'i'.

$$\text{Micro-average F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{(\text{Precision} + \text{Recall})} \quad (3)$$

Precision and recall are calculated using the below formulae:

$$\text{Precision} = \frac{TP}{(TP + FP)} \quad (4)$$

$$\text{Recall} = \frac{TP}{(TP + FN)} \quad (5)$$

Where

TP: True Positives

FP: False Positives

FN: False Negatives

In earlier research work many experiments have been conducted where different algorithms have been used with different data representation schemes and for different datasets a few performances are summarized in the Table 1.

3. PROPOSED WORK

From the literature survey it is quite evident that a number of classifiers have already been experimented for the text classification. Our aim is to use classifiers of varied background like SVM, Naïve Bayes, k-NN classifiers which are already known to provide better performances for text classification. Also ensemble methods like decision tree ensemble and SVM ensemble can be used to obtain a better understanding about the effects of using a different data representation on the performance of various classifiers. Our aim is to design a feature set using semantic information in the text data to select features that help enhance the classification accuracy.

Table 1. Comparison of performances of different algorithms on three different datasets

Algorithm	Data Representation	Data Set	Accuracy
Centroid Based Algorithm	Single Terms	Reuters-21578	56.30
Centroid Based Algorithm	Phrase based		53.90
Decision Tree	Bag-of-Words (Vector Space Model)	Yahoo news group Data	73.80
Naïve Bayes			83.10
k-NN			75.70
Subspace method			79.60
Adaptive Combination of classifiers			82.21
Naïve Bayes	Bag-of-Words (Vector Space Model)	Ling-spam (Chinese)	96.63
Support Vector Machine			96.85
Decision Tree			98.48

Due to the enormous amount of information available in the form of digital documents on the internet, there is an eminent need to have a system which can efficiently process the documents such that has the capability to reduce the number of features and still provide a high degree of accuracy. For machine learning to be as close as possible to humans' process of identifying documents classes, the approach needs to be close to how humans decide on the question of classification of documents. Most humans can decide on the class of a document based on observing the words in the document, which mainly consist of nouns, verbs and adjectives associated with those nouns. A similar approach has been proposed by using POS (part-of-speech) tagging where words in documents can be identified by machine through the tags attached to them by the POS tagger.

The First important step in text categorization is text pre-processing, which involves cleaning data available in the dataset so that it can be processed easily. This may require removing tags and other unnecessary information in the dataset files i.e. removing noise from the documents. Next steps are stop words removal and stemming, which are required again to reduce unimportant words and squeeze the words with same stem to represent the stemmed words. For our experiment we use the Reuters-21578 Dataset, which is a standard dataset for text classification experiments. Apart from the basic steps of preprocessing our aim is to incorporate some semantic information with the tokens, and for this purpose Part-Of-Speech Tagging (POS Tagging) is the method to be utilized. Stanford Log-linear Part-Of-Speech Tagger by The Stanford Natural Language Processing Group has been utilized for POS Tagging.

3.1 Mathematical Model

Following is the relevant mathematics related to the proposed system and that is represented by set theory:

Let the System be S where

S= {*I, O, F, C***}**

Where,

I = Set of inputs

O = Set of output

F = Function of implementation

C = Set of constraints and assumptions

I = {D}, Where, D is a collection of input documents

$D = \{D_1, D_2, D_3 \ldots D_n\}$

Each D_i is represented as a vector of term weights of the form $D_i = \{tw_1, tw_2, tw_3 \ldots tw_m\}$

O = {S |where S is a set of datasets obtained by choosing different semantic components out of Documents}

F = {*F1, F2, F3, F4*} is a set of functions which comprise the total functionality of the system, where:

F1 = Preprocessing of set of Input documents

$F1 = \{I, F_{1a}, F_{1b} |$ I = input documents,

F_{1a} =Noise removal,

F_{1b} =POS Tagging}

F2 = Feature Extraction

$F2 = \{F_{2a} | F_{2a}$ = Feature Set generation}

F3 = Document Indexing

$F3 = \{F_{3a} | F_{3a}$ = Feature Vector generation or Document Indexing}

F4: Performance evaluation on WEKA classifiers

Constraints:

I ϵ Reuters 21578 Dataset

F1 is designed as per the structure of I

WEKA Classifiers have been used for performance analysis.

Tables 2 and 3 shown below provide a brief idea about the environment required for the development of the proposed work.

Table 2. Software Requirements

Operating System	Min 32-bit (Windows XP or above)
Programminng Language	Java JDK 1.6 (or above)
IDE	Eclipse for Java
Tools	WEKA (3.7.11) for evaluating resulting datasets.

Table 3. Hardware Requirements

Processor	P-IV or above
RAM	Min 2GB
HDD	Min 20 GB

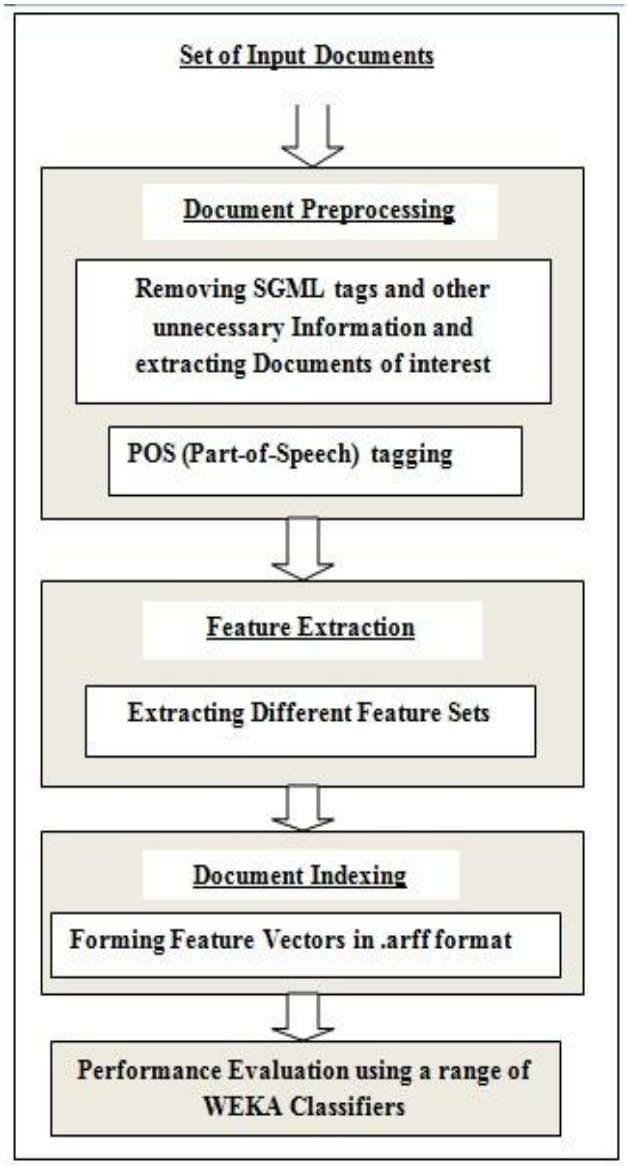

Figure 2: System Architecture

4. RESULTS

4.1 Dataset

The Reuters 21578 dataset has been used as input document set to be classified. It is a standard dataset used in many of the earlier research experiments for text classification. The Reuters-21578 Distribution 1.0 is scattered among a collection of 22 files. The first 21 files of this collection (reut2-000.sgm through reut2-020.sgm) contain 1000 documents, while the last (reut2-021.sgm) contains 578 documents [35]. The Reuters collection contains a variety of documents like multi-label classified documents as well as documents belonging to single category/label only, while for this experimentation work, files with single category labels only are considered.

For the Reuters-21578 collection the documents are Reuters newswire stories, and the categories are five different sets of content related categories. For each document, a human indexer decided which categories from which sets a particular document belonged to. The category sets are illustrated in Table 4. More details about this document collection can be obtained from [35]:

Table 4. Category sets of Reuters-21578 collection [35].

Number of Category Set	Number of Categories	Number of Categories w/1+ Occurrences	Number of Categories w/20+ Occurrences
EXCHANGES	39	32	7
ORGS	56	32	9
PEOPLE	267	114	15
PLACES	175	147	60
TOPICS	135	120	57

Only the top 10 categories of the TOPICS category set were selected out of the collection consisting of 135 categories in all. Thus the selected categories for this experiment are ‘earn’, ‘acq’, ‘money-fx’, ‘grain’, ‘crude’, ‘trade’, ‘interest’, ‘ship’, ‘corn’ & ‘wheat’. While ‘earn’ is the category with highest number of documents out of these whereas ‘corn’ and ‘wheat’ are actually coinciding categories with category ‘grain’, thus the last two categories do not have any document belonging to single class label only.

4.2 Result Set

The original file from the Reuters dataset were processed through Java programs for removing tags and other unnecessary information which is not helpful for the classification experiments done for this project work. Initially after removing tags and other irrelevant information the files were processed to extract single label documents of ten largest categories in the dataset in TOPICS category set.

The original structure of the document before any preprocessing operations is shown in the Figure 3.

```
</REUTERS>
<REUTERS TOPICS="NO" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET" OLDID="5545" NEWID="2">
<DATE>26-FEB-1987 15:02:20.00</DATE>
<TOPICS></TOPICS>
<PLACES><D>usa</D></PLACES>
<PEOPLE></PEOPLE>
<ORGS></ORGS>
<EXCHANGES></EXCHANGES>
<COMPANIES></COMPANIES>
<UNKNOWN>
&#5;&#5;&#5;F Y
&#22;&#22;&#1;f0708&#31;reute
d f BC-STANDARD-OIL-&lt;SRD>-TO   02-26 0082</UNKNOWN>
<TEXT>&#2;
<TITLE>STANDARD OIL &lt;SRD> TO FORM FINANCIAL UNIT</TITLE>
<DATELINE>    CLEVELAND, Feb 26 - </DATELINE><BODY>Standard Oil Co and BP North America
Inc said they plan to form a venture to manage the money market
borrowing and investment activities of both companies.
    BP North America is a subsidiary of British Petroleum Co
Plc &lt;BP>, which also owns a 55 pct interest in Standard Oil.
    The venture will be called BP/Standard Financial Trading
and will be operated by Standard Oil under the oversight of a
joint management committee.

 Reuter
&#3;</BODY></TEXT>
</REUTERS>
```

Figure 3: Structure of a single document in the original data file

After data cleaning, the collection of original dataset files were transformed into a collection of 8 files containing the documents for each single category in a single file, since out of the top 10 categories of the collection only 8 of these categories have documents with single classification label.

Thus a subset of the Reuters-21578 collection was obtained for experimentation.

The POS (part-of-speech) tagging requires complete sentences to be tagged in order for the correct part-of speech to be identified hence after extracting documents of interest POS (part-of-speech) tagging was performed on the original files with only tags and extra information removed. This resulted in POS tags attached to each word in the document which can be shown in the Figure 4.

```
SINGLE_JJ CATEGORY_NN :_: EARN_VB
CHAMPION_NN PRODUCTS_NNP <_JJR CH_NN >_JJR APPROVES_VBZ
STOCK_NNP SPLIT_NNP
Champion_NNP Products_NNPS Inc_NNP said_VBD its_PRP$
board_NN of_IN directors_NNS approved_VBD a_DT two-for-one_JJ
stock_NN split_NN of_IN its_PRP$
common_JJ shares_NNS for_IN shareholders_NNS of_IN record_NN
as_IN of_IN April_NNP 1_CD ,_, 1987_CD ._.
The_DT company_NN also_RB said_VBD its_PRP$ board_NN voted_VBD
to_TO recommend_VB to_TO
shareholders_NNS at_IN the_DT annual_JJ meeting_NN April_NNP
23_CD an_DT increase_NN in_IN the_DT
authorized_JJ capital_NN stock_NN from_IN five_CD mln_CD to_TO
25_CD mln_CD shares_NNS ._.
Reuter_NNP
```

Figure 4: Document after POS Tagging

After POS tagging of the documents five different datasets were formed by obtaining:

- only verbs,
- only nouns,
- nouns and verbs,
- nouns, verbs and adjectives, and
- nouns, verbs, adjectives and adverbs

The previous step gives five different dataset representations for the same set of documents chosen for the experiment. After this step of segregation into five different sets, unique words were chosen for each set separately and these unique words were then used as features for the classification process. Thus for generating the .arff files term frequency has been used as a measure and a lower cut-off has been used to eliminate words which appear in less than 5% of the total number of documents used in the classification process.

This resulted in five different datasets of documents to be processed in .arff (attribute relationship file format) format and these different data representations were then compared and analysed. The classification was performed using predesigned classifiers available in WEKA tool (version 3.7.11).A variety of classifiers were chosen to evaluate the performance in order to observe the behaviour of the dataset among different classes of classifiers. The different classifiers that were utilized are:

- Naive Bayes Classifier
- SMO Algorithm (polyKernel)
- 1-Nearest Neighbour Algorithm ,
- Random Committee Algorithms,
- Decision Table Classifier,
- PART
- J48 Classifier
- Random Forest Classifier
- MultiLayer Perceptron

A graph representing the performance summary of the obtained datasets is shown in figure 5. The graph is plotted between microaveraged-F1 values along the y-axis and the various classifiers on the x-axis for all five datasets.

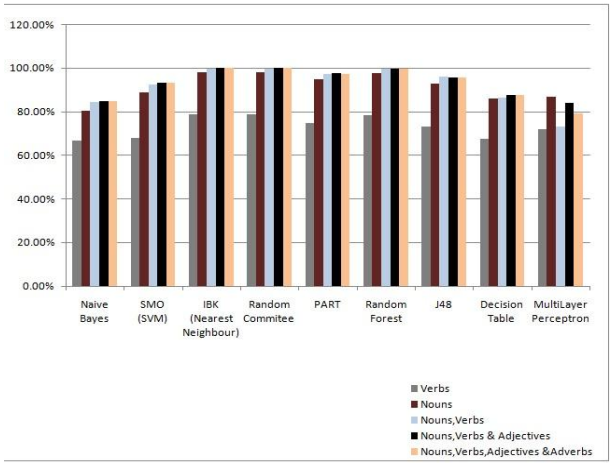

Figure 5: Performance Summary

A table showing the micro-averaged F1and macro-averaged F1 values for the various classifiers used for evaluation shows that consistently best performing dataset is the one with combination of nouns, verbs, and adjectives of the documents (see Table 5).

Table 5. micro-averaged and macro-averaged F1 values for consistently best performing dataset

Classifier Name	**Micro-average F1 values**	**Micro-average F1 values**
Naive Bayes Classifier	0.845	0.670
SMO Algorithm (polyKernel)	0.930	0.788
1-Nearest Neighbour Algorithm	**0.998**	**0.993**
Random Committee Algorithms	**0.998**	**0.993**
Decision Table Classifier	0.876	0.618
PART	0.975	0.924
J48 Classifier	0.955	0.841
Random Forest Classifier	0.996	0.989
MultiLayer Perceptron	0.841	0.434

5. CONCLUSION AND FUTURE WORK

Although the field of Text categorization has seen many innovations to enhance the accuracy and efficiency of the classification task, there are still many avenues of further exploration. A new data representation approach is proposed in order to achieve enhanced classification accuracy with the help of semantic information in the text data. This approach is quite closer to human approach of classification by observing the features which are words of the documents and mostly these words are nouns verbs and adjectives in the documents.

The performance of a dataset consisting of a combination of nouns, verbs and adjectives in the documents has shown a consistently high classification performance in terms of micro-averaged F1 measure. The best dataset yielded a correct classification percentage of almost 99.8% which seems like a promising performance.

Further learning performance can be enhanced with the use of combination approaches like using a combination of data representation and feature selection techniques. Adaptive learning which helps in building the knowledge base and also uses the information stored in the knowledge base is another avenue of research in text classification. The context identification of text data is still a field to be further explored.

6. ACKNOWLEDGMENT

I am thankful to my guide Prof. N. N. Pise, Department of Computer Engineering, MIT Pune, for his valuable and timely guidance, encouragement and instrumental support, which helped me to understand the depth and width of the topic. I am also thankful to my co-guide Prof. R. A. Agrawal, Department of Computer Engineering, MIT Pune whose guidance has helped me to understand the topic better because of her knowledge about the domain. Also I would like to express my sincere thanks to other faculty members of department of Computer Engineering, MIT Pune, for their extended help and suggestions at every stage.

7. REFERENCES

[1] Sebastiani F., "*Machine Learning in Automated Text Categorization*", ACM Computing Surveys, vol. 34 (1), 2002, pp. 1-47.

[2] Ikonomakis M., Kotsiantis S. and Tampakas V.: "*Text Classification Using Machine Learning Techniques*", WSEAS Transactions on Computers, Volume 4, August 2005.

[3] Sebastiani F. "Text categorization.", In Laura C. Rivero, Jorge H. Doorn and Viviana E. Ferraggine (eds.), The Encyclopedia of Database Technologies and Applications, Idea Group Publishing, Hershey, US, 2005, pp. 683-687.

[4] Harish B. S., Guru D. S. and Manjunath S.; "*Representation and Classification of Text Documents: A Brief Review*", in IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition" RTIPPR, 2010.

[5] Patra A. and Singh D.: "*A Survey Report on Text Classification with Different Term Weighing Methods and Comparison between Classification Algorithms*", International Journal of Computer Applications, Volume 75, August 2013.

[6] Shen D, Sun J-T, Yang Q, Chen Z: *"Text Classification Improved through Multigram Models*" at ACM Transactions at *CIKM'06,* Nov. 2006, Virginia, USA.

[7] Giannakopoulos G, Mavridi P, Paliouras G, Papadakis G, Tserpes K: "*Representation Models for Text Classification: a comparative analysis over three Web document types"*, ACM Transactions at WIMS'12, June 2012 , Romania.

[8] Gayathri K, Marimuthu A: "*Text Document Pre-Processing with the KNN for Classification Using the SVM*", Proceedings of 7th International Conference on Intelligent Systems and Control (ISCO 2013) IEEE.

[9] Zhixing Li, Zhongyang Xiong, Yufang Zhang, Chunyong Liu, Kuan Li: "*Fast text categorization using concise semantic analysis*", Pattern Recognition letters (2011), Elsevier.

[10] Keikha M, Khonsari A, Oroumchian F: "*Rich document representation and classification: An analysis*", Knowledge-Based Systems (2009), Elsevier.

[11] Suganya S, Gomathi C, ManoChitra S: "*Syntax and Semantics based Efficient Text Classification Framework*", International Journal of Computer Applications, Volume 65, March 2013.

[12] Chagheri S, Calabretto S, Roussey C, Dumoulin C: "*Feature Vector Construction Combining Structure and Content for Document Classification*", 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 IEEE.

[13] Celik K; Gungor T: "*A comprehensive analysis of using semantic information in text categorization*", International Symposium on Innovations in Intelligent Systems and Applications, 2013 IEEE.

[14] Kulkarni A.R. ; Tokekar V; Kulkarni P: "*Identifying context of text documents using Naïve Bayes classification and Apriori association rule mining*", CSI Sixth International Conference on Software Engineering, 2012.

[15] Niharika S., SnehaLatha V., Lavanya D.R.: "*A Survey on Text Categorization",* at the International Journal of Computer Trends and Technology- Volume3, 2012.

[16] Yan Xu: "*A Study for Important Criteria of Feature Selection in Text Categorization*", 2nd International Workshop on Intelligent Systems and Applications (ISA), 2010, IEEE.

[17] Li Y. H. and Jain A. K., "*Classification of Text Documents*", The Computer Journal, Vol. 41, No. 8, 1998, IEEE Journal.

[18] Wang Ziqiang, Qian Xu: "*Text Categorization Based on LDA and SVM*", 2008 International Conference on Computer Science and Software Engineering, IEEE.

[19] Jiang Xiao-yu, Fan Xiao-zhong, Chen Kang: "*Chinese Text Classification Based on Summarization Technique*", Third International Conference on Semantics, Knowledge and Grid, 2007 IEEE.

[20] Jiang Xiao-Yu, Fan Xiao-Zhong, Wang Zhi-Fei, Jia Ke-Liang: "*Improving the Performance of Text Categorization using Automatic Summarization*", International Conference on Computer Modeling and Simulation,2009 IEEE.

[21] Ragas H, Koster Cornelis H.A., "*Four Text Classification Algorithms Compared on a Dutch corpus*", In Proceedings of ACM Transactions SIGIR. '98.

[22] Joachims, T. (1998). "*Text Categorization with Support Vector Machines*: *Learning with Many Relevant Features".*Proceedings of ECML-98, 10th European Conference on Machine Learning.

[23] Ozg¨ur A., ¨Ozg¨ur L., and G¨ung¨or T., "*Text Categorization with Class-Based and Corpus-Based Keyword Selection*", P. Yolum et al.(Eds.): ISCIS 2005, Springer.

[24] Farkas Jennifer, "*Improving the Classification Accuracy of Automatic Text Processing Systems Using Context Vectors and Back-Propagation Algorithms*", at the Proceedings of the 1996 Canadian Conference on Electrical and Computer Engineering.

[25] Chen Z H, Huang L and Murphey Y Li: "*Incremental Learning for Text Document Classification*", International Joint Conference on Neural Networks, Orlando, Florida, USA, August 2007, IEEE.

[26] Jiang S, Pang G, Wu M, Kuang L: "*An improved K-nearest-neighbor algorithm for text categorization*", Expert Systems with Applications 39, 2012 Elsevier.

[27] Korde V; Mahender C. N.; "*Text Classification And Classifiers:A Survey*", at International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, March 2012.

[28] Antonie M., Zai'ane O, "*Text Document Categorization by Term Association*", at the Proceedings of ICDM 2002, IEEE, pp.19-26 ,2002.

[29] Khan Aurangzeb, Baharudin Baharum, Lee Lam Hong, Khan Khairullah: "*A Review of Machine Learning Algorithms for Text-Documents Classification*", In Journal Of Advances In Information Technology, Vol. 1, February 2010.

[30] Larkey L. S and Croft W. B, "*Combining Classifiers in Text Categorization"*, In Proceedings of ACM SIGIR'96.

[31] Qingxuan Chen, Dequan Zheng, Tiejun Zhao，Sheng Li: "*A Fusion of Multiple Classifiers Approach Based on Reliability function for Text Categorization*", Fifth International Conference on Fuzzy Systems and Knowledge Discovery, 2008 IEEE.

[32] Z.-H. Zhou., "*Ensemble learnin."* In: S. Z. Li ed. Encyclopedia of Biometrics, Berlin: Springer, 2009, 270-273.

[33] Silva Catarina, Ribeiro Bernardete: "*RVM Ensemble for Text Classification*", International Journal of Computational Intelligence Research. Vol. 3, pp 31–35, 2007.

[34] Lahlou F. Z., Mountassir A, Benbrahim H and Kassou I: "*A Text Classification Based Method for Context Extraction from Online Reviews*", 8th International Conference on Intelligent Systems: Theories and Applications (SITA), 2013 IEEE.

[35] Lewis, D., "Reuters-21578 text categorization test collection Distribution 1.0 README file (v 1.3)", 14 May 2004. Available online at *http://www. daviddlewis.com / resources / testcollections/ reuters21578/ readme.txt.*