International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume: 3 | Issue: 2 | Jan-Feb 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 - 6470

Analysis of Text Classification Algorithms: A Review

Nida Zafar Khan[1], Prof. S. R. Yadav[2]

[1]Research Scholar, [2]Assistant Professor
Department of Computer Science Engineering, MITS, Bhopal, Madhya Pradesh, India

ABSTRACT
Classification of data has become an important research area. The process of classifying documents into predefined categories based on their content is Text classification. It is the automated assignment of natural language texts to predefined categories. The primary requirement of text retrieval systems is text classification, which retrieve texts in response to a user query, and text understanding systems, which transform text in some way such as answering questions, producing summaries or extracting data. In this paper we are studying the various classification algorithms. Classification is the process of dividing the data to some groups that can act either dependently or independently. Our main aim is to show the comparison of the various classification algorithms like K-nn, Naïve Bayes, Decision Tree, Random Forest and Support Vector Machine (SVM) with rapid miner and find out which algorithm will be most suitable for the users.

Keywords: *Text Mining, K-nn, Naïve Bayes, Decision Tree, Random Forest and Support Vector Machine, Rapid miner*

1. INTRODUCTION

Text mining or knowledge discovery is that sub process of data mining, which is widely being used to discover hidden patterns and significant information from the huge amount of unstructured written material. Text mining is largely growing field of computer science simultaneously to big data and artificial intelligence. Text mining and data mining are similar, except data mining works on structured data while text mining works on semi-structured and unstructured data. Data mining is responsible for extraction of implicit, unknown and potential data and text mining is responsible for explicitly stated data in the given text [1]. Today's world can be described as the digital world as we are being dependent on the digital / electronic form of data. This is environment friendly because we are using very less amount of paper. But again this dependency results in very large amount of data. Even any small activity of human produces electronic data. For example, when any person buys a ticket online, his details are stored in the database.

Today approx 80% of electronic data is in the form of text. This huge data is not only unclassified and unstructured (or semi-structured) but also contain useful data, useless data, scientific data and business specific data, etc. According to a survey, 33% of companies are working with very high volume of data i.e. approx. 500TB or more. In this scenario, to extract interesting and previously hidden data pattern process of text mining is used. Commonly, data are stored in the form of text. Broadly there are five steps involved in Text Data Mining. They are:

1. Text Gathering
2. Text Pre-processing
3. Data Analysis (Attribute generation & selection)
4. Visualization (Applying Text Mining algorithms)
5. Evaluation

For this text mining uses techniques of different fields like machine learning, visualization, case-based reasoning, text analysis, database technology statistics, knowledge management, natural language processing and information retrieval [2].

2. TEXT PRE-PROCESSING

The pre-processing itself is made up of a sequence of steps. The first step in text-pre-processing is the morphological analyses. It is divided into three subcategories: tokenization, filtering and stemming [3].

A. **TOKENIZATION:** Text Mining requires the words and the endings of a document. Finding words and separating them is known as tokenization.

B. **FILTERING:** The next step is filtering of important and relevant words from our list of words which were the output of tokenization. This is also called stop words removal.

C. **STEMMING:** The third step is stemming. Stemming reduces words variants to its root form. Stemming of words increases the recall and precision of the information retrieval in Text Mining. The main idea is to improve recall by automatic handling of word endings by reducing the words to their word roots, at the time of indexing and searching. Stemming is usually done by removing any attached suffixes and prefixes (affixes) from index terms before the actual assignment of the term to the index.

3. CLASSIFICATION

Classification is a supervised learning technique which places the document according to content. Text classification is largely used in libraries. Text classification or Document categorization has several applications such as call center routing, automatic metadata extraction, word sense disambiguation, e-mail forwarding and spam detection, organizing and maintaining large catalogues of Web resources, news articles categorization etc. For text classification many machine learning techniques has been used to evolve rules (which helps to assign particular document to particular category) automatically [1]. Text classification (or text categorization) is the assignment of natural language documents to predefined categories according to their content. Text classification is the act of dividing a set of input documents into two or more classes where each document can be said to belong to one or multiple classes. Huge growth of information flows and especially the explosive growth of Internet promoted growth of automated text classification [4].

4. CLASSIFICATION METHODS

1. Decision Trees

Decision tree methods rebuild the manual categorization of the training documents by constructing well-defined true/false queries in the form of a tree structure where the nodes represent questions and the leaves represent the corresponding category of documents. After having created the tree, a new document can easily be categorized by putting it in the root node of the tree and let it run through the query structure until it reaches a certain leaf. The main advantage of decision trees is the fact that the output tree is easy to interpret even for persons who are not familiar with the details of the model [5].

2. k-Nearest Neighbor

The categorization itself is usually performed by comparing the category frequencies of the k nearest documents (neighbors). The evaluation of the closeness of documents is done by measuring the angle between the two feature vectors or calculating the Euclidean distance between the vectors. In the latter case the feature vectors have to be normalized to length 1 to take into account that the size of the documents (and, thus, the length of the feature vectors) may differ. A doubtless advantage of the k-nearest neighbor method is its simplicity.

3. Bayesian Approaches

There are two groups of Bayesian approaches in document categorization: Naïve [6] and non-naive Bayesian approaches. The naïve part of the former is the assumption of word independence, meaning that the word order is irrelevant and consequently that the presence of one word does not affect the presence or absence of another one. A disadvantage of Bayesian approaches [7] in general is that they can only process binary feature vectors.

4. Neural Networks

Neural networks consist of many individual processing units called as neurons connected by links which have weights that allow neurons to activate other neurons. Different neural network approaches have been applied to document categorization problems. While some of them use the simplest form of neural networks, known as perceptions, which consist only of an input and an output layer, others build more sophisticated neural networks with a hidden layer between the two others. The advantage of neural networks is that they can handle noisy or contradictory data very well. The advantage of the high flexibility of neural networks entails the disadvantage of very high computing costs. Another disadvantage is that neural networks are extremely difficult to understand for an average user [4].

5. Vector-based Methods

There are two types of vector-based methods. The centroid algorithm and support vector machines. One of the simplest categorization methods is the centroid algorithm. During the learning stage only the average feature vector for each category is calculated and set as centroid-vector for the category. A new document is easily categorized by finding the centroid-vector closest to its feature vector. The method is also inappropriate if the number of categories is very large. Support vector machines (SVM) need in addition to positive training documents also a certain number of negative training documents which are untypical for the category considered. An advantage of SVM [8] is its superior runtime-behavior during the categorization of new documents because only one dot product per new document has to be computed. A disadvantage is the fact that a document could be assigned to several categories because the similarity is typically calculated individually for each category.

5. PERFORMANCE EVALUATION

- ➢ **Precision**: exactness – what % of tuples that the classifier labeled as positive are actually positive
- ➢ Precision = TP/(TP+FP)
- ➢ **Recall:** completeness – what % of positive tuples did the classifier label as positive?

  Recall = TP/(TP+FN)
- ➢ Perfect score is 1.0.
- ➢ Inverse relationship between precision & recall.
- ➢ F **measure (**F1 or F**-score)**: harmonic mean of precision and recall,

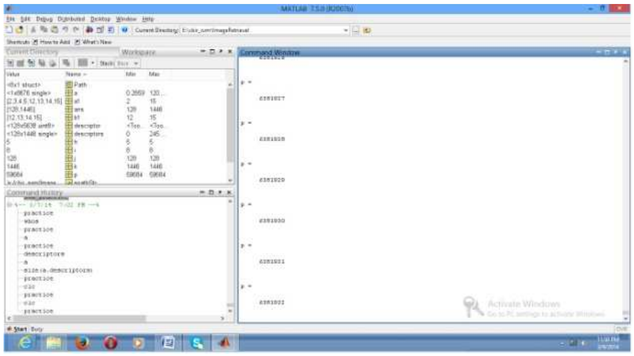
  F=2 × (precision × recall) / (precision + recall)

6. IMPLEMENTTION TOOLS

MATLAB (matrix laboratory) is a fourth-generation high-level programming language and interactive environment for numerical computation, visualization and programming. MATLAB is developed by MathWorks. It allows matrix manipulations; plotting of functions and data; implementation of algorithms; creation of user interfaces; interfacing with programs written in other languages, including C, C++, Java, and Fortran; analyze data; develop algorithms; and create models and applications. It has numerous built-in commands and math functions that help you in mathematical calculations, generating plots and performing numerical methods. MATLAB's Power of Computational Mathematics, MATLAB is used in every facet of computational mathematics. Following are some commonly used mathematical calculations where it is used most commonly:

1. Dealing with Matrices and Arrays
2. 2-D and 3-D Plotting and graphics
3. Linear Algebra
4. Algebraic Equations
5. Non-linear Functions
6. Statistics
7. Data Analysis
8. Calculus and Differential Equations
9. Numerical Calculations
10. Integration
11. Transforms
12. Curve Fitting
13. Various other special functions

MATLAB is a high-performance, efficient and interactive language for technical computing environment. It integrates Computation, visualization, graphical, processing and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical syntactic notation and graphical form. Typical uses include mathematical matrix form and other computation algorithm development Data acquisition Modeling, image processing, Data processing, simulation, and prototyping Data analysis, exploration, and visualization Scientific and engineering drawing and graphics Application development, including graphical user interface building MATLAB(A Technical Computing Tool) is an interactive programming tool whose basic data element is an array (Matrix form) in different dimensional scheme, that does not require to specify dimensioning. This allows you to solve many technical computing problems in different format, especially those with matrix and vector formulations, in a small fraction of the time it would take to write a program in a specific scalar non interactive language like as C or FORTRAN. The name MATLAB is stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK and many other technical projects. Today, MATLAB engines enable to incorporate the LAPACK libraries, embedding the state of the art in software for matrix computation and programming.



```
practice
whos
practice
a
practice
descriptors
a
size(a,descriptors)
practice
clc
practice
clc
practice
```

Figure 1: MATLAB Command Window

MATLAB has evolved over many periods of years with different inputs from many more users. In university research environments, it is the standard and efficient instructional tool for introductory and advanced courses in mathematics, engineering, and medical science. In engineering industry, MATLAB is the tool of choice for better high-productivity research, development, proactive and analysis. MATLAB provide basic features a family of add-on application-specific solutions called toolboxes. Very most important to most and licensed users of MATLAB, toolboxes allow you to learn and apply specialized computing technology. Basically, Toolboxes are comprehensive collections of various MATLAB functions (M-files) and MEX file which is extends the MATLAB environment to solve particular classes of technical computing problems.

7. EXPECTED OUTCOMES

The proposed text mining algorithm is a replacement for conventional text mining approach. Conventional text mining approach is a mature way to use the correlations of features in the text for mining. Only when the large-scale database of texts is available in the dataset, the proposed scheme can exploit the correlations of external text and significantly reduce false rate of text data.

8. REFERENCES

[1] Yuefeng Li, Libiao Zhang, Yue Xu, Yiyu Yao, Raymond Y.K. Lau and Yutong Wu, "Enhancing Binary Classification by Modeling Uncertain Boundary in Three-Way Decisions", JOURNAL OF IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, 2017.

[2] Y. Li, A. Algarni, and N. Zhong, "Mining positive and negative patterns for relevance feature discovery," in Proceedings of KDD'10, 2010, pp. 753–762.

[3] F. Sebastiani, "Machine learning in automated text categorization, "ACM Computing Surveys, vol. 34, no. 1, pp. 1–47, 2002.

[4] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in Proceedings of 11th conference on Uncertainty in Artificial Intelligence, 1995, pp. 338–345.

[5] T. Joachims, "Transductive inference for text classification using support vector machines," in ICML, 1999, pp. 200–209.

[6] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu, "Building text classifiers using positive and unlabeled examples," in Proceedings of ICDM'03, 2003, pp. 179–186.

[7] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with naıve bayes," Expert Systems with Applications, vol. 36, no. 3, pp. 5432–5435, 2009.

[8] C. C. Aggarwal and C. Zhai, "A survey of text classification algorithms," in Mining text data, Springer, 2012, pp. 163–222.

[9] M. A. Bijaksana, Y. Li, and A. Algarni, "A pattern based two stage text classifier," in Machine Learning and Data Mining in Pattern Recognition, Springer, 2013, pp. 169–182.

[10] L. Zhang, Y. Li, C. Sun, andW. Nadee, "Rough set based approach to text classification," in 2013 IEEE/WIC/ACM International Joint Conferences, vol. 3, 2013, pp. 245–252.

[11] M. Haddoud, A. Mokhtari, T. Lecroq, and S. Abdeddam, "Combining supervised term-weighting metrics for svm text classification with extended term representation," Knowledge and Information Systems, pp. 1–23, 2016.

[12] F. Sebastiani, "Machine learning in automated text categorization," ACM Comput. Surv., vol. 34, no. 1, pp. 1–47, 2002.

[13] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in Proceedings of ECML'98, pp. 137–142, 1998.

[14] C. Manning, P. Raghavan, and H. Sch ¨ utze, Introduction to information retrieval, Cambridge University Press, Cambridge, 2008, vol. 1.

[15] T. Joachims, "A probabilistic analysis of the rocchio algorithm with tfidf for text categorization," in Proceedings of ICML'97, 1997, pp. 143–151.

[16] J. Nam, J. Kim, E. L. Mencia, I. Gurevych, and J. Furnkranz, "Large-scale multi-label text classification - revisiting neural networks, "in Proceedings of ECML PKDD 2014, 2014, pp. 437–452.

[17] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in Proceedings of AAAI'15, 2015, pp. 2267–2273.

[18] A. Schwering and M. Raubal, Spatial relations for semantic similarity measurement, Springer, 2005.

[19] L. Zhang, Y. Li, Y. Xu, D. Tjondronegoro, and C. Sun, "Centroid training to achieve effective text classification," in 2014 International Conference on Data Science and Advanced Analytics, 2014, pp. 406–412.

[20] T. Joachims, "A support vector method for multivariate performance measures," in Proceedings of ICML'05, 2005, pp. 377–384.

[21] C. J. Burges, "A tutorial on support vector machines for pattern recognition," Data mining and knowledge discovery, vol. 2, no. 2, pp. 121–167, 1998.

[22] Z. Pawlak, "Rough sets, decision algorithms and bayes' theorem,"European Journal of Operational Research, vol. 136, no. 1, pp. 181–189, 2002.

[23] Y. Yao, "Three-way decisions with probabilistic rough sets," Information Sciences, vol. 180, no. 3, pp. 341–353, 2010.

[24] G. Forman, "An extensive empirical study of feature selection metrics for text classification," The Journal of machine learning research, vol. 3, pp. 1289–1305, 2003.