Comparison of Text Categorization Accuracy with Different Document Representation

Ishant Sharma,Karan Khanna
Monalisa Das,Oindrila Samanta
Vikas Saran
Computer Science and Engineering Department
Motilal Nehru National Institute of Technology
Allahabad
{**ishant**‖**karan**}.*cse*16@*mnnit.ac.in*

Dr. Ranvijay,Manisha Yadav
Motilal Nehru National Institute of Technology
Allahabad
ranvijay@mnnit.ac.in, manishaar@mnnit.ac.in

***Abstract*—As we know that the digitization of text is increasing enormously, the need to categorize documents and classify text has become absolutely necessary. The response time of textual data or the time consumed in information retrieval can be reduced by categorization and classification of text. Therefore the task of organizing, categorizing and classifying texts and digitized documents is of extreme importance. This paper discusses the different text categorization algorithms like Naive Bayes, K-Nearest Neighbour and Support vector machine on well known 20 newsgroup dataset. The documents of the dataset have been represented as Bag of words(BOW) , Term frequency(TF) weighted form and Term frequency Inverse document frequency(TFIDF) weighted form. The accuracy of each classifier is tested and analysis has been done in each of the document representations and the best classification technique is suggested.**

***Keywords*—Text categorization; SVM; Nave Bayes; TF- IDF; KNN;BOW; Machine Learning**

I. Introduction

The amount of data generated every minute online is huge. How to process tons of such text is a trending topic for research. Text classification is one of the major problems among them. Text classification is the process of assigning a new document to an already known category. It is used as an essential task in many areas like information retrieval, email classification, spam detection, language classification etc. Most researchers in the recent past, have worked to find new text categorization algorithms in order to improve classification accuracy and reduce complexity. But very little research has been done in the area of document representation. Traditionally there are 3 models:

- vector space model
- probabilistic model
- inference network model

In this paper, we represent the documents in three forms: Bag of Words(BOW) model , Term Frequency(TF) weighted model and Term Frequency Inverse Document Frequency(TFIDF) weighted model.

Six types of text categorization methods prevalent now-a-days include: (1) K-nearest neighbors (KNN) ; (2) Naive Bayes; (3) Rocchio's method; (4) SVM ; (5) Regression models; (6)Decision trees. In this paper, we use three major categorization algorithms - Naive Bayes, KNN and SVM on different document representation models and suggest the best combination among all.

II. Motivation

The World Wide Web is enormously expanding every day. As a result, humans cannot classify documents on their own. So we need new technology to classify documents automatically. Hence, document classification is one of the most trending topic of research now-a-days. It has led to increasing research in techniques like Data mining, Machine Learning and Natural Language Processing(NLP). Classification problem is a major task in areas of information retrival(IR), language classification, spam Identification, classifying online news documents into categories like sports, entertainment, business etc.

Researchers have done a lot of work in text classification field. Most of this work is based on finding new algorithms to classify documents. In this paper, we suggest a method which works best among all, keeping in mind the underlying document representation models. We have tested our classifiers on 20 newsgroup [9] dataset which is a collection of approximately 20,000 newsgroup. documents, partitioned (nearly) evenly across 20 different newsgroups. This improvement would surely help in the prediction of documents which would be of immense important in the forthcoming era of digitization. Related Work has been enlisted in Section III ; Section IV covers the Problem Statement; in Section V the Solution Approach has been discussed; Result and Analysis has been given in Section VI; Section VII and Section VIII contains the Conclusion and Future Work, respectively.

III. Related Work

A lot of research is currently going on in relation to the various problems prevalent in the document classification and the techniques involved it it,with an attempt to improve its accuracy while dealing with the heterogenity and dimensionality of the data available.

Similar work can be cited in some of the papers as given below:

1) **Mingyong Liu and Jiangang Yang [1]** proposed an improvement of TF-IDF weighting on vector space model.They introduced a new parameter to represent the in-class characteristics, and called this class frequency, which calculates the term frequency in documents within one class. They conducted the experiments on a well-known data mining tool named WEKA using some commonly used algorithms like Naive Bayes, Bayes Network, KNN [3] and SVM.

Method	NB	Bayes Network	KNN
	20news	20news	20news
TF-IDF	61.9 %	65.3%	55.3%
TF-IDF-CF	77.1%	77.7%	64.9%

The accuracy achieved with SVM is 69.1% for TF-IDF weightage and 78.7% for TF-IDF-CF weightage.

2) **Ron Bekkerman and James Allan [2]** used bi-grams to improve the classification accuracy of 20 Newsgroup Dataset [9].They applied 4-fold cross-validation and used the popular SVM classifier. The achieved result is 91.80.4 of accuracy,whereas the baseline result of the distributional clustering setting on BOW document representations (with-out bigrams) is 91.3 0.4%.

3) **Thorsten Joachims [4]** compared the Rocchio classifier,its probabilistic variant and standard nave bayes classifier on three text categorization tasks. The results obtained are:

Techniques	20 Newsgroup
PrTFIDF	90.3
Bayes	88.6
TFIDF	82.3

4) **Wongkot Sriurai [5]** compared the feature processing techniques of Bag of Words (BOW) with the topic model (using LDA approach). Text categorization algorithms such as Naive Bayes (NB), Support Vector Machines (SVM) and Decision tree are used for experimentation.Then the results proved that the topic-model approach for representing the documents yield the best performance based on F1 measure of 79 an improvement of 11.1 over the BOW[6] model.

IV. Problem Statement

In text categorization problem, we are given a training set D(Train) = (d(1) , 1), . . . , (d(n) , n) of labeled text documents where each document d(i) belongs to a document set D and the label of d(i) is within a predefined set of categories C = c(1) , . . . , c(m). The goal in text categorization is to devise a learning algorithm that, given the training set D(train) as input, will generate a classifier (or a hypothesis) h : D C that will be able to accurately classify unseen documents from D. To classify a document, we have to first preprocess the document and then represent it in a way so that we can do statistical analysis on it easily. Document representation can be done in three ways: Bag of words (BOW) , Term frequency weighted matrix of document and TFIDF[8] weighted matrix of document. Our aim is to represent documents in each of these forms, classify them using Naive Bayes, KNN[3] and SVM classifiers, and then compare the accuracy of each to find the most suitable method for classification. We have tested our classifiers on 20 newsgroup [9] dataset, which is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups.

Our work is divided into five stages:

1) Preprocessing Refining the emails in the dataset, stemming, stopword removal, and tokenization.
2) Representation of documents into BOW [6] (bag of words), TF, TF-IDF.
3) Generating Model using the training set.
4) Testing the documents on the generated model calculating accuracy.
5) Comparing different classification models on the same 20 NG dataset.

V. Solution Approach

Our Solution Approach has the following steps:

1) **Preprocessing:**
   a) Reorganise Dataset into two categories - Likes and Dislikes. Example: If we have taken 6 of the 20 newsgroups [9] as our dataset, Categories = [‘electronics‘, ‘crypt‘, ‘space‘, ‘hockey‘, ‘motorcycles‘ and ‘forsale‘] then in this step, we will divide them into 2 categories - Likes and Dislikes. One such division can be like this. Likes = [‘hockey‘, ‘crypt, electronics] Dislikes = [‘motorcycles‘, ‘forsale, ‘space‘].
   b) Remove incompatible files - Some files might be incompatible with utf-8. We will remove them in this step.
   c) Tokenization - Convert data into list of strings.
   d) Remove stopwords - Some words like a, an, the does not play any significant role in classification, these words are called stopwords. We remove such words from our data.
   e) Refine emails in data files -If we closely look at the dataset, the headers of email has some unnecessary information.
   We delete this information in the header of emails.
   We delete only lines in the email that starts with ’Path:’, ’Newsgroups:’, ’Xref:
   Example of an email Header:
   Path:
   cantaloupe.srv.cs.cmu.edu!rochester!udel!gatech!
   howland.reston.ans.net!usc!cs.utexas.edu!q
   t.cs.utexas.edu!news.Brown.EDU!
   noc.near.net!bigboote.
   WPI.EDU!bigwpi.WPI.EDU!kedz
   From: kedz@bigwpi.WPI.EDU (John Kedziora)
   Newsgroups: misc.forsale.
   Subject: Motorcycle wanted.
   Date: 22 Feb 1993 14:22:51 GMT
   Organization: Worcester Polytechnic Institute
   Lines: 11
   Expires: 5/1/93

Message-ID: ¡1man-jrja0@bigboote.WPI.EDU¿
NNTP-Posting-Host: bigwpi.wpi.edu
Sender:
Followup-To:kedz@wpi.wpi.edu
Distribution: ne
Organization: Worcester Polytechnic Institute

2) **Document Representation**
   a) **Bag of words (BOW [6]) - Boolean Representation** : In this representation, attributes are binary variables indicating the presence or absence of a term in the document (ignoring the number of occurrences) and are modeled by a discrete distribution. We form a matrix where rows indicate document and columns indicate features. Then we indicate the presence or absence of features in a document by 1 or 0.
   b) **Term Frequency [8] Representation** In the term-count representation attributes are random variables taking values from the set of natural numbers (0, 1, 2 ...) and are modeled by the multinominal distribution. Term frequency of a term j in document i is the number of occurences of term j in document i divided by the total number of terms in document i.
   c) **TFIDF [8] Representation** In the TFIDF [8] representation, document attributes are considered as normally distributed continuous variables. It puts weighting to a term based on its inverse document frequency. It means that if the more documents a term appears, the less important that term will be, and the weighting will be less. IDF for a feature is same for the overall dataset. It can be depicted as:

$$TFIDF = TF \times IDF =$$

$$Log(tf(i,j)+1) \times log(\frac{N+1}{n(j)})$$

Here, tf(i,j) is term frequency of term j in document i.
n (j) is the the number of ducuments that contain the term j.
N represents the total number of documents in the dataset.
We form a matrix where rows indicate document and columns indicate features.
The matrix is filled with TF-IDF weights of features for each document.

3) **Splitting dataset into training and testing set** The dataset is split into training and testing set. We train the classifier with the training set and then predict the class of testing set using the trained classifier. The classifiers used are -
   a) NB
   b) SVM
   c) KNN
4) **Plot Results** For each of the classifiers, we plot the result of classification accuracy where the train-test split is varied from 99 percent to 1 percent. We have used matplotlib for plotting purpose which is a python package.
5) **Comparing accuracy of different classifiers**

The solution approach can be summed up in the following manner:

1) **Framework Structure**

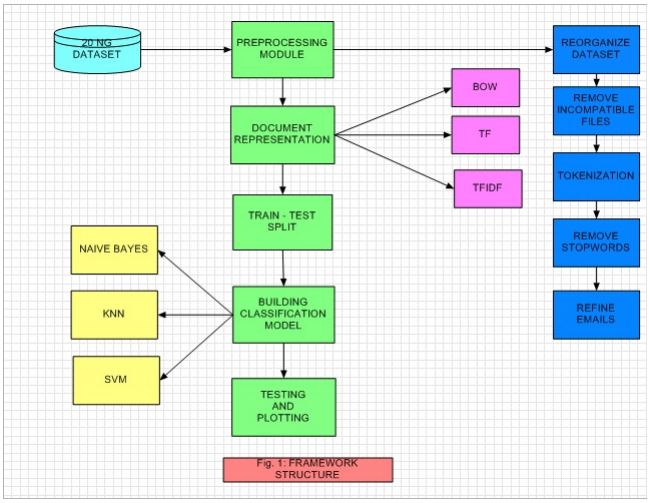

Fig. 1. Framework

2) **Description** The various experiments have been performed by taking the following classification techniques into account, namely :
   a) Nave Bayes [10]
   b) Support Vector Machine [10]
   c) K- Nearest Neighbour [3] [10]

   In each of the techniques, we used the following approach:
   - The train-test split ratio has been varied from 99% to 1% for the given data.
   - At each train-test split, precision is calculated for Bag of Words, TF, and TFIDF representation of data and is plotted.
   - We have performed the experiments in 2 parts :
     - Using 6 of the 20 newsgroup(electronics‘, ‘crypt‘, ‘space‘, ‘hockey‘, ‘motorcycles‘ and ‘forsale‘) categories.
     - Using the entire 20 newsgroup [9] dataset.

   The graphs corresponding to the various techniques, based on the conducted experiments are:

VI. Result and Analysis

For our experiment, we have used the subset of dataset.
This subset includes 6 of the 20 newsgroups :’space’, ’electronics’ , ’crypt’ , ’hockey’ , ’motorcycles’ and ’forsale’.

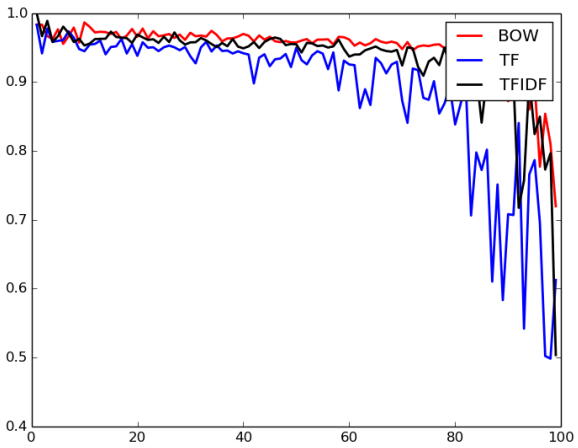

Fig. 2. (y Accuracy, x Split ratio) [Naive Bayes]

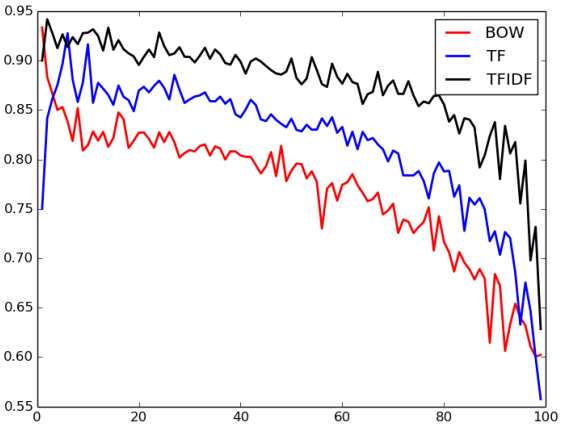

Fig. 3. (y Accuracy, x Split ratio) [KNN]

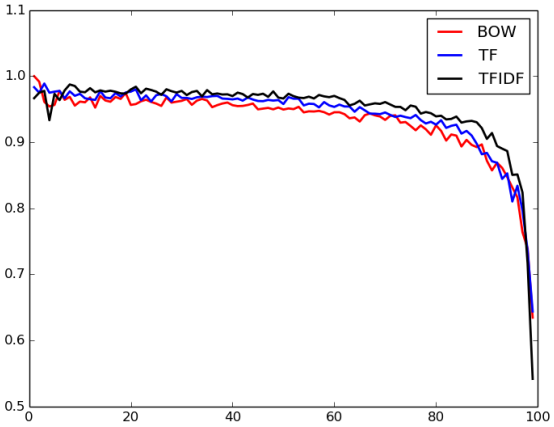

Fig. 4. (y Accuracy, x Split ratio) [SVM]

We assume that we like 'hockey', 'crypt' and 'electronics' newsgroups and we dislike the others.

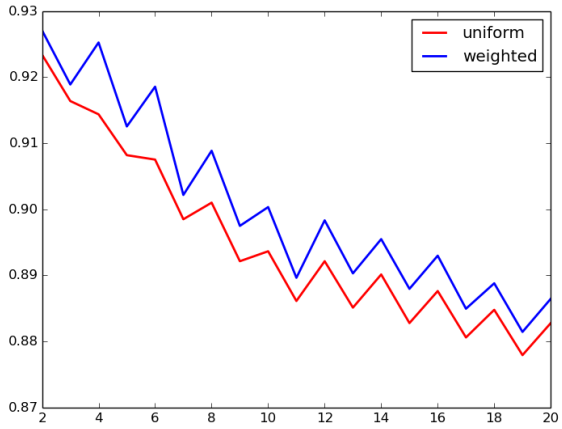

Fig. 5. (y Accuracy, x Kth Nearest Neighbour) [KNN (Uniform and Weighted)]

Method	Accuracy
BOW-NB-20% test	0.97
TF-NB-20% test	0.94
TFIDF-NB-20% test	0.95
TFIDF-SVM-20% test	0.97
TFIDF-SVM-90% test	0.92
TFIDF-SVM-5 FOLD 20 classes	0.893 (+/- 0.003 std)
TFIDF-5-NN - Distance Weights-20% test	0.90
TFIDF-5-NN-Uniform Weights-20% test	0.93
TFIDF-5-NN-Distance Weights-5 FOLD	0.909 (+/- 0.004 std)
TFIDF-5-NN-Distance Weights-5 FOLD-20 classes	0.756 (+/- 0.003 std)

VII. Conclusion

The results show that text classification can be effectively done by simple tools like TFIDF [8] and SVM. We have found that TFIDF along with SVM have the best performance. TFIDF with SVM perform well both for 2-class problem and 20-class problem.

VIII. Future Work

As a future work, we need to research on regression models used in text classification. Our future work also includes incorporating background knowledge about words, like using WordNet for their synonyms and finding relationships between words to improve accuracy. There are many words which have more than one synonyms having different meaning. We will work on building a disambiguation strategy to identify proper synonym of any word.

REFERENCES

[1] Mingyong Liu and Jiangang Yang, *"An improvement of TFIDF weighting in text categorization"*, 2012.

[2] Ron Bekkerman , James Allan, *"Using Bigrams in Text Categorization"* , 2003.

[3] Bruno Trstenjak, Dzenana Donko and Sasa Mikac, *"KNN with TF-IDF Based Framework for Text Categorization"*, 2013.

[4] Thorsten Joachims *"A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization"*.

[5] Wongkot Sriurai *"Improving Text Categorization by Using a Topic Model, International Journal Advanced Computing, Vol.2, Issue 6, pp 21-27"*, 2011.

[6] Juan Ramos *"Using TF-IDF to Determine Word Relevance in Document Queries"*.

[7] Ami Singhal *"Modern Information Retrieval: A Brief Overview. IEEE, 2001: 2-4"*.

[8] Matthew A. Russell *"Mining the Social Web. OReilly Media; Second Edition"*, October 22, 2013.

[9] 20 Newsgroup Dataset *"http://qwone.com/ jason/20Newsgroups/"*.

[10] Text Categorization Algorithms *"http://answers.google.com/answers/main?cmd=threadviewid=225316"*.