Original Research Paper

Volume-9 | Issue-6 | June-2019 | PRINT ISSN No. 2249 - 555X

Computer Science

# A COMPREHENSIVE STUDY ON CLASSIFICATION OF AUTOMATED CATEGORIZATION OF WEB SITES: A PROPOSED METHODOLOGY

**Amish D. Vyas** Research Scholar, JJTU (Rajasthan)

**Dr. Yogesh Kumar Sharma*** Associate Professor (HOD), JJTU (Rajasthan) *Corresponding Author

**ABSTRACT** Contemporary web is comprised of trillions of pages and everyday tremendous amount of requests are made to put more web pages on the WWW. It has been difficult to manage information present on web than to create it. Web page categorization can be defined as an approach to categorize the web pages based on a set of predefined categories to manage large web content. Yahoo! and ODP are the examples of web directories in which pages are categorized manually or semi automatically, but it is a very time consuming task. There are many ways of categorizing web pages using different techniques. An approach to categorize web pages automatically on the basis of characteristics of web pages using neural network based single discrete perceptron training algorithm which is extended by selecting web page specific features to categorize web pages of predefined categories with high accuracy. The idea is presented with the help of two specific and major categories of web pages chosen for categorization that are newspaper and education. Classification of Web pages is one of the challenging and important task as there is an increase in web pages in day to day life provided by internet. There are many ways of classifying web pages based on different approach and features. In this paper, a soft computing approach is proposed for classification of websites based on features extracted from URLs alone. The Open Directory Project dataset was considered and the proposed system classified the websites into various categories using Naive Bayes approach. The agenda of this paper is first to introduce the concepts related to web mining and then to provide a comprehensive review of different classification techniques. One of the classification algorithms used in WebDoc is based on Bayes' theorem from probability theory. This paper focuses upon three aspects of this approach: different event models for the naive Bayes method, different probability smoothing methods, and different feature selection methods. In this paper, we report the performance of each method in terms of recall, precision, and F-measures. Experimental results show that the WebDoc system can classify Web documents effectively and efficiently.

**KEYWORDS :** Terms- classification, WebDoc, naive Bayes method.

## 1.0 INTRODUCTION:

World Wide Web (WWW) is a widespread and collaborating medium with excellent growth of amount. World Wide Web has made it essential for users to operate automated tools in finding the desired information resources. The World Wide Web is the collection of text files, documents, images, and other forms of data in unstructured, semi structured and structured form. The Web is the largest data source in the world. Classification plays a vigorous role in many information management tasks and reclamation tasks.

Document classification refers to the task of "developing a system that is able to automatically classify a text document into a number of categories relevant to the document". Due to the extensive use of the World Wide Web, the huge amounts of information on the Web make an attractive resource. The lack of logical organization of Web documents makes retrieving relevant information from the Web a laborious and time consuming task, and motivates the development of automatic Web document classification systems. Automatic document classification is an active and challenging field of research, and an extensive range of algorithms has been proposed. Typically-used methods include the decision tree method, k-nearest neighbor method (kNN), Naive Bayes method (NB), Bayesian networks, neural networks (NNet), support vector machines (SVM), and subspace model. This paper describes an automated document classification system, WebDoc (The Web Document Classification System), which was developed by researchers in the Department of Computer Science and Engineering at Mississippi State University. WebDoc uses the Library of Congress classification scheme to classify HTML documents that have been downloaded from the Web. The WebDoc system introduced in this paper was implemented using a naive Bayes method based on Bayes' theorem from probability theory. The study is focused upon two different Naive Bayes models: a multi-variate Bernoulli event model and a multinomial event model. In this paper, two different probability smoothing methods were tested: additive smoothing method and Good-Turing smoothing method. Four feature selection criteria were tested: inverse document frequency (IDF), information gain (IG), mutual information (MI) and $\chi 2$ (CHI). In the WebDoc system, the Library of Congress Subject Headings (LCSHs) is used as the indexes for the Web documents. The rest of this paper is organized as follows. In section two, we begin with an overview of the WebDoc classification system. We follow that with an introduction of how to use the naive Bayes method in a document classification system in section. The NLP tags the original Web document with syntactic and semantic tags (such as noun and astronomy) and parses the document (thus making it possible to isolate sentential components such as noun phrases). The knowledge base construction component builds a knowledge base of information that includes the Library of Congress (LCC) subject headings and their interrelationships as well as other information used during classification. The index generation component generates a set of candidate indexes for each document in a test set of documents. (In this paper, we use the terms subject headings and indexes to mean the same thing.)

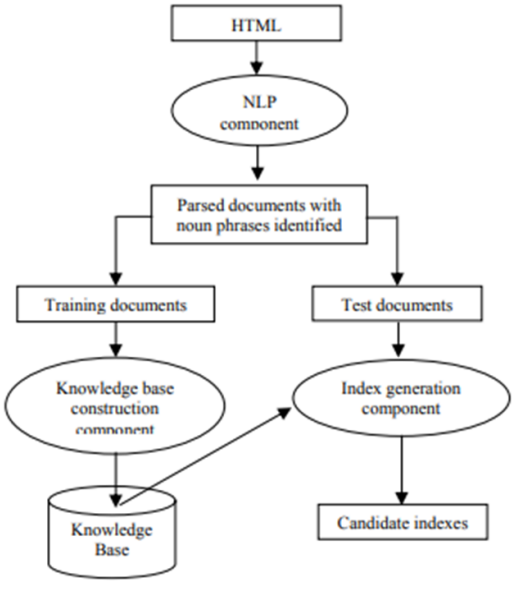


**Figure 1. Architecture of the WebDoc system**

## 2.0 LITERATURE REVIEW

**Min-Yen Kan, (2004)** Uniform resource locators (URLs), which mark the address of a resource on the World Wide Web, are often human-readable and can hint at the category of the resource. This paper

explores the use of URLs for web page categorization via a two-phase pipeline of word segmentation/expansion and classification. We quantify its performance against document-based methods, which require the retrieval of the source document.

**Patrick Dave P, (2017)** The Internet is a powerful instrument that contains hundreds to thousands of resources. There is a need to categorize these resources based on certain categories in order to organize the contents of the Web better. This research aims to build a corpus that would be representative of pre-defined educational categories. This study will experiment on seven different algorithms that will be able to categorize web pages based on educational domain. Many studies about web categorization have already been conducted but is based on a general set of categories. This research will focus primarily on a predefined set of categories that are closely related to educational domains. With the use of machine learning, the classifier will be able to analyze what a web page is all about and determine its category. The study will also compare the different classifiers used. As a result, the system will be able to assign a web page to a particular educational domain and can be used by schools to determine the categories of web pages frequently requested by students. Linear SVM was also able to build a lexicon for the different categories. The top words for each category were then determined using this lexicon.

**Pooja Vinod Nainwani, (2018)** Classification of Web pages is one of the challenging and important task as there is an increase in web pages in day to day life provided by internet. There are many ways of classifying web pages based on different approach and features. This paper explains some of the approaches and algorithms used for the classification of webpages. Web pages are allocated to pre-determined categories which is done mainly according to their content in Web page classification. The important technique for web mining is web page classification because classifying the web pages of interesting class is the initial step of data mining. The agenda of this paper is first to introduce the concepts related to web mining and then to provide a comprehensive review of different classification techniques.

## 3.0 METHODOLOGY:

The naive Bayes method (NB) is a simple Bayesian classifier based on Bayes' theorem from probability theory. In the WebDoc system, the stem forms of words occurring in the training documents were used as the features to represent each document. The basic steps in the naive Bayes method are as follows:

Training:

- Identify the individual stem words occurring in all the training documents in the training set.
- Generate the feature vector for each document in the training document set and store it along with the correct indexes in the knowledge base.
- Calculate the probability for each index.

Testing:

- Identify the individual stem words occurring in a given test document.
- Generate the feature vector for this document.
- Calculate the probability for this document given each index.
- Calculate the probability for each index in the set of indexes for this document and normalize it with Bayes' theorem, this value is the weight of this index.
- Select the indexes with a weight higher than a predefined threshold as the candidate indexes for this document.

**Naive Bayes models** Although a naive Bayes classifier is a simple and popular technique used in the document classification area, it has been implemented by different researchers with two different generative models: multi-variate Bernoulli event models and multinomial event models [12]. In the multi-variate Bernoulli model, a binary representation is used for the value of a feature in the feature vector, which mean the possible value for each feature is only 0 or 1. A value of 1 for feature Ai indicates that feature Ai (stem form of a noun phrase) occurred in that document (xi = 1). A value of 0 for Ai indicates that feature Ai did not occur in that document (xi = 0). The occurrence frequencies of these features in the documents are not captured. In this model, a document is seen as an "event" and the absence or presence of words is an attribute of the event. In the multinomial event model, the number of occurrences of each feature Ai (stem form of a noun phrase) in the document is captured and each feature vector is represented by a list of occurrence frequencies of all features. The value 0 of feature Ai means that this feature did not occur in the document. In order to avoid the effect of the varying lengths of the documents, all the occurrence frequencies are normalized before being used.

**Probability smoothing:**

Smoothing is a "technique used to better estimate probabilities when there is insufficient data to estimate probabilities accurately". The goal of various smoothing techniques is to make the distribution of probabilities more uniform. One principle of the various smoothing methods is the sum of the all probabilities must be 1. Two smoothing methods were used in the WebDoc system, additive smoothing and Good-Turing smoothing. The additive smoothing method is one of the simplest smoothing methods used in practice. In this method, the occurrence frequency of each feature was increased by 1. Then the estimated probability of each feature given a conclusion can be calculated with the following formulation:

$$P(A_i|C_j) = \frac{N_{ij}+1}{\sum_{i=1}^{n} N_{ij}+n}$$

where $N_{ij}$ is the actual occurrence frequency of feature Ai given conclusion $C_j$ and n is the size of feature vector.

The Good-Turing smoothing method is based on the Good-Turing estimate. In this method, the probability of an occurred feature is replaced with a smaller probability. The sum of the smaller probabilities is subtracted from 1.0; this difference is distributed evenly among the unseen features. In this method, let *r* represent the frequency of a given feature. Then Nr is the number of features with a frequency of *r*. The value $r^*$ is the estimated frequency, which is calculated based upon the frequencies and the N*r* values:

$$r^* = (r+1)\frac{E(N_{r+1})}{E(N_r)}$$

where E(*x*) is the expectation of the random variable *x*. Most of the $N_r$ values will be 0 for a large value of *r*. To account for these 0s, Church and Gale average with each nonzero value N*r* the zero N*r* values that surround it. Order the nonzero values by *r*. Let *q, r,* and t be successive indexes of nonzero values. Replace N*r* by Z*r*:

$$Z_r = \frac{N_r}{0.5(t-q)}$$

So the expected N*r* is estimated by the density of Nr for large *r*.

Let *b* represent the slope of the line defined where the x-axis represents log® and the y-axis represents log($Z_r$). Then $r^*$ is calculated as:

$$r^* = r(1+\frac{1}{r})^{b+1}$$

The probability of each feature that occurred at least once is $r^*/N$ where *N* is the sum of the frequencies. The difference between 1.0 and the sum of the nonzero probabilities is distributed evenly among the nonoccurring features.

**Feature Selection:**

The goal of the feature selection is to try to remove non-informative features and reduce the dimensionality of the feature vector. Four feature selection methods were used in the WebDoc system: inverse document frequency (IDF), information gain (IG), mutual information (MI), and $\chi^2$ (CHI Square).

Inverse document frequency is computed based on collection frequency. The collection frequency of a term is the number of documents in which that term occurs. The IDF value of term *i* is *log(N/Ni)*, where N is the total number of documents in the collection and *Ni* is the collection frequency of term *I* Information gain (IG) is a measure based on entropy. This method measures how much additional information you can get from each feature by including a particular index and select the optimal one. Given a set of training documents whose size is s and the size of each category is $s_i$, the expected information that is needed to classify a given document is.

$$I(s_1, s_2, \ldots, s_m) = -\sum_{i=1}^{m}\frac{s_i}{s}\log\frac{s_i}{s}$$

For each feature of the feature vector *A*, assume it has v different values, the information gain of feature *A* based on the entropy is:

$$E(A) = \sum_{j=1}^{v}\frac{s_{1j}+\ldots+s_{mj}}{s} I(s_{1j},\ldots s_{mj})$$

The definition of mutual information used in our experiments was adapted from the one used. As before, let A represent a feature in the feature vector and let C represent a subject heading. The mutual information between A and C is:

$$I(A, C) = \log \frac{P(A_i \cap C_j)}{P(A_i) \times P(C_j)}$$

The mutual information may be estimated as

$$I(A, C) \approx \log \frac{a \times n}{(a+c) \times (a+b)}$$

Where n is the size of training documents. a is the number of documents in which both *A* and *C* occur. *b* is the number of documents in which *A* occurs but *C*. c is the number of documents in which *C* occurs but *A*. The final MI value for a feature is the average of all values for different categories.

The $\chi^2$ (CHI) method is a method similar to the MI method. Assume d is the times when none of *A* and *C* occurs, the estimation of $\chi^2$ value of *A* and *C* is:

$$\chi^2(A, C) = \frac{n \times (ad - cb)^2}{(a+c) \times (b+d) \times (a+b) \times (c+d)}$$

F-measure is the harmonic mean of recall and precision. Recall, Precision and F-Measure are calculated as follows:

$$\text{Precision} = \frac{\text{Number of relevant documentsretrieved}}{\text{Total number of documentsretrieved}}$$

$$\text{Recall} = \frac{\text{Number of relevant documentsretrieved}}{\text{Total number of relevant documents}}$$

$$\text{F- measure} = \frac{\text{(2 x Recall x Precision)}}{\text{(Recall + Precision)}}$$

## 4.0 RESULTS:

A total of 722 documents downloaded from the Web were used as the data in our experiments. All these documents have been assigned the correct LCSHs by an expert librarian. The 5-fold cross-validation method was used to divide the documents into a training set and test set. The performance of the WebDoc system was evaluated by the well-known measures of precision, recall, and F-measure. For each experiment, all the candidate indexes were filtered according to their weight. In order to make a comparison, all the weights were normalized into the range from 0 to 1. Then the threshold was set from 0 to 0.9 with step 0.1. The experimental results for two naive Bayes models were given in table 1. The comparison results for different smoothing methods were listed in table 2. In the table 3 and table 4, the experimental results for four feature selection methods were listed.

**Table 1. Multi-variate Bernoulli Model (MB) vs. Multinomial Model (MN)**

| Thre. | Precision | | Recall | | F-Measure | |
|---|---|---|---|---|---|---|
| | MB | MN | MB | MN | MB | MN |
| 0 | 0.14 | 0.14 | 1 | 1 | 0.25 | 0.25 |
| 0.1 | 0.17 | 0.3 | 0.87 | 0.99 | 0.29 | 0.46 |
| 0.2 | 0.2 | 0.41 | 0.87 | 0.92 | 0.32 | 0.57 |
| 0.3 | 0.23 | 0.55 | 0.87 | 0.78 | 0.36 | 0.65 |
| 0.4 | 0.24 | 0.61 | 0.87 | 0.71 | 0.38 | 0.66 |
| 0.5 | 0.27 | 0.63 | 0.87 | 0.68 | 0.41 | 0.65 |
| 0.6 | 0.32 | 0.63 | 0.87 | 0.66 | 0.47 | 0.65 |
| 0.7 | 0.35 | 0.66 | 0.83 | 0.63 | 0.49 | 0.64 |
| 0.8 | 0.53 | 0.68 | 0.69 | 0.48 | 0.6 | 0.57 |
| 0.9 | 0.63 | 0.72 | 0.63 | 0.43 | 0.63 | 0.54 |

**Table 2. Experimental results of smoothing methods**

| Thre. | Precision | | | Recall | | | F-Measure | | |
|---|---|---|---|---|---|---|---|---|---|
| | NO | Add | GT | NO | Add | GT | NO | Add | GT |
| 0 | 0.14 | 0.14 | 0.14 | 1 | 1 | 1 | 0.25 | 0.25 | 0.25 |
| 0.1 | 0.3 | 0.16 | 0.18 | 0.99 | 1 | 1 | 0.46 | 0.28 | 0.31 |
| 0.2 | 0.41 | 0.19 | 0.21 | 0.92 | 1 | 1 | 0.57 | 0.32 | 0.35 |
| 0.3 | 0.55 | 0.22 | 0.23 | 0.78 | 1 | 1 | 0.65 | 0.35 | 0.37 |
| 0.4 | 0.61 | 0.23 | 0.25 | 0.71 | 1 | 1 | 0.66 | 0.38 | 0.39 |
| 0.5 | 0.63 | 0.25 | 0.27 | 0.68 | 1 | 1 | 0.65 | 0.41 | 0.42 |
| 0.6 | 0.63 | 0.3 | 0.33 | 0.66 | 1 | 1 | 0.65 | 0.46 | 0.5 |
| 0.7 | 0.66 | 0.39 | 0.45 | 0.63 | 0.97 | 0.94 | 0.64 | 0.56 | 0.61 |
| 0.8 | 0.68 | 0.56 | 0.58 | 0.48 | 0.8 | 0.79 | 0.57 | 0.66 | 0.67 |
| 0.9 | 0.72 | 0.63 | 0.63 | 0.43 | 0.7 | 0.69 | 0.54 | 0.66 | 0.67 |

**Table 3. Experimental results of feature selection methods (IDF vs. IG)**

| Thre. | Precision | | Recall | | F-Measure | |
|---|---|---|---|---|---|---|
| | IDF | IG | IDF | IG | IDF | IG |
| 0 | 0.14 | 0.14 | 1 | 1 | 0.25 | 0.25 |
| 0.1 | 0.22 | 0.28 | 1 | 0.99 | 0.36 | 0.44 |
| 0.2 | 0.3 | 0.37 | 0.99 | 0.93 | 0.46 | 0.53 |
| 0.3 | 0.39 | 0.47 | 0.93 | 0.81 | 0.55 | 0.59 |
| 0.4 | 0.5 | 0.53 | 0.82 | 0.74 | 0.62 | 0.62 |
| 0.5 | 0.59 | 0.57 | 0.73 | 0.69 | 0.65 | 0.62 |
| 0.6 | 0.63 | 0.6 | 0.68 | 0.67 | 0.65 | 0.63 |
| 0.7 | 0.66 | 0.62 | 0.65 | 0.64 | 0.66 | 0.63 |
| 0.8 | 0.69 | 0.66 | 0.57 | 0.5 | 0.63 | 0.57 |
| 0.9 | 0.74 | 0.7 | 0.43 | 0.44 | 0.55 | 0.54 |

**Table 4. Experimental results of feature selection methods (MI vs. $\chi^2$)**

| Thre. | Precision | | Recall | | F-Measure | |
|---|---|---|---|---|---|---|
| | MI | $\chi^2$ | MI | $\chi^2$ | MI | $\chi^2$ |
| 0 | 0.14 | 0.14 | 1 | 1 | 0.25 | 0.25 |
| 0.1 | 0.23 | 0.28 | 1 | 0.98 | 0.37 | 0.43 |
| 0.2 | 0.29 | 0.37 | 0.99 | 0.93 | 0.45 | 0.53 |
| 0.3 | 0.34 | 0.46 | 0.93 | 0.8 | 0.5 | 0.58 |
| 0.4 | 0.4 | 0.51 | 0.82 | 0.74 | 0.54 | 0.6 |
| 0.5 | 0.48 | 0.55 | 0.75 | 0.7 | 0.59 | 0.62 |
| 0.6 | 0.57 | 0.58 | 0.69 | 0.68 | 0.62 | 0.63 |
| 0.7 | 0.62 | 0.6 | 0.67 | 0.64 | 0.64 | 0.62 |
| 0.8 | 0.64 | 0.65 | 0.58 | 0.5 | 0.61 | 0.57 |
| 0.9 | 0.67 | 0.68 | 0.47 | 0.41 | 0.55 | 0.51 |

Our experimental results indicate that: First, compared with the previous versions of WebDoc, whose results were reported in, we obtained an increase in the F-measure of almost 20 percentage points (i.e., 67.19%). Second, compared with the reported results of other automated document classification systems, the performance of WebDoc is favorable, especially considering that some of those researchers whose systems had higher recall, precision, and/or Fmeasures than ours were not attempting to classify documents as unstructured and varied as the Web documents that we worked with. For example, in, the total number of categories used by Quek's web document classification system is only seven (Course, Student, Faculty, department, Staff, Research project, and other). And the experimental data is limited to the homepages of the computer science departments of four universities. In, Yang made a comparison of ten different classification algorithms on the Reuters corpus, which is a standard data set for the evaluation of document classification systems. The BEP value achieved by yang's naive Bayes method is 66%, which is also similar to the performance of WebDoc system. Third, in the WebDoc system, the multinomial event model classifier had a better performance than the multivariate Bernoulli event model. This result is consistent with that in. Fourth, two smoothing methods, additive smoothing and the Good-Turing smoothing methods, increased the recall value of the classifier greatly but decreased the precision. The F-measure results demonstrate that when a higher threshold is set, both smoothing methods are helpful for generating more correct indexes and did improve the performance of the classifier. Fifth, although four different feature selection methods were used in the WebDoc, none of them improved the performance notably.

## 5.0 CONCLUSION:

WebDoc is an automated classification system that assigns Web documents to appropriate Library of Congress subject headings based upon the text in the documents. In this paper, the architecture and design of WebDoc were presented. WebDoc used the Bayes' theorem as basic algorithm and was implemented with two different models: a multi-variate Bernoulli event model and a multinomial event model. Two different probability smoothing methods and four different feature selection measures were applied in the Web Doc.

## REFERENCES:


1. McCallum and K. Nigam, (1998), "A comparison of event models for Naïve Bayes text classification", in AAAI/ICML-98 Workshop on Learning for Text Categorization, pp. 41-48.
2. D.D. Lewis and M. Ringuette, (1994), "A Classification of two learning algorithms for text categorization", in Proc. of 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR'94), pp. 81-93.
3. S.T. Dumais, J. Platt, D. Heckerman, and M. Sahami, (1998), "Inductive learning

algorithms and representations for text categorization", in Proc. of the 17th Int. Conf. on Information and Knowledge Management (CIKM'98), pp. 148-155.
4. Apte and F. Damerau and S. M. Weiss, (1994), "Automated learning of decision rules for text categorization", ACM Trans. on Information Systems, Vol. 12, no.3, pp. 233-251,.
5. S. Wermter, (2000), "Neural network agents for learning semantic text classification", Information Retrieval, Vol. 3, no. 2, pp. 87 - 103, Jul.
6. A.S. Weigend, E.D. Weiner, and J.O. Peterson, (1999), "Exploiting hierarchy in text categorization", Information Retrieval, Vol. 1, no. 3, pp.193- 216,.
7. E. Leopold, and J. Kindermann, (2002), "Text categorization with support vector machines. How to represent texts in input space?", Machine Learning, Vol. 46, no. 1-3, pp. 423-444.
8. D. Bennett and A. Demiritz, (1998), "Semi-Supervised support vector machines", Advances in Neural Information Processing Systems, Vol. 11, pp. 368-374,.