ANALYSIS ON VARIOUS ROBUST CLASSIFIERS ON MULTI-DIMENSIONAL CLASSIFICATION

Niraj Patel
Department of Computer Science, S. K. Somaiya Degree College of Arts, Science and Commerce, India

Abstract

The study focuses exclusively on the analysis of new text classification methods in this publication. The objective of text classification is to automatically categorise a set of papers from a preset list into categories. The research is based on a combination of data collection and data mining models. The main characteristics of the technologies concerned are outlined in this research. This research uses three algorithms for the classification of papers into distinct categories, which are trained on two independent datasets. Regarding the above classification algorithms, due to their simplicity, Nave Bayes is likely to serve as a text classification model.

Keywords:

Information Retrieval, Vector Space Model, Natural Language Processing, Classification

1. INTRODUCTION

Recently, studies on text mining have become more important since a growing quantity of electronic documents from a range of sources are available. This includes information which is unstructured or semi-structured. The key objective of text mining is the collection of information from textual resources by users, such as recovery, categorization (supervised, unattended, and semi-supervised) and summary. NLP, data mining, and machine learning approaches collaborate to automatically classify documents and to find patterns of different sorts of content [5].

Text Classification (TC) is an important component of text mining, which is seen as the manual construction, using knowledge-engineering techniques, of an expert knowledge converts documents into a set of categories [6] by way of a manual definition of a set of logical rules, also known as training. For instance, each incoming item should instantly identify a theme, such as sports, politics, or business. The task of data mining begins with training packages $D=(d_1.....d_n)$ comprising documents previously tagged C_1, C_2 (e.g. sports, politics). The objective then consists of constructing a classification model that can allocate a new domain document to the correct class.

Two phases of text classification are essentially involved. Stage of training and assessment. As described in the above paragraph, documents are pre-processed during training and trained to create the classifier with a learning algorithm. A classifier test is performed during the testing phase. Many classical learning algorithms for data training are available, such as Decision-Bodies, Nave Bayes, SVM, K-Nearest Neighbor (kNN), Neural Network (NNET), etc.

During this study, the study analysed the problem of text classification, using Naive Bayes Classification, Vector Space Models for Text Classification, and new Stanford Tagger methods for text classification. The new categories of news documents are classified into three groups. In order to analyze the effectiveness of every algorithm [7]-[10], the study attempted to compare the efficiency and accuracy of the algorithms. Two separate datasets, 20 Newsgroup and New Dataset, have been researched for five categories.

2. RELATED WORK

The method in [1] text classifying and classifying is directly linked with our work. The principal aim of text mining is to enable users to extract information from textual resources and to work together on operations like retrieval, classification (supervised, unattended and semi-supervised) and summarization, NLP, data-mining and machine-learning, in order to categorise and discover patterns in various types of documentation. For their efficiency, they compared several text classifications.

Another study paper connected to my investigation [2] states that four distinct methods for the classification of documents are being investigated: the nave Bayes classifier, neighbouring classifier, decision trees, and the subspace method. These were individually and in combination with seven class news categories from Yahoo. The study of three ways of classification: simple vote, dynamic selection of classifier and adaptive classifier combination. Experimental results show that the naive Bayes classifier and the subspace approach overpower the two other data set classifiers. In contrast to the best individual classification system, several classifying combinations did not always increase the classification accuracy. The adaptive classifier combination method introduced performed best among the three alternative combination methods.

In [3], Automatic Text Classification is an automated, semi-monitored learning job that automatically assigns a specific document to a number of pre-defined categories based upon textual content and extracted attributes. It also has a tight connection with my studies. This study discusses the general strategy for the automatic classification of texts, which involves pre-processing, selecting functions using various strategies for statistics and semantics, and modelling using appropriate machine-learning algorithms. This paper discusses a number of the key problems in the automatic classification of text, such as the treatment of unstructured texts, the use of a large number of attributes, the success of purely statistical methods of preprocessing for semantic and natural language processing methods, lack of materials, and the choice of an appropriate machine learning process.

In another research on the vector spatial model [4], it was stated that different approaches to the vector spacious model were used to calculate the similarity of the search engine hits and, above all, that the study would lead to an understanding of the problems and issues with the use of the vector spatial model in information collection.

3. METHODOLOGY

The text classification process is described in Fig.1. This technique details documents, preprocessing, indexing, extraction, algorithms for classification and measurement of performance, whereas in the second section, text classification models employed in the study are explained.

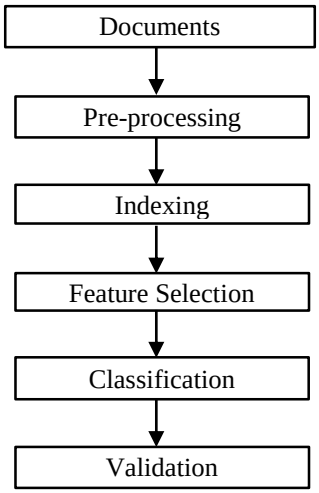

Fig.1. Document Classification Process

3.1 DOCUMENTS COLLECTION

This is the initial step in the classification process in which the study collects the various types of documents (formats) such as .html, .pdf, .doc, web content, etc.

3.2 PRE-PROCESSING

The initial phase of pre-processing is to provide the texts in clear word format. There are several features in the documents prepared for the next step in text classification. The actions taken are usually:

- *Tokenization*: A document is viewed as a string, then divided into a token list.
- *Removing Stop Words*: Stop words such as, a, and... occur often, thus the minor words need to be removed. Remove stop words.
- *Stemming Word*: Use the stemming method, which turns many word forms into the same canonical form. This step is the way to link tokens with their base forms.

3.3 INDEXING

The presentation of documents is one of the techniques for pre-processing used to minimise and facilitate handling of documents, and the document needs to be converted from the full text version into a document vector. The Vector space model is the most often used document display; documents are represented by word vectors. One generally has a collection of documents represented by word for word Matrix documents.

3.4 FEATURE SELECTION

The selection of features to build the vehicle space, improving the scalability, efficiency, and accuracy of the text classification, is performed after preprocessing and indexing the main text classification stage. The primary principle of FS is to select a sub-set of characteristics from the source texts. FS is carried out in accordance with the pre-determined value of the word, by keeping words with the highest mark.

3.5 CLASSIFICATION

Automatic categorization of documents has been noted to be an active concern. Documents can be categorised using three methods – unattended, monitored, and semi-monitored. The task of automated text categorization has been actively investigated for several years, and rapid progress has been made in this area, including the Baysian classification methods for machine learning, Decision Tree, K-nearest neighbour (KNN), SVMs, and Neural Networks.

3.5.1 Naïve Bayes:

Naive Bayes Classifiers (NB) can be trained in a supervised learning environment in certain kinds of probability models very efficiently. Naive Bayes is a basic way to build classification models that assign class labels, represented as vectors of function values, to problem cases in which class labels are selected from a certain finite set. There is not one single method for the development of these classifications, but an algorithm family based on a common principle: all Bayes nave classifiers assume that, given the class variable, the value of a particular feature is independent of the value of every other feature. For example, if an apple is red, round and 3 inches in diameter, the fruit may be considered an apple. A naive classification in Bayes assumes that each of these features contributes, independently of the possible relationships between colour, roundness and diameter, to the likelihood that this fruit is an apple.

The advantage of naive Bayes is that the only thing needed to calculate the necessary parameters for classification is a modest amount of training data. In fact, a Naive Bayes classifier converges quicker than discriminatory modelling, such as logistic regression, therefore less training data is necessary. If the NB assumption is conditional independence. Even when the NB assumption does not take place, in practice, an NB classification typically still works very effectively.

Step 1: Using data preprocessing procedures, such as stopping word removal and stemming, present in the test document.

Step 2: Tokenize your data and save your words in real-time memory along with your category.

Step 3: Checking the word probability stored in the data base for every single word from the test document. If the word occurs in this category, the likelihood is added and repeated for all the terms in that test text.

Step 4: Calculation of the probabilities for each category, and accurate matching is the most likely.

3.5.2 Vector Space Model:

The vector space model or term vector model is an algebraic paradigm for representation as vectors of identifiers of text documents, such as index terms, for example. It is used to filter information, retrieve information, and index and rank relevant information. The SMART Information Retrieval System was its initial application. The technique for vector space can be split into

three stages. The first stage is the indexing of a document in which terms of content are extracted from the text. The second phase consists of weighing the indexed terms so that the documents relevant to the user are retrieved better. The last step classifies the document according to a similarity measure in respect of the query. Documents are sometimes considered to be a bag of words or terms. Each paper is shown as a vector. The phrase weight is not more than 0 or 1, however. A number of variants in TF or the TF-IDF system are used for each word weight.

Step 1: Applying the data available in the test document to pre-processing procedures. In other words, stop the removal of words, strokes and tokenization.

Step 2: Compare words in the category from the test document to those in the database.

Step 3: Add the weights of the words in the test document.

Step 4: Compute for each category the total weights and match the category with the highest weight.

3.5.3 Stanford Tagger:

The Vector A Part of a Spell Tagger (POS Tagger) is a computer software which uses more fine-grained POS tags, for instance, the words and portions of a spoken language (and tokens), such as a noun, verb, adjective, etc. The process of marking a word in a text (corpus of words) according to a particular section of the speech, both in relation to its definition and its context — i.e., in relation to adjacent and related words in a sentence, sentence or paragraph — is in corpus linguistics, part-to-language tagging (POS tagging or POST), which is also called grammatical or word category disambiguation. POS tagging is now done manually, utilising algorithms that combine discreet phrases as well as occulted parts of speech according to a range of descriptive tags in the computational linguistic environment.

Step 1: Find out each test file's phrase, noun, verb, and adjective.

Step 2: The Stanford Tagger module is used to perform this.

Step 3: Compare this in the database with the nouns, verbs, and adjectives.

Step 4: Whatever category is the most appropriate fit for this particular noun, verb, and adjective.

4. RESULTS AND DISCUSSION

The experimental evaluation of classifiers usually attempts to measure the effectiveness of a classifier rather than focusing on the issues of efficiency, i.e. its capacity to make the appropriate predicting on classification. Many measures, such as accuracy and reminders, failure, error and accuracy, have been utilised.

4.1 DATASET SOURCES

Approximately 20 000 newsgroup documents are collected in the 20 newsgroups, divided up (almost) uniformly among the 20 newsgroups. It was collected by Ken Lang initially. The collection of 20 newsgroups is a popular data set for testing in text applications, such as text classification and text clustering. The data is arranged into 20 newsgroups, each with a different subject. There is also a close relationship between some of the newsgroups, while others are not really connected.

The other new data package is a compilation of some 500 news items from various newspapers divided into five separate newsgroups, including business, nation, sports, technology, and world news. It was first gathered for this project's training and testing purposes. Some items are closely linked, while others, as in the 20 newsgroup, are very unrelated.

This research used around nine different newsgroups from 20 datasets of newsgroups with over 1200 training articles and news from a new five-category dataset. These classifiers are tested using fifty random documents (news), which are randomly selected from the web and do not relate to training information.

The quality of a data source has a major impact on the performance of a classification system. Important and redundant data characteristics not only increase mining costs but also, in some situations, degrade the quality of the result.

Our research uses the first data set of 20 newsgroups, which was downloaded from the 20 newsgroups website. The HTML news studies are pre-processed

- Parsing of the document (remove headers and tags in the HTML files) and
- Removal of low-frequency words and stop words.

A total of over 1200 materials from nine distinct courses and a series of exam data were used in the study.

The second dataset used in our research is the news datasets downloaded for this investigation from several news websites, for example, Jagran and Herald. In total, around 500 materials from 5 separate training classes and a set of test data as described above were used for testing in the study.

The study compared the three algorithms (Naive Bayes, Vector Space Model and Stanford Tagger) on our testing data sets using two sets of training texts. In this study, the result is still considered as a gold standard by humans to allow the study to compare the results of classification with the algorithm and to observe how it acts.

Table.1. Comparative results with 20 Newsgroups dataset

Models Metrics	Naive Bayes	Vector Space Model	Stanford Tagger
Accuracy	95.91	96.28	94.21
Precision	93.21	93.45	91.99
Recall	94.92	94.59	92.14
F-measure	93.27	93.58	91.91

Table.2. Comparative results with News dataset

Models Metrics	Naive Bayes	Vector Space Model	Stanford Tagger
Accuracy	98.49	98.12	96.38
Precision	95.60	95.35	94.11
Recall	96.77	97.10	94.26
F-measure	95.73	95.42	94.02

If your training package is small, high bias/low variance classification devices have an advantage compared with low

bias/high variance classifiers, since these classifications fit. But low bias/high variance classifiers begin to win when your workout is growing, because high bias classifiers are not sufficiently capable of generating exact models. Long documents are poorly represented as they have weak similarity values with regard to the vector space model of text classification. Searching keywords must match document terms accurately; word substrings may lead to a false positive match. Documents with a similar background and a different term vocabulary will not be linked, resulting in a false negative match. In the vector space format, the order in which the terms appear inside the document is lost. Theoretically, the terms are independent statistically. Some terms often exist in papers that are less important, because superfluous weight increases and can lead to inconsistencies.

In terms of precision and computing efficiency, Nave Bayes seems to be the top classifier compared to numerous common classifiers (Table.1 and Table.2). VSM is better with the data set of the newsgroup, as the data set is relatively short and the features are less irrelevant (Table.1).

5. CONCLUSION

Text Classification is an essential field of application in text mining because it is costly and time-consuming to categorise millions of text documents manually. Automatic classification of text is therefore created with pre-classified sample papers, which are far more accurate and time-efficient than manual categorization of text. When the data entered into the classifier is less noisy, efficient outcomes are obtained.

The study used 3 different text classification models, namely, the vector space model (VSM), the Naive Bayes Classifier (NB) model and the Stanford Tagger model, consisting of comparatively less data and evaluating the results in subsets of datasets. The study also evaluated three text classification models. The study also took account and assessed the results of all three methodologies, which the study considers as gold standards, by human interpretation. Based on this assessment, the study revealed cases when the NB classifier approach functioned far better than the two remaining classifiers.

For the investigation of the more controlled method in the text categorization of diverse datasets, future work in this field should be addressed. The effectiveness of the above algorithms is compared. Compared the controlled algorithms for text classification to the semi-monitored and unattended algorithms. Use the modules of the previously studied algorithm to identify the most efficient algorithm. More details on the usage of the text classification Stanford Tagger for more tags and the use of such methods for word sense disambiguation.

REFERENCES

[1] Mucahit Altintas and Cuneyd Tantug, "Machine Learning Based Ticket Classification in Issue Tracking Systems", *Proceedings of International Conference on Artificial Intelligence and Computer Science*, pp. 1-6, 2014.

[2] S. Silva, R. Pereira and R. Ribeiro, "Machine Learning in Incident Categorization Automation", *Proceedings of IEEE 13th Iberian Conference on Information Systems and Technologies*, pp. 1-6, 2008.

[3] S.P. Paramesh and K.S. Shreedhara, "Automated IT Service Desk Systems Using Machine Learning Techniques", *Proceedings of IEEE International Conference on Data Analytics and Learning*, pp. 331-346, 2018.

[4] M.M. Mironczuk and J. Protasiewicz, "A Recent Overview of the State-of-the-Art Elements of Text Classification", *Expert Systems with Applications*, Vol. 106, pp. 36-54, 2018.

[5] L. Breiman, "Bagging Predictors", *Machine Learning*, Vol. 24, No. 2, pp. 123-140, 1996.

[6] T. Dietterich, "Ensemble Methods in Machine Learning", *Proceedings of International Workshop on Multiple Classifier Systems*, pp. 1-15, 2000.

[7] M. Ikonomakis, S. Kotsiantis and V. Tampakas, "Text Classification using Machine Learning Techniques", *WSEAS Transactions on Computers*, Vol. 4, No. 8, pp. 966-974, 2005.

[8] Yashima Ahuja and Sumit Kumar Yadav, "Multiclass Classification and Support Vector Machine", *Global Journal of Computer Science and Technology Interdisciplinary*, Vol. 12, No. 11, pp. 14-20, 2012.

[9] A. Sharma and S. Dey, "A boosted SVM based Ensemble Classifier for Sentiment Analysis of Online Reviews", *ACM SIGAPP Applied Computing Review*, Vol. 13, No. 4, pp. 43-52, 2013.

[10] Nikolay Butakov Maxim Petrov and Anton Radice, "Multitenant Approach to Crawling of Online Social Networks", *Procedia Computer Science*, Vol. 101, pp. 115-124, 2016.