Analysis of Text Classification with various Term Weighting Schemes in Vector Space Model

Shitanshu Jain, S. C. Jain, Santosh Vishwakarma

Abstract: Term Weighting Scheme (TWS) is a key component of the matching mechanism when using the vector space model In the context of information retrieval (IR) from text documents, the this paper described a new approach of term weighting methods to improve the classification performance. In this study, we propose an effective term weighting scheme, which gives highest accuracy with compare to the text classification methods. We compared performance parameter of KNN and Naïve Bayes Classification with different Weighting Method, Weight information gain, SVM and proposed method.We have implemented many term-weighting methods (TWM) on Amazon data collections in combination with Information-Gain and SVM and KNN algorithm and Naïve Bayes Algorithm.

Keywords: Text Mining, Text Classification, Term Weighting, KNN, Naïve Bayes, SVM

I. INTRODUCTION

Data- Mining methods are frequently used for extracting exact information from large data-sets [1]. Data mining and text mining deal with structured and unstructured Data. IN Data Mining it deals with structure data and in text-mining it deal with unstructured or Semi- structured data. For example any text written document, E-mail etc are example of semi structured data set. The objective of Text-Mining is to extract the un-identified knowledge and information from multiple data sources [2]. Various Text-classification methods are used to generate (convert) the text-document into pre-define classes [3] Information retrieval is finding the information from collection of the documents. User process the query to information retrieval system and information retrieval system evaluate the rank of the document and index the documents in the collection, And display the result to the user which is most relevant to the user query. [4] In the Vector Space model, a document is represented as a vector in the term- spaces, Dj = (W1j , ...,Wkj),where K is the size of the set of terms (features). The value of WKj between (0, 1) represents how much the term tk contributes [5] to the semantics of document Dj .

Revised Manuscript Received on August 30, 2020.
***** Correspondence Author

Shitanshu Jain PhD Scholar, Amity University, Noida, (Uttar Pradesh), India.
Dr. S. C. Jain, Director, ASET, Amity University, Noida, (Uttar Pradesh), India.
Dr. Santosh K. Vishwakarma, Associate Professor, Department of CSE, Manipal University Jaipur, India.

II. TERM-WEIGHTING SCHEME: SURVEY AND ANALYSIS

Term weighting is a procedure that takes place during the text indexing process in order to assess the value of each term to the document. Term weighting is the assignment of numerical values to terms that represent their importance in a document in order to improve retrieval effectiveness [6]. Essentially it considers the relative importance of individual words in an information retrieval system, which can improve system effectiveness, since not all the terms in a given document collections are of equal importance. Weighing the terms is the means that enables the retrieval system to determine the importance of a given term in a certain document or a query. It is a crucial component of any information retrieval system, a component that has shown great potential for improving the retrieval effectiveness of an information retrieval system [5].

In Vector Space Model (VSM), each text document is represented as a vector of index terms in which each term is associated with a weight (score) that measures how formative/ discriminative the correspondent term is. The method which assigns a weight to a term is called Term Weighting Scheme (TWS). The Basic idea of vector space model is representing the document in computer understandable form. In space Model, any text document is represented as vectors or dimensions. Each dimension of space is to represent as a single feature of the vector and the weight is calculated by various weighting schemes. The document can be represent as D = (t1,W1;t2,W2….tn,Wn) which ti is a terms Wi is the weight of the ti in the document d. Reflect the importance of the term in a document use term weighting.[7]. We created two small documents in a corpus and combine these two documents and perform the Term Occurrences, Term Frequencies (TF), and Term Frequency-Inverse Document Frequencies (TF-IDF) work in Rapid-Miner.

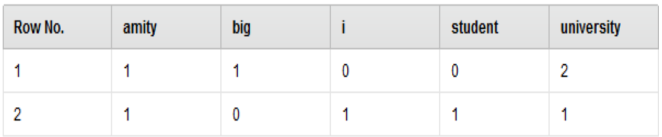
Doc1: Amity University is very big university
Doc2: I am Student is Amity University

A. Term occurrence

This is the most basic term weighting method for vectorization of word token. In this method we have crated term occurrence word vector (Nij), which perform counting of the word means word token occurrence (Ti) in every document (Dj).

$$N_{ij}= T_i(D_j) \qquad (1)$$

The number of occurrence of word Ti in document Dj is represented by term occurrence (Nij).

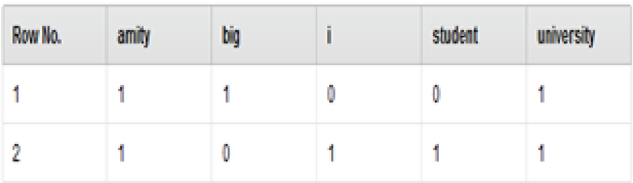


Row No.	amity	big	i	student	university
1	1	1	0	0	2
2	1	0	1	1	1

Fig. 1.Term Occurrences for two documents

B. Binary Term Frequency

The only difference in this method that it converts the occurrence of the words only into 0 and 1 as represented that change in figure 2 'University' word is occurring two times in doxument1 but rapid miner change 2 to 1 in document1, it shows that single time occurrence of the university word



Row No.	amity	big	i	student	university
1	1	1	0	0	1
2	1	0	1	1	1

Fig. 2.BinaryTerm frequency for two documents

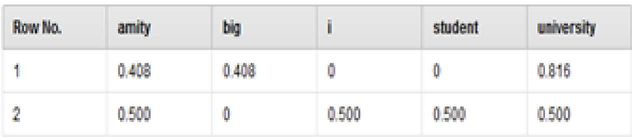
C. Term Frequency (TF)

Fundamental operation of the term-frequency is very easy and simple, Term-Frequency is find out the factor of the occurrence of particular word taken to the sum of number of word token in a document. To implement this we extract Token Number operator inside the Process Documents to get the total number of word token in each document.

Term Frequency word vectors are normalized vectors and same as unit vector normalization, the norm of a vector (Euclidean) or its size or length. We are able to locate the norm with the use of Pythagorean Theorem. Subsequently the norm of the document1 Term-Frequency vector 0.408 and the document2 Term-Frequency Vector 0.500.

$$TFij = \frac{Nij}{\sum_{K} Nij} \qquad (2)$$

TFij represent the number of occurrence of term i in document j. With a view to look at all the document equally, we need all of the file vector to have the equal value of length So we divide every document term frequency vector by means of its respective norm to get Documents Term Frequency, so if we divide every term frequency in every documents term frequency vector by means of its norms so we will get normalized term frequency vector for each file as shown in figure 3.



Row No.	amity	big	i	student	university
1	0.408	0.408	0	0	0.816
2	0.500	0	0.500	0.500	0.500

Fig. 3.Term frequency for two documents

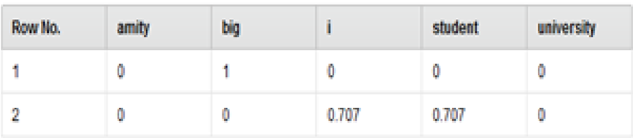
D. Term Frequency Inverse Document frequency (TFIDF)

TF-IDF is stands for Term Frequency-Inverse Document Frequency. The process of this method is used in information retrieval and text mining. It works on dataset or word of document-collection. The process of TF-IDF is used to define The time taken by a specific word in a document, number of documents with that specific word and ratio between the documents with that specific term by all documents. Stop words, high frequency and low frequency word are removed by TF-IDF. Tf-IDF methods are applicable for text summarization and classification.

$$IDF(W) = log\frac{N}{dft} \qquad (3)$$

$$TF\text{-}IDF = TFij * IDF(W) \qquad (4)$$

Hence the TF-IDF of the first document and the second document are given in figure 4.



Row No.	amity	big	i	student	university
1	0	1	0	0	0
2	0	0	0.707	0.707	0

Fig. 4.TF-IDF for two documents

III. METHODOLOGY

The experiments in this paper are carried out with the open source data mining tool RapidMiner 9.1. It is based on Java with a user interface for designing the approach by use of different operators. It uses an advanced concept, where many operators are used for designing the solution of a given problem. Naïve-Bayes and K-Nearest Neighbor algorithms (NB and KNN) of classification methods are used in data set for classification. Both the algorithms are used to apply on the dataset for training of the dataset and generate performance model, based on this performance model we perform testing of the dataset and check and evaluated the performance of the dataset in term of accuracy in the testing dataset.

Naïve Bayes algorithm uses the probabilistic-classifier for the classification of text-document collection. Navie-Bayes Model is very high reactive for selection of the attribute which manages only low dimensions. This method is very easy to implement [8]. The equation for Naive Bayes is given below:

$$p(C_i/X) = \frac{p(X/C_i).p(C)}{p(X)} \qquad (5)$$

P(C|X): Posterior Probability which is computed given feature, X for the Class 'C'.

P(C): Prior Probability for Target Class 'C'.

P(X|C): Possible Probability of feature 'X' given Class 'C'.

P(X): Prior Probability of the feature.

Target class is represented for the object as which will have the maximum Posterior Probability and this prediction done by Naïve-Bayes Classifier. So which has the highest posterior probability is the Target Class.

K-Nearest Neighbor (KNN) algorithm is important method for text-classification and it is easy to use and implemet in data mining. Value of 'K' is user-defined means given by the user and arbitrary .This algorithm is predict feature of the Target Class and based on neighbors; as to which class an unknown object will belong to is done using votes taken from the neighbors [9]. The object will belong to its closest neighbor which is also called Target-Class for the object. The measure used for determining neighbors (i.e. how far or how close they are) is done using distance functions such as Euclidean, Manhattan, Hamming distance, etc. The mathematical expression for k-NN distance functions can be given as:

$$\text{Euclidean Distance} = \sqrt{\sum_{i=1}^{k} (x_i - y_i)} \qquad (6)$$

$$\text{Manhattan Distance} = \sqrt{\sum_{i=1}^{k} |x_i - y_i|} \qquad (7)$$

$$\text{Hamming Distance} = \sum_{i=1}^{k} |x_i - y_i| \qquad (8)$$

Where x and y are the two target class labels. The distance is calculated for all the neighbors and the class with minimum distance from the object is predicted as the target class for the unknown objects.

IV. RESULT AND DISCUSSION

In this work, we have implemented this work using Naive Bayes and K-NN classifiers, we can implement this work with many classification algorithm means this implementation work in not limited for above mentioned classifier in data-mining. Many other popular classification techniques are used for dataset classification and analyses of output. This work performs a comparative study of many classification algorithms along with their different performance parameters. We analyze the output of all the classifiers and make a study on their performance and accuracy in making predictions.

Table I showing below, Performance table shows the comparative-study for many classification techniques along with different weighting scheme and We have performed further analysis using graphs. These graphs show the analysis of different performance parameters which we have used for various different classifiers such as K-NN, Naive Bayes,

Table-I: Text classification with different term weight scheme

Term Weight	Classification Method	Parameter Accuracy
Term Occurrence	KNN	96.53%
Term Occurrence	Naïve Bayes	91.68%
Term Binary Occurrence	KNN	96.53%
Term Binary Occurrence	Naïve Bayes	91.68%
Term frequency (TF)	KNN	96.43%
Term frequency (TF)	Naïve Bayes	91.77%
TF-IDF	KNN	96.38%
TF-IDF	Naïve Bayes	91.71%

In table 1, we present the KNN and naïve Bayes Classification methods with different term weighting schemes, which shows highest performance, considering both Term Occurrence and Binary Term Occurrence. KNN with term occurrence weighting scheme has 96.53% accuracy and KNN with Binary term occurrence weighting scheme has 96.53% accuracy.

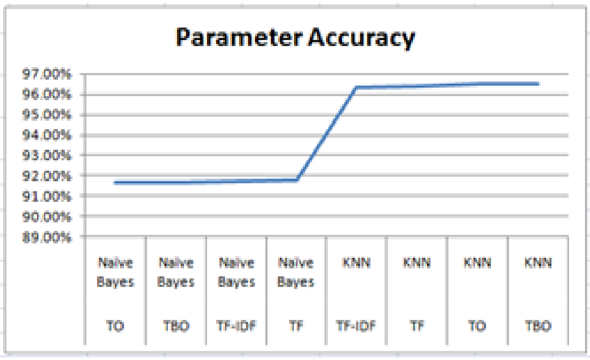


Fig.5. Text classification Accuracy with different term weight scheme

Table-II: Text Classification with different term weight using Information Gain

Term Weight and information gain	Classification Method	Parameter Accuracy
Term Occurrence	KNN	64.62
Term Occurrence	Naïve Bayes	38.42
Term Binary Occurrence	KNN	38.42
Term Binary Occurrence	Naïve Bayes	64.62
Term frequency (TF)	KNN	64.62
Term frequency (TF)	Naïve Bayes	38.42
TF-IDF	KNN	38.42
TF-IDF	Naïve Bayes	64.62

In table 2, we present the KNN and naïve Bayes Classification methods with different term weighting schemes and information gain which shows highest performance, considering both Term Occurrence and Binary Term Occurrence. KNN with term occurrence weighting scheme has 64.62% accuracy and KNN with Binary term occurrence weighting scheme has 64.62% accuracy. Results are very close in this implementation.

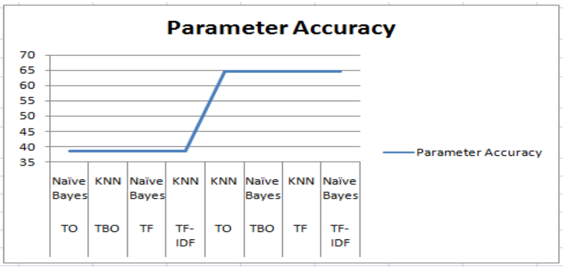


Fig.6. Accuracy of Text Classification with different term weight using Information Gain

Table-III: Text Classification with different term weight using SVM

Term Weight and SVM	Classification Method	Parameter Accuracy
Term Occurrence	KNN	86.22
Term Occurrence	Naïve Bayes	53.82
Term Binary Occurrence	KNN	86.22
Term Binary Occurrence	Naïve Bayes	56.49
Term frequency (TF)	KNN	85.93
Term frequency (TF)	Naïve Bayes	59.66
TF-IDF	KNN	91.23
TF-IDF	Naïve Bayes	57.58

In table 3, we present the KNN and naïve Bayes Classification methods with different term weighting schemes and SVM which shows highest performance, considering both Term Occurrence and Binary Term Occurrence.

KNN with term occurrence weighting scheme has 86.22% accuracy and KNN with Binary term occurrence weighting scheme has 86.22% accuracy.

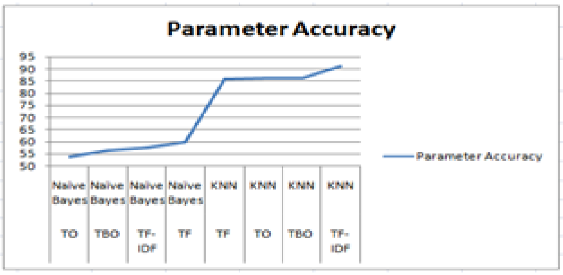


Fig.7. Accuracy of Text Classification with different term weight using SVM

In the above mentioned Performance table shows the comparative study for basically two classification techniques K-NN and Naïve-Bayes along with their performance - parameters such as Accuracy, Classification Error rate, Kappa, Precision and Recall.

We generate classification model of all KNN and naïve bayes classification method and also all three table shows comparative study for different techniques classifiers with their performance-parameter. We can see in all tables, we started our analysis with K-NN and Naïve-Bayes and K-NN is achieved highest accuracy of 96.53% and also Naïve-Bayes classifier is achieved high accuracy of 91.68%.

This work may not be appropriate for classification of Amazon-reviews where there are not a lot of features but the focus and emphasis is on Feedbacks. K-NN has achieved overall best performance in comparison to other classifiers.

V. CONCLUSION

The purpose of this paper is to give approaching of different Term Weighting Scheme presented for Text- Classification. The proposed method gives highest accuracy with compare to the traditional methods of text classification. The reason is that it has the highest number of true positives thus increasing the accuracy. The Accuracy parameter has increased in our proposed approach. Based on the various Term Weighting Scheme, Information Gain and SVM. We find that the proposed method outperforms in most of the metrics.

REFERENCES

1. J. Han and M. Kamber, Data Mining- Concepts and Techniques, 3rd edition San Francisco,USA, Morgan Kaufmann, Boston, MA, USA, Elsevier, 2006.
2. Ramzan Talib, Muhammad Kashif Hanif, ShaeelaAyesha , and Fakeeha Fatima ,Text Mining Techniques, Applications and Issues, International Journal of Advanced Computer Science and Applications, Volume 7 No. 11, 2016.
3. Chauhan Shrihari and Amish Desai, A Review on Knowledge Discovery using Text Classification Techniques in Text Mining, International Journal of Computer Applications (0975 – 8887), Volume 111 , No 6, February 2015.
4. M. Balamurugan, E.Iyswarya, "A Trend Analysis of Information Retrieval Models" International Journal of Advanced Research in Computer Science, Volume 8, No. 5, May-June 2017.
5. Salton, G. , and Buckley, C. 1988, Term-weighting approaches in automatic text retrieval, Inf. Process. Manage, 24(5), 513–523.
6. Salton G. and Mc-Gill M, Introduction to Modern Information Retrieval, McGraw-Hill Book Company, New-York, NY, 1983.
7. S.Brindha , Dr. K.Prabha , Dr. S.Sukumaran, The Comparison Of Term Based Methods Using Text Mining, International Journal of Computer Science and Mobile Computing, IJCSMC, Volume-5, Issue.-9, September 2016, page 112 – 116.
8. Dey, Lopamudra, et al., Sentiment Analysis of Review Datasets Using Naive Bayes and K-NN Classifier, arXiv preprint arXiv-1610.09982, (2016).

AUTHORS PROFILE

Shitanshu Jain has completed bachelors and master's degree of engineering in Computer Science & Engineering from RGPV University, Bhopal. He is currently PhD scholar in Amity University Madhya-Pradesh. His specialization includes Machine learning and data mining algorithms.

Dr. S.C.Jain is working a Director in the ASET , Amity University, Madhya-Pradesh. He has completed his bachelor's degree, master's degree and PhD in CSE from BITS Pilani, IIT Kharagpur and College of Defence Management. He is a doctorate in the field of Networking..

Dr. Santosh K. Vishwakarma is working as Associate Professor in CSE Deptt., Manipal University Jaipur. He is completed his bachelor's and master's degree in CSE. He is a doctorate in the field of Information Retrieval. He holds 15 years of Teaching Experience in reputed Institute. His research interest includes data mining, text mining, and predictive analysis