RESEARCH ARTICLE OPEN ACCESS

Survey Paper on Document Classification and Classifiers

Upendra Singh [1], Saqib Hasan [2]
UG Students [1] & [2]
Department of Computer Science and Engineering
Madan Mohan Malaviya University of Technology
Gorakhpur - 273010
UP - India

ABSTRACT
The rapid growth of World Wide Web has rendered the document classification by humans infeasible which has given impetus to the techniques like Data mining, NLP and Machine Learning for automatic classification of textual documents. With the high availability of information from diverse sources, classification tasks have attained paramount importance. Automated text classification has been considered as a vital method to manage and process a vast amount of documents in digital forms. This paper provides an insight into text classification process, its phases and various classifiers. It also aims at comparing and contrasting various available classifiers on the basis of few criteria like time complexity and performance.
Keywords:- Data Mining, Natural Language Processing, Classifier, Text classification, Machine Learning.

I. INTRODUCTION

With the increasing availability of digital documents from diverse sources, text classification is gaining popularity day in and day out. There is a mushroom growth of digital data made available in the last few years, data discovery and data mining have worked together to extract meaningful data into useful information and knowledge [10]. Text mining refers to the process of deriving high quality information from text. It is conducive in utilizing information contained in textual documents in various ways including discovery of patterns, association among entities etc. and this is done with the amalgamation of NLP(Natural Language Processing), Data Mining and Machine learning techniques.

Infeasibility of human beings to go through all the available documents to find the document of interest precipitated the rise of document classification. Automatically categorizing documents could provide people a significant ease in this realm. Text classification assigns documents one or more predefined categories. The notion of classification is very general and has many applications within and beyond information retrieval (IR). For instance, text classification finds its application in automatic spam detection, sentiment analysis, automatic detection of obscenity, personal email sorting and Topic specific or Vertical Searches. An example of classification would be automatically labeling news stories with subjects like “business”, “entertainment”, “sports” etc.

2.1 Document Collection

II. CLASSIFICATION PROCESS

From the perspective of automatic text classification systems, classification task can be sequenced

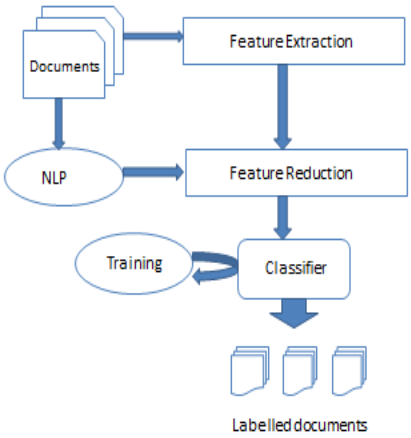

Fig 2.1 Steps of Text Classification

Text classification starts with this step of collecting various types of documents including different formats like html, .pdf, .doc, web content etc.

2.2 Tokenization

Tokenization, when applied to documents, is the process of substituting a sensitive data element with a non-sensitive equivalent, referred as token that has no extrinsic or exploitable meaning or value. A document is considered as a string, and then partitioned into a list of tokens. Stop words such as "the", "a", "and", etc. are frequently occurring; therefore the insignificant words need to be removed.

2.3 Feature Extraction

Feature extraction is the process of selecting a subset of the terms occurring in the training set and using only this subset as features in text classification. Feature extraction serves two main purposes. First, it makes training and applying a classifier more efficient by decreasing the size of the effective vocabulary. Second, feature selection often increases classification accuracy by eliminating noise features. A noise feature is one that, when included in the document representation, increases the classification error on new data. Additional features can be mined from the classifiable text; however nature of such features should be highly dependent on the nature of classification to be carried out. If web sites need to be separated into spam and non-spam websites, then the word frequency distribution or the ontology is of little use for the classification, because of widespread tactics by the spammers to copy and paste mixture of texts from legitimate web sites in creation of their spam web sites [2].

2.4 Natural Language Processing

Feature extraction and reduction phases of text classification process are performed with the help of Natural Language Processing techniques. Linguistic features can be extracted from texts and used as part of their feature vectors [3]. For example parts of the text that are written in direct speech, use of different types of declinations, length of sentences, proportions of different parts of speech in sentences (such as noun phrases, preposition phrases or verb phrases) can all be detected and used as a feature vector or in addition to word frequency feature vector [4].

2.5 Feature Reduction

Feature reduction a.k.a. Dimensionality reduction is about transforming data of very high dimensionality into data of much lower dimensionality such that each of the lower dimensions manifest much more information. The computational complexity of any operations with such feature vectors will be proportional to the size of the feature vector (Yang & Pedersen, 1997), so any methods that reduce the size of the feature vector while not significantly impacting the classification performance are very welcome in any practical application. Additionally, it has been shown that some specific words in specific languages only add noise to the data and removing them from the feature vector actually improves classification performance.

The set of feature reduction operations involves a combination of three general approaches [5]:
1. Stop words;
2. Stemming;
3. Statistical filtering.

Stop words like: "a", "the", "but" are required by the grammar structure of any language but inculcate no meaning. Likewise, stemming converts different word form into similar canonical form. Statistical filtering practices are used to glean those words that have higher statistical significance. Most represented statistical filtering approaches are: odds ratio, mutual information, cross entropy, information gain, weight of evidence, χ 2 test, correlation coefficient [6], conditional mutual information maxmin [8], and conformity/uniformity criteria [7]. In simple terms, most formulas give high scores to words that appear frequently within a category and less frequently outside of a category (conformity) or to the opposite (non-conformity). And additionally higher scores are given to words that appear in most documents of a particular category (uniformity).

2.6 Classification

With each passing day, automatic classification of documents in predefined categories is gaining active attention of many researchers. Supervised, unsupervised and semi supervised are the methods used to classify documents. The last decade has seen the unprecedented and rapid progress in this area, including the machine learning approaches such as Bayesian classifier, Decision Tree, K-nearest neighbor(KNN), Support Vector Machines(SVMs), Neural Networks, Rocchio's.

III. CLASSIFIERS

3.1 K-Nearest Neighbour

K nearest neighbors is an elegant supervised machine learning algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions).K-NN works on a principle that the points (documents) which are close in the space belong to

the same class. The algorithm assimilates all training samples and predicts the response for a new sample by analyzing a certain number (**K**) of the nearest neighbors of the sample by using some similarity measure such as Euclidean distance measure etc., the distance between two neighbors using Euclidean distance can be found using the given formula.

$$Dist(X,Y) = \sqrt{\sum_{i=1}^{D}(Xi - Yi)^2}$$

A major demerit of the similarity measure used in k-NN is that it uses all features in computing distances which degrades its performance. In myriad document data sets, only smaller number of the total vocabulary may be useful in categorizing documents. A probable approach to tackle this problem is to learn weights for different features (or words in document data etc.) [11]. Proposed Weight Adjusted k-Nearest Neighbor (WAKNN) classification algorithm is based on the k-NN classification paradigm which can enhance the performance of text classification [12].

3.2 Support Vector Machine

Initially, Support vector machines (SVM) was developed for building an optimal binary (2-class) classifier but thereafter the technique was extended to regression and clustering problems. The working principle of SVM is to find out a hyper plane (linear/non-linear) which maximizes the margin. Maximizing the margin is equivalent to:

$$\underset{w,b,\zeta_i}{\text{minimize}} \quad \frac{1}{2}w^T w + C(\sum_{i=1}^{N}\zeta_i)$$

$$\text{subject to} \quad y_i(w^T x_i - b) + \zeta_i - 1 \geq 0, \quad 1 \leq i \leq N$$

$$\zeta_i \geq 0, \quad 1 \leq i \leq N$$

SVM is a partial case of kernel-based methods. It binds feature vectors into a higher-dimensional space using a kernel function and builds an optimal linear discriminating function in this space or an optimal hyper-plane that is congruent with the training data. The kernel is not explicitly defined in case of SVM. Instead, a distance between any 2 points in the hyper-space needs to be defined.

The key features of SVMs are the use of kernels, the absence of local minima, the sparseness of the solution and the capacity control obtained by optimizing the margin. Besides the advantages of SVMs - from a practical point of view they have some drawbacks. An important practical question that is not entirely solved, is the selection of the kernel function parameters - for Gaussian kernels the width parameter [sigma] - and the value of [epsilon] in the [epsilon]-insensitive loss function.

3.3 Naïve Bayes

The Naive Bayes classifier is a probabilistic classifier based on Bayes theorem with strong and naïve independence assumptions. It is supposed to be one of the most basic text classification techniques with various applications in email spam detection, personal email sorting, document categorization, sexually explicit content detection, language detection and sentiment detection.

Experiments witness that this algorithm performs well on numeric and textual data. Though it is often outperformed by other techniques such as boosted trees, random forests, Max Entropy, Support Vector Machines etc., Naive Bayes classifier is quite efficient since it is less computationally intensive (in both CPU and memory) and it necessitates a small amount of training data. The assumption of conditional independence is breached by real-world data with highly correlated features thereby degrading its performance.

3.4 Neural Networks

Neural networks can be used to model complex relationships between inputs and outputs to find patterns in data. By using neural networks as a tool, data warehousing firms are gathering information from datasets in the process known as data mining. A neural network classifier is a network of units, where the input units usually represent terms, the output unit(s) represents the category. For classifying a text document, its term weights are assigned to the input units; the activation of these units is propagated forward through the network, and the value that the output unit(s) takes up as a consequence determines the categorization decision.

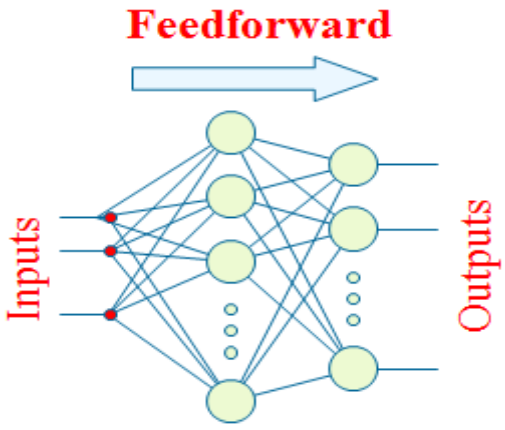

Fig 3.4 Simple Neural Network Demonstration

Suitability for both discrete and continuous data makes neural network a popular choice for text classification purpose.

3.5 Rocchio's Algorithm

The Rocchio's algorithm is based on a method of relevance feedback found in information retrieval systems which stemmed from the SMART Information Retrieval System around the year 1970. In this algorithm, a prototype vector is built for each class. A prototype vector is average vector over all training document vectors that belong to class c_i.

$$C_i = \alpha * centroid_{C_i} - \beta * centroid_{\overline{C_i}}$$

Similarity between text document and each of prototype vectors is determined and text document is assigned to the class having maximum similarity. The algorithm is based on the assumption that most users have a general conception of which documents should be denoted as relevant or non-relevant.

This algorithm is deemed as very fast learner and easy to implement. Although easy to implement, this algorithm suffers from poor classification accuracy. The selection of values for the constants alpha and beta plays a vital role in its performance.

IV. PROPOSED METHODOLOGY

When confronted with a need to build a text classifier, the first question to ask is how much training data is there currently available? None? Very little? Quite a lot? Or a huge amount, growing every day? For many problems and algorithms, hundreds or thousands of examples from each class are required to produce a high performance classifier and many real world contexts involve large sets of categories.

Training a supervised classifier with little data may not turn out beneficial. So it is advisable to cling to a semi-supervised classifier. In case of availability of huge amount of data, it may be best to choose a classifier based on the scalability of training or even runtime efficiency. The general rule of thumb is that each doubling of the training data size produces a linear increase in classifier performance, but with very large amounts of data, the improvement becomes sub-linear.

V. CONCLUSION

Text classification is a widespread domain of research encompassing Data mining, NLP and Machine Learning. It has witnessed much heed owing to the high growth rate of internet and relevance of internet search engines. This review paper circumscribes existing literature and explores the document representation and analysis of feature extraction methods and broaches to different available classifiers. Various methods of classification and feature extraction have been compared and contrasted with all coeval methods based on different parameters like time complexities and performance. It is deemed that no single representation scheme and classifier can be mentioned as a general model for any application. Performance of different algorithms varies according to the data collection. However, SVM with term weighted VSM representation scheme has shown some potential results in the tasks of text classification up to some extent but still universal acceptance of this algorithm remains implausible.

REFERENCES

[1] F. Sebastiani, "Text categorization", Alessandro Zanasi (ed.) Text Mining and its Applications, WIT Press, Southampton, UK, pp. 109-129, 2005.

[2] Fetterly, D., Manasse, M. & Najork, M. (2005). Detecting phrase-level duplication on the world wide web. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. pp. 170-177). : ACM Press, Salvador, Brazil

[3] Hunnisett, D. S. & Teahan, W.J. (2004). Context-based methods for text categorisation. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. pp. 578-579). : ACM Press, Sheffield, United Kingdom

[4] Stamatatos, E., Kokkinakis, G. & Fakotakis, N. (2000). Automatic text categorization in terms of genre and author. *Computational Linguistics*, *26*, pp. 471-495.

[5] Liu, H. & Motoda, H. (1998). Feature Selection for Knowledge Discovery and Data Mining. : Kluwer Academic Publisher.

[6] Ng, H. T., Goh, W. B. & Low, K.L. (1997). Feature selection, perception learning, and a usability case study for text categorization. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. pp. 67-73).

[7] Chen, C., Lee, H. & Hwang, C. (2005). A Hierarchical Neural Network Document Classifier with Linguistic Feature Selection. *Applied Intelligence*, *23*, pp. 277-294.

[8] Wang, G. & Lochovsky, F.H. (2004). Feature selection with conditional mutual information maximin in text categorization. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management* (pp. pp. 342-349).

[9] Yang, Y. & Pedersen, J.O. (1997). A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. pp. 412-420). : Morgan Kaufmann Publishers Inc, San Francisco, CA, USA

[10] Rupali Bhaisare,T. RajuRao 2013 "Review On Text Mining With Pattern Discovery".

[11] Muhammed Miah, "Improved k-NN Algorithm for Text Classification", Department of Computer Science and Engineering University of Texas at Arlington, TX, USA.

[12] Fang Lu Qingyuan Bai, "A Refined Weighted K-Nearest Neighbours Algorithm for Text Categorization", IEEE 2010.

[13] Kwangcheol Shin, Ajith Abraham, and Sang Yong Han, "Improving kNN Text Categorization by Removing Outliers from Training Set", Springer-Verlag Berlin Heidelberg 2006.

[14] Robert Burbidge, Bernard Buxton2000's An Introduction to Support Vector Machines for DataMining.

[15] Vidhya. K.A G.Aghila, "A Survey of Naïve Bayes Machine Learning approach in Text Document Classification", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, 2010.

[16] S. M. Kamruzzaman, Chowdhury Mofizur Rahman: "Text Categorization using Association Rule and Naive Bayes Classifier" CoRR, 2010.

[17] MIgual E .Ruiz, Padmini Srinivasn, "Automatic Text Categorization Using Neural networks", Advaces in Classification Research, Volume VIII.

[18] J.J. Rocchio. Document Retrieval Systems–Optimization and Evaluation. PhD thesis, Harvard Computational Laboratory, Cambridge, MA, 1966.

[19] J.J. Rocchio. Relevance feedback in information retrieval. In The SMART Retrieval System—Experiments in Automatic Document Processing, pages 313–323, Englewood Cliffs, NJ, 1971. Prentice Hall, Inc

[20] Kjersti Aas and Line Eikvil "Text Categorization: A Survey" Report No. 941. ISBN 82-539-0425-8. , June, 1999.