Implementation of Various Classifiers For Text Mining And Text Categorization

Veeraganti Soumya
Asso. Prof.
Department of CSE
Siddhartha Institute of Technology and Sciences
Narapally, Hyderabad, Telangana, India

Aseena Shaik Babu
Asso. Prof.
Department of CSE
Siddhartha Institute of Technology and Sciences
Narapally, Hyderabad, Telangana, India

***Abstract*- Text data mining has gotten a lot of attention recently, and text categorization is one of the most interesting disciplines. Because of the increasing increase of textual documents, it has grown in popularity. These materials are related with a limited number of categories, such as medical, sports, and Olympic Games. This text categorization can open up a number of possibilities for developing multi-label learning approaches, especially for text-based data. One of the important components employed by automatic text classification is text data mining, which is the process of uncovering useful learning patterns from text-based information. It is accomplished through the development of new machine learning techniques. The ML framework, in any case, produces less expressivity. This ML framework is implemented using a Train-Test scenario. If the present framework is found to be insufficient, the Train-Test-Retrain process is created, which is a difficult and time-consuming process. In this study, we compared the performance of three classifiers for a text mining dataset: Naive Bayes, Decision Tree J48, and Multi-Layer Perceptron (MLP). Simulation is used to determine the performance of all three classifiers, and the experimental findings are shown. It is clear from the data that the decision Tree J48 classifier outperforms the other two classifiers.**

Key words: Text categorization, Text data mining, Multi-Layer Perceptron (MLP), Decision Tree J48, Naive Bayes and Train- Test scenario

1.INTRODUCTION

Text mining is the technique of obtaining high-quality data from text, also known as text data mining, and is also referred to as content analysis. High-quality data is frequently inferred by imagining patterns and employing techniques such as statistical pattern learning. In general, text data mining entails organizing the content of the information (commonly parsing, with the expansion of some inferred phonetic features and the expulsion of others, and subsequent inclusion into a dataset), extracting patterns from the organized information, and finally assessing and understanding the result. The term 'high quality' in text data mining referred to a combination of application, distinctiveness, and appeal. Information extraction, text summarization, text clustering, text classification, construction of granular taxonomies, relation modelling of entities (that is, learning relations among named things), and sentiment analytics are all common text data mining tasks. Data recovery, lexical inquiry to consider word frequency distributions, pattern recognition, labeling/annotation, information extraction, and data mining methodologies combining connection and link investigation, representation, and predictive investigation are all part of text analysis. The main goal is to apply natural language processing (NLP) and analysis methodologies to translate text into valuable information for investigation.

1.1 Text analytics

Text analysis refers to a set of linguistic, quantitative, and machine learning methodologies for framing and structuring the data content of textual data sources in order to get commercial knowledge, exploratory data exploration, study, or analysis [1]. This word is typically interchangeable with text data mining; in fact, Ronen Feldman changed a 2000 picture of 'text data mining' [2] to depict 'text analysis' [3] in 2004. The latter phrase is now more widely used in a corporate context, whereas the term 'text data mining' was used in a handful of the previous application domains in the 1980s, most notably government intelligence and life-sciences research[4]. Text analysis also refers to the use of text analysis to respond to commercial issues, whether alone or in conjunction with the investigation and assessment of handled, numerical data. According to the adage, 80 percent of commercial information begins in an unstructured state, essentially text [5]. These methods and processes identify and update knowledge factors, commercial principles, and connections in a text-based style that is resistant to mechanized operation.

2. RELATED WORK

Decision tree approaches [8] reproduce non-automatic requests for training data in the form of a tree-based structure, with a node representing the queries and a leaf representing the specific dataset type. 'Over-fitting' is the negative rating in decision tree approach. It is simple to use and beneficial. There are two types of Bayesian techniques: Nave and Non-Nave Bayesian methodologies. The naive techniques are divided into two categories: multivariate and multinomial. Both groups are based on the dispersion of terms in data documents [10]. An N-gram is a never-ending sequence of n characters that make up a long section of a material [11]. The most forward-thinking N-grams are kept. This method generates a basic number of parts that appear differently in relation to text analysis by taking into account word separators and punctuation marks; additionally, it is extremely tolerant of spelling errors and is unaffected by all progressions models based on discrete letters (Chinese language, DNA orders...). Its success in dialect recognition can be attributed to its content classification [12].

3. CLASSIFIERS

The text dataset is analyzed using three different types of classifiers:

- Naïve Bayes
- Decision Tree J48
- Multi-Layer Perceptron (MLP)

3.1 Naive Bayes

One of the most fundamental ways to building classifiers is Naive Bayes: frames that allocate labels for class to problem cases, represented as vectors of attribute measurements, with labels extracted from a constrained set. It is a combination of procedures, not a single technique, that is used to train these classifiers, as in a typical rile: Given the class variable, complete nave Bayes classifiers assume that the measure of a given characteristic is independent of the measure of other attributes. For example, if a fruit is red, round, and around 10 cm in diameter, it is likely to be considered an apple. The nave Bayes classifier considers each of these characteristics independently in determining whether or not this organic product is an apple, disregarding any possible correlations between the shading, roundness, and diameter parameters. In the context of supervised learning, naive Bayes classifiers can be efficiently taught for a few types of probability frameworks. The maximum likelihood technique is used in many real-time applications for parameter approximation for naive Bayes frameworks. Alternatively, one can use the naïve Bayes framework without using any Bayesian procedures or taking into account Bayesian likelihood. Despite their naïve model and clearly warped assumptions, naive Bayes classifiers have performed admirably in a variety of uncertain real-time situations. An analysis of the Bayesian characterization issue in 2004 revealed that the clearly implausible feasibility of naïve Bayes classifiers is motivated by solid hypothetical reasons [5]. In 2006, a comprehension-based comparison of Bayes classification with other classification approaches revealed that various methodologies, such as random forest [6,], outperform Bayes classification.

3.2 Decision Tree J48

J48 is the most well-known decision tree-based categorization technique, as the name implies. Initially, it classifies photos based on their properties and creates a tree structure. The tree structure is explained in a straightforward manner. The Decision Tree J48 is an extension of the ID3 algorithm, and it is used mostly for its straightforward methodology for locating concealed pixels in images. The photos were grouped in a leaf structure and trimmed during categorization. These pixels were classified by labelling, and the information on each pixel was extracted and tested. The perfect pixel is chosen from the resultant pixel, and these classifiers are useful for dealing with both discrete and continuous values.

3.3Multi-Layer Perception (MLP)

The Multi-Layer Perception (MLP) is a feed forward artificial neural network (ANN) technique in which categorization is done by mapping the input images. The mapping is based on the training and testing dataset's features. The mapping is done using the back-propagation approach in this case. By doing so, the MLP creates a directed graph of nodes that are subsequently connected to one another. A non-linear activation function is provided for each node in the graph. MLP's datasets were also trained using supervised learning techniques, which are useful for *categorizing* non-linear data. For resolving the complications, it uses a stochastic fitness function.

4. EXPERIMENTAL RESULTS

The 'Letter dataset' is used in this paper for text analysis and text categorization. The classification is carried out using the Weka tool, which employs three classifiers: Naive Bayes, Decision Tree J48, and Multi-Layer Perceptron (MLP). The results of each

classifier are presented in this section in great detail.
Time taken to build model: 1.49 seconds

Correctly Classified Instances 64.405 %	12881
Incorrectly Classified Instances 35.595 %	7119
Kappa statistic	0.6298
Mean absolute error	0.032
Root mean squared error	0.1383

Detailed Accuracy by Class

4.2 Results of Decision Tree J48 classifier

Correctly Classified Instances	19269	96.345 %
Incorrectly Classified Instances	731	3.655 %
Kappa statistic	0.962	
Mean absolute error	0.0044	
Root mean squared error	0.0471	
Relative absolute error	5.9985 %	
Root relative squared error	24.4918 %	
Total Number of Instances	20000	

4.3 Results of Multilayer perceptron

Time taken to build model: 344.96 seconds

=== Evaluation on training set ===

Correctly Classified Instances	16491	82.455 %
Incorrectly Classified Instances	3509	17.545 %
Kappa statistic	0.8175	
Mean absolute error	0.0153	
Root mean squared error	0.1093	
Relative absolute error	20.6997 %	
Root relative squared error	56.8155 %	

4.1 Results of Naïve Bays classifier

=== Evaluation on training set ===

Relative absolute error	43.3058 %
Root relative squared error	71.9279 %
Total Number of Instances	20000

Total Number of Instances 20000

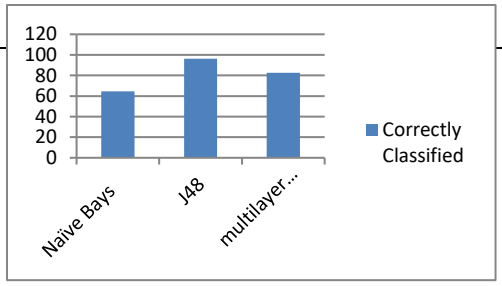

Figure 1: Comparison of classifiers

5. CONCLUSION

The 'Letter dataset' is used in this work for text analysis and categorization. For text datasets, a performance analysis is undertaken using this dataset for three classifiers: Naive Bayes, Decision Tree J48, and Multi-Layer Perceptron (MLP). Weka is used to model these, and the results are tabulated and compared. It is obvious from the findings that the Decision Tree J48 classifier performs better in terms of accuracy and precision rate.

REFERENCES

[1]Bruno Trstenjaka, SasaMikacb, DzenanaDonkocm, "KNN with TF-IDF Based Framework for Text Categorization", 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 69: 1356 – 1364, 2014.

[2] Michal Hrala and Pavel Kral, "Evaluation of the Document Classification Approaches", doi: 10.1007/978-3-319-00969-8_86, 2013.

[3] Ashis Kumar Mandal1and Rikta Sen, "Supervised Learning Methods for Bangla Web Document Categorization", International Journal of Artificial Intelligence & Applications (IJAIA), 5(5), 2014.

[4] Erlin, Unang Rio, "Text Message Categorization of Collaborative Learning Skills in Online Discussion Using

Support Vector Machine", 2013 International Conference on Computer, Control, Informatics and Its Applications , 2013.
[5] Joachims, T, "Transductive inference for text classification using support vector machines", Proceedings of ICML-99, 16th International Conference on Machine Learning, eds. Morgan Kaufmann Publishers, San Francisco, US: Bled, SL, 1999, pp. 200–209
[6] Addis, A., "Study and Development of Novel Techniques for Hierarchical Text Categorization", PhD Thesis Electrical and Electronic Engineering Dept., University of Cagliari, Italy, 2010.
[7] Feldman, R & Sanger, J, "The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data", Cambridge University Press New York, 2006.
[8] D. E. Johnson, F. J. Oles, T. Zhang, T. Goetz, "A decision-tree-based symbolic rule induction system for text categorization", IBM Systems Journal, 2002.
[9] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer, "KNN Model-Based Approach in Classification doi: 10.1007/978-3-540-39964-3_62: 986-996, 2003.
[10] C. C. Aggarwal, and C. Zhai, "Mining text data", doi: 10.1007/978-1-4614-3223-4, 2012
[11] HamoodAlshalabi, Sabrina Tiun, Nazlia Omar, Mohammed Albared, "Experiments on the Use of Feature Selection and Machine Learning Methods in Automatic Malay Text Categorization", 4th International Conference on Electrical Engineering and Informatics , pp. 734-739, 2013.
[12] Z. Wei, D. Miao, J.-H. Chauchat, "N-grams based feature selection and text representation for Chinese Text Classification", International Journal of Computational Intelligence Systems, 2 (4), 2009, pp. 365-374.
[13] T. Joachims, "A statistical learning model of text classification for support vector machines", in: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2001, pp. 128- 136.
[14] Forman, G. "An Experimental Study of Feature Selection Metrics for Text Categorization", Journal of Machine Learning Research, 2003, pp. 1289-1305.
[15] Chaitrali S. Dangare and Sulabha S. Apte, "Improved Study of Heart Disease Prediction System using Data Mining Classification Techniques", International Journal of Computer Applications, 47(10), 2012.