DESIGN AND IMPLEMENTATION OF AN EFFICIENT FRAMEWORK FOR WEB PAGE CLASSIFICATION

A Thesis submitted in fulfillment of the requirement for the award of the degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Vinod Kumar

(Registration No: 951103005)

Under the guidance of

Dr. Neeraj Kumar

Associate Professor, CSED, TU.

COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT

THAPAR UNIVERSITY, PATIALA – 147004

AUGUST 2016

CERTIFICATE

I, Vinod Kumar, Regn. No. 951103005, hereby declare that the thesis entitled **"Design and Implementation of an Efficient Framework for Web Page Classification"** submitted to the Department of Computer Science and Engineering at Thapar University, Patiala, Punjab, India is an authenticated record of my own work for the award of the degree of "Doctor of Philosophy" under the supervision of Dr. Neeraj Kumar. This report has not been submitted to any other institution for award of any other degree.

Vinod Kumar

Regn. No. 951103005

Place: Patiala

Date: 21/10/16

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Verified by:

Dr. Neeraj Kumar,

Associate Professor,

Computer Science and Engineering Department,

Thapar University,Patiala.

ABSTRACT

With an evolution of Internet and related technologies, there is a great need of an efficient web page categorization for getting the fast response with respect to searching and classification of various documents on the web. Due to large number of user's request, there may be a performance bottleneck during searching and classification of web documents with respect to various QoS parameters such as response time and congestion. Classification helps in searching, sorting, retrieval, and querying of various documents. World Wide Web (WWW) contains huge repository of information in the form of web pages. But, size of Internet is growing day-by-day which results an efficient classification of different web pages to achieve higher accuracy. The huge repository of information poses challenge to collect and process the relevant related information of a particular domain.

Most of the solutions reported in the literature are not adequate to address above issues for getting a fast response time with respect to web page categorization. Also, most of the existing techniques reported in the literature are semi-automatic. Using these techniques, higher level of accuracy cannot be achieved. So, traditional text classification techniques are difficult to apply on the rapidly growing web-based contents. Moreover, manual categorization of these billions of web pages to achieve high accuracy is a cumbersome and tough task.

To address these issues, in this thesis, novel techniques for web page categorization are proposed. In these techniques, personality features are collected and assigned weights. Then, the proposed classifiers are trained based on these special features. The proposed techniques are based on the identification of specific

and relevant features of the web pages. In the proposed scheme, first extraction and evaluation of features are done followed by filtering the feature set for categorization of domain web pages. A feature extraction tool(FET) based on the HTML document object model(DOM) of the web page is developed in the proposed scheme.

In first technique, binary classification, feature extraction and weight assignment are based on the collection of domain-specific keyword list developed by considering various domain pages such as course, student, faculty etc. Moreover, the keyword list is reduced on the basis of *ids* of keywords in keyword list. Also, stemming of keywords and tag text is done to achieve a higher accuracy. An extensive feature set is generated to develop a robust classification technique. The proposed technique was evaluated using a machine learning method in combination with feature extraction and statistical analysis using support vector machine kernel as the classification tool.

In second technique, multiclass classification, on-page personality feature sets are extracted and weights are assigned based on feature frequency on web document for each domain. A combined feature set is proposed. Algorithms are designed and these are tested and validated with respect to various data sets collected from different domain categories such as E-Newspaper, Education, Research, Online shopping, Resume. Results obtained depict that proposed classifier successfully classified news domain pages, education, resume, online shopping, and research web pages from large database repository. Accuracy of the proposed classifier is found to be satisfactory from a large data set of different categories. Also, there is a 10–15 % overall performance gain using the proposed scheme in comparison to the other existing schemes. The results obtained confirm the effectiveness of the proposed scheme in terms of its accuracy in different categories of web pages.

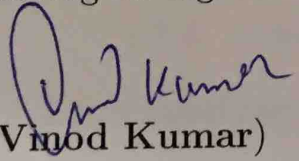
ACKNOWLEDGMENTS

I deem it my duty to express a word of hearty gratitude to all those helping hands that in the process of writing this thesis have become part and parcel of this endeavor. First of all I wish to acknowledge the benevolence of omnipotent God who gave me strength and courage to overcome all obstacles.

With profound sense of gratitude and heartiest regard, I express my sincere feelings of indebtedness to my Guide Dr Neeraj Kumar, Associate Professor, Computer Science Engineering Department, Thapar University for his valuable guidance, advice, motivation, encouragement, moral support, sincere effort, invaluable co-operation, generous and positive attitude with which he solved my queries and provide delightful ambiance for learning, exploring and making this thesis possible. It has been a great pleaser and experience to work under his sanctuary.

I am grateful to Head of Department, Dr. Maninder Singh and Professor, Dr. Deepak Garg, (former HOD) who made my study a relevant experience during my stay in the department. I am much beholden to Director, Dean Research and Management of Thapar University, who provided me all the necessary resources and encouraged to produce results. I sincerely thank Ph.D. committee members, computer science and engineering department faculty and support staff for their constant motivation.

Most importantly, I would like to thank my Family for showing me the right direction out of the blue, to help me stay calm in the oddest of the times and keep moving for logical conclusion of proposed work.



(**Vinod Kumar**)

Contents

List of Figures

List of Tables

LIST OF IMPORTANT ABBREVIATIONS

DOM	Document Object Model
KNN	K Nearest Neighbour
NB	Naive Bayes
NN	Neural Network
SVM	Support Vector Machine
GA	Genetic Algorithms
RL	Relational Learning
QoS	Quality of Service
IoT	Internet of Things
D_i	Domain Pages
FET	Feature Extraction Tool
KW	Keyword
KWL	Keyword List
IDF	Inverse Document Frequency
TF	Term Frequency
RBF	Radial Basis Function
P	Precision
R	Recall
F1	F-measure
Sens	Sensitivity
Acc	Accuracy
CV	Cross Validation
ROC	Receiver Operating Characteristics

Chapter 1

Introduction

World Wide Web (WWW) data is in the form of web pages and web sites composed of web pages. From last two decades, with the popularity of Internet and related technologies, there is a steep increase in the web pages access from different communities of users across the globe. Daily billions of web pages are added to the huge repository of web-based contents in the form of web pages. These web pages contain information about almost everything. Contents on the web are freely published by a very large number of people. Therefore, according to Pierre [1], it is important to describe and organize the large content present on the web in order to realize web's full potential.

The database repository that contains such a large collection of web pages may be centralized or distributed. Although overhead to maintain a centralized repository is less but it has a single point of failure. So, this makes the popularity of distributed repository as it can be accessed from anywhere by the end users even on-the-fly as per their choice. This type of scenario creates a large scope for the enhancement in terms of management of database repository to keep relevant web pages.

Internet has reached to masses and size of information has surpassed peta bytes.

Professionals, amateur users, producers, creator and consumers are continuously adding various data items each day in the existing database repository. Rapid industrial growth, global business economy, global marketing, scientific developments, research in diversified areas, vibrant national and international politics, sports events and zeal to use and exchange information and data result in exponential increase in the of size of Internet and web pages. There is no solid editorial control over the verity of contents. Searching relevant contents is a challenge for individuals and search engines.

Classification of web content can reduce the efforts to many folds. Web page classification is an intellectual task and it requires human intervention. Dmoz's open directory project [53] (ODP) of web contents uses services of approximately seventy five thousand domain experts. These domain experts manually or semi-automatically classify web content with great effort so that the classified content can be used judicially and effectively for various purposes. Classified web pages can be used for fast and easy retrieval of the contents from the database repository. It also helps in developing question-answer sessions, focused crawling, targeted search and collecting domain knowledge. Therefore, it is extremely important to organize and describe the large content present on the web in order to realize web's full potential. According to Pinkerton, B. [2] in proceedings of the First World Wide Web Conference Geneva, Switzerland, web is a great information resource. Automatic web classification is an important and indeed essential for organizing and understanding web content for different applications as shown in figure 1.1.

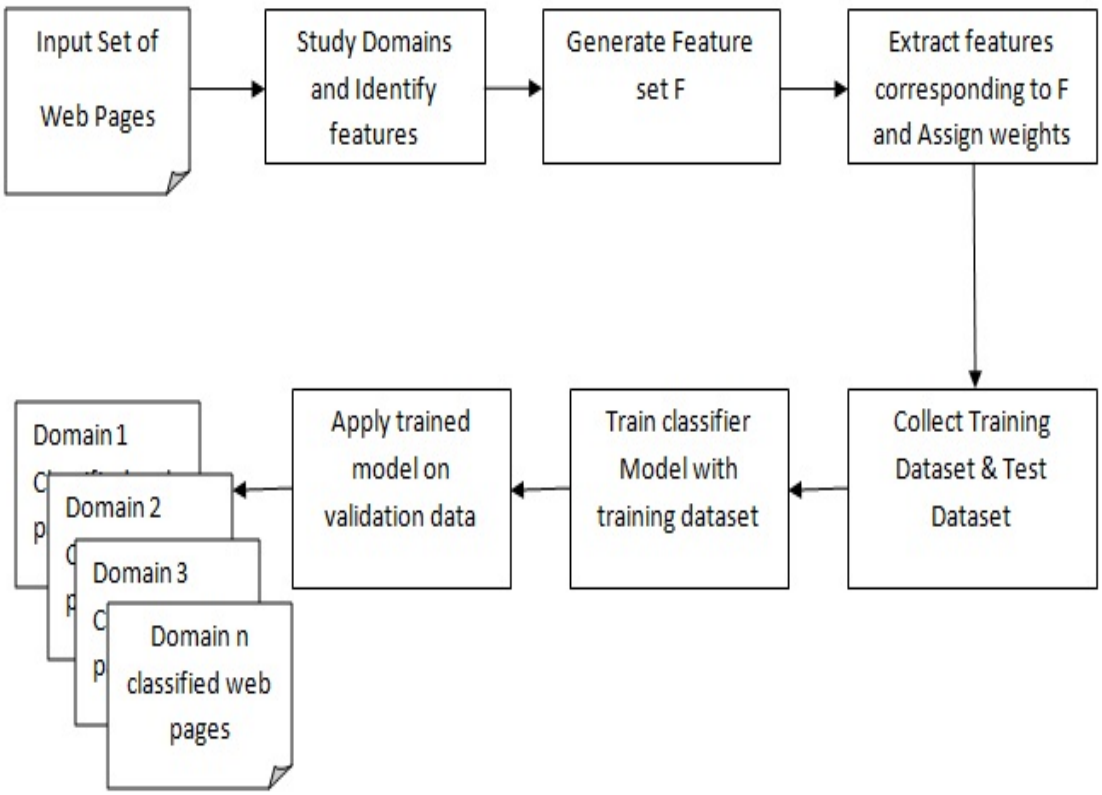


Figure 1.1: Flow chart- Classification Process

A classifier applied to the WWW faces a huge-scale dimensionality problem since it must handle billions of Web pages, tens of thousands of features, and hundreds of categories (Chih-Ming *et al.* [3]). The data available on the web is in the form of text, images, audio, video, graphics and many other forms [1,2,,4,8,14,18 *et al.*]. So, the dynamic nature of web and large scale explosion of web pages may put a threat to efficient information retrieval tasks, John M. Pierre [4]. Automated classification is an important task required for a number of applications as proposed by Lin Li *et al.* [5], Chakrabarti *et al.* [6], Yang, H [7], Nie *et al.* [8] . This proposal has covered different types of classification along with various features and characteristics of the web pages to be considered during the categorization. To quantify the results of classification precision and recall metrics are used (Pietramala *et al.* [9]). F1-measure is generally used to check the performance of the classifier.

1.1 Web Page Classification/Categorization

Web page classification also popularly called as web page categorization. It is the process of allocating a web page to one or more predefined categories. Web page classification is considered a challenging problem because of huge and exponentially increasing size of WWW. Supervised learning techniques need to be developed for automatic and easier classification. To perform the task of automatic classification, number of features needs to be identified and selected based on the purpose of classification. Feature sets are created from the shortlisted features and a set of labeled data set is generated to train a classifier. Web page classification process contributes toward management of information, retrieval of information, focused crawling, development of web directories and link analysis.

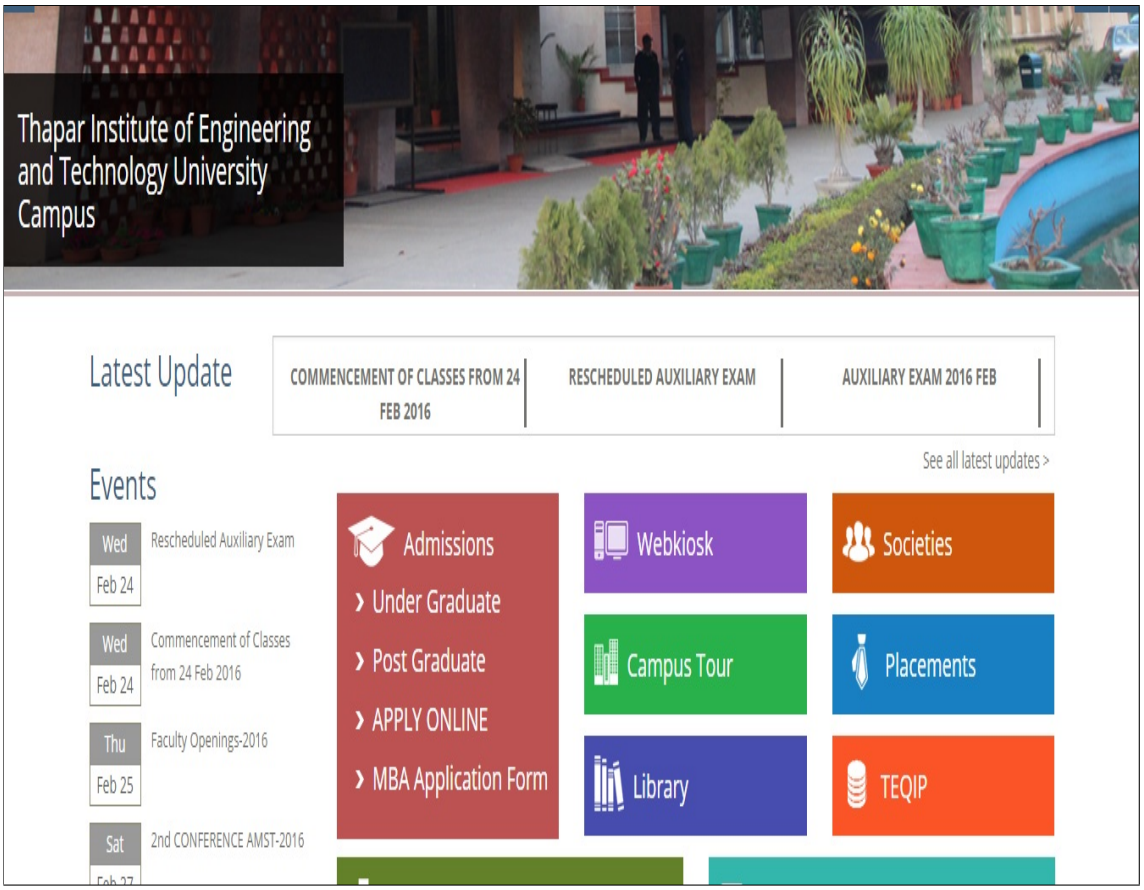


Figure 1.2: Sample University Home Page

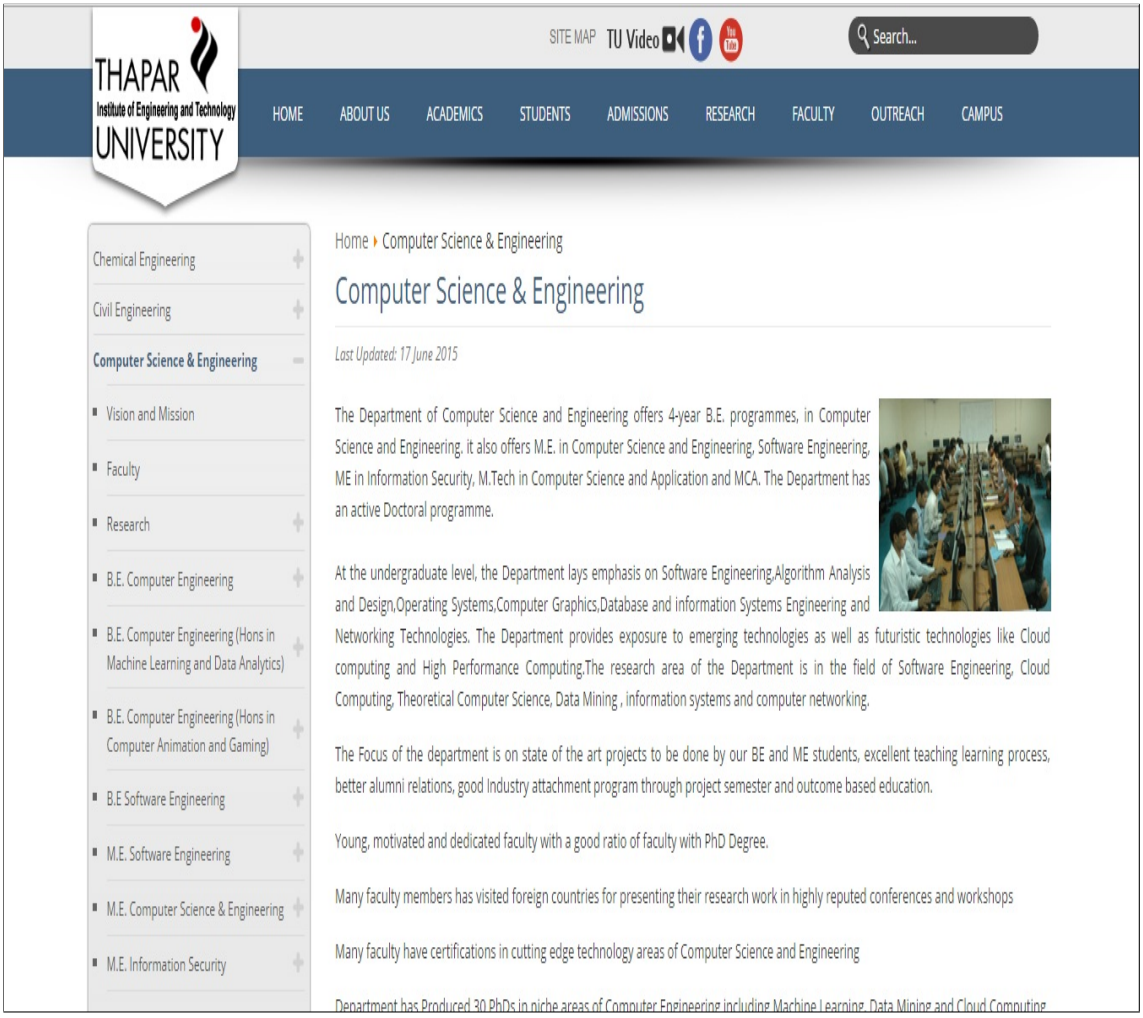


Figure 1.3: Sample University Education Department Page

Classification of web also helps in improving the quality of search result. Along with this, invent of semantic web, categorization of web documents open doors for many other applications. Web pages are developed using html tags and presentation and layout out information is quite different from the simple text presentation.

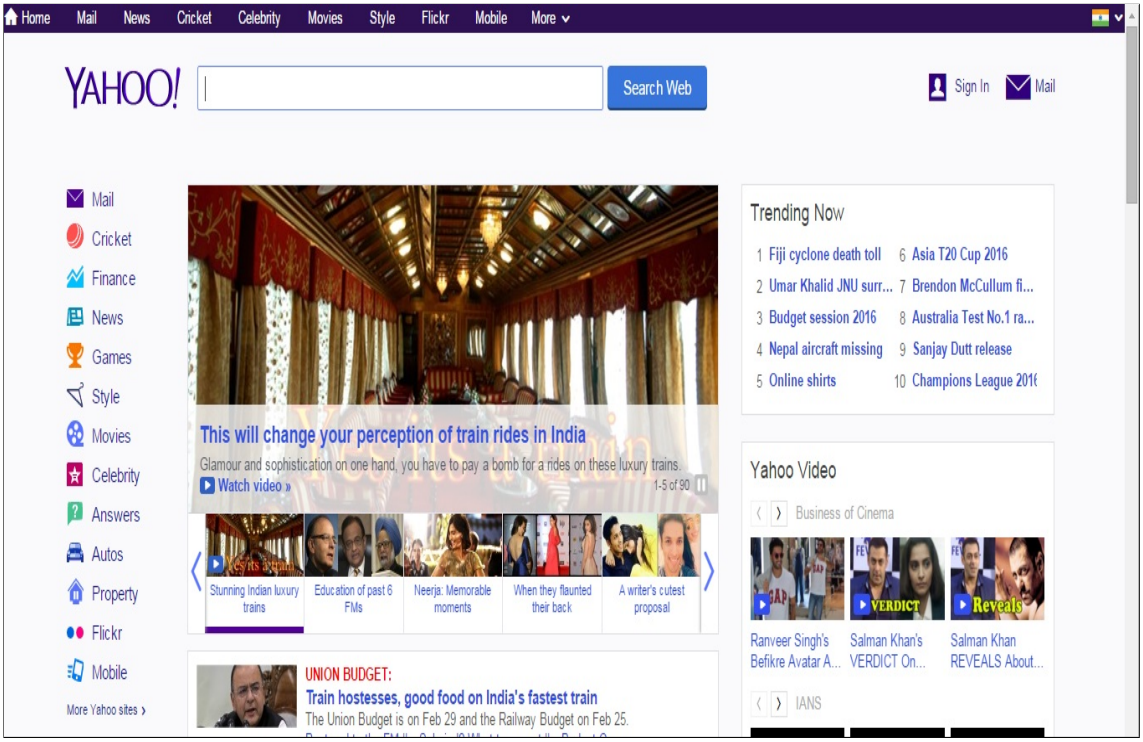


Figure 1.4: Sample Search Engine Home Page

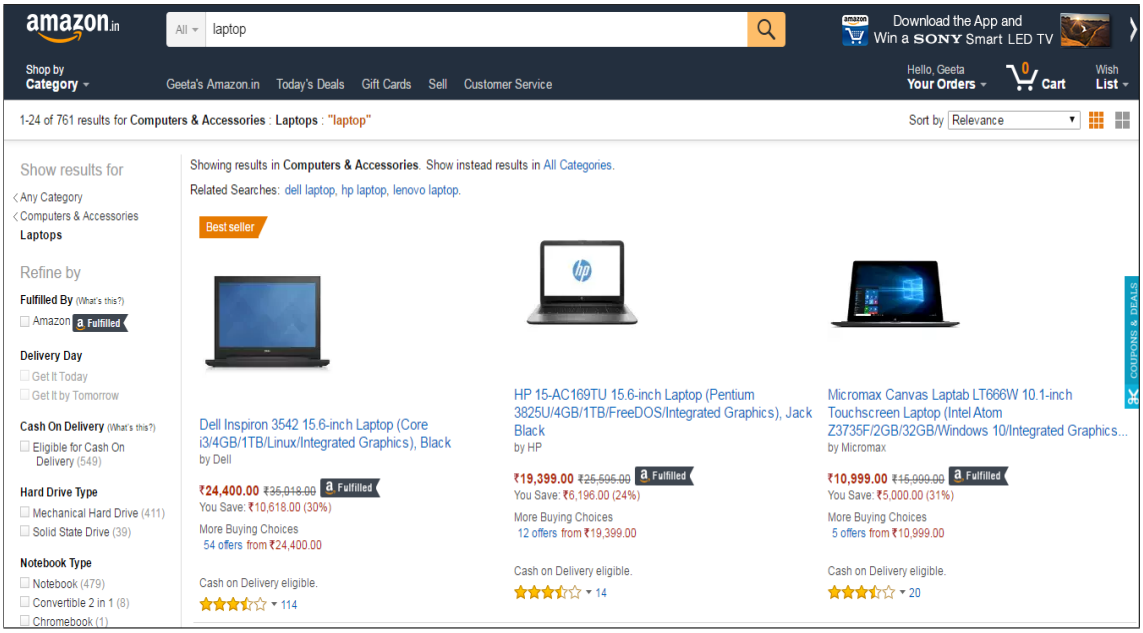


Figure 1.5: Sample Online shopping Page

Traditional classification techniques are not sufficient for web classification. Innovative machine learning schemes and algorithms are needed for web document classification. Even the features required for classification purpose are different from simple text document. Many research techniques are being developed to produce cost effective and computational less expensive solution.

1.2 Types of Web Page Classification

Web documents, after classification can be organized mainly into various structures. It depends on the requirements whether to arrange the documents in flat order or hierarchical order. Each of the structure has its own significance and purpose. For example news web pages can be represented as hierarchical way where a news page could be "sports", "politics" or "entertainment". Further "sports" page could be "cricket", "football" or "baseball" and "politics" page could be "national" or "international". In flat structure organization each page type is at same level.

Based on the organization of categories, Web page classification can be divided into flat classification and hierarchical classification. In flat classification, one category does not supersede another, while in hierarchical classification, the categories are organized in a hierarchical tree-like structure, in which each category may have a number of subcategories. An illustration as shown in Figure 1.7 will address the issue of hierarchical classification further.

- Flat Structure Classification: The categories like "education", "health", "business" and "sports", of web documents can be arranged in the forms of a flat categorization. Documents types are considered parallel. There is no further requirement of classification corresponding to each category as shown in figure 1.6.

- Hierarchical Structure Classification: In hierarchical classification category can

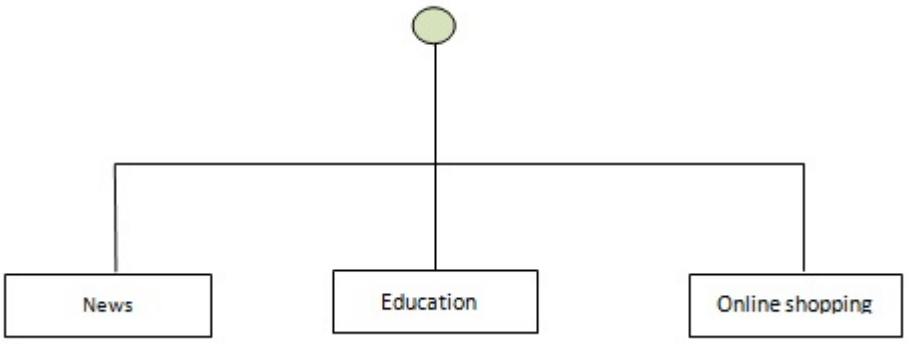


Figure 1.6: Flat classification

supersede the other category, one category can further be arranged into sub categories. The level can go up to meaningful arrangement. “Health” page can be arranged into “Physical Health” and “Mental Health”. Similarly “Sports” page can be arranged corresponding to various sports like “Indoor” and “outdoor”. “Indoor” sports pages can be further arranged as “Chess”, “Billiards” or “Table tennis” and “Outdoor” sports pages can be further arranged as “Hockey”, “Cricket” or “Football” as shown in figure 1.7.

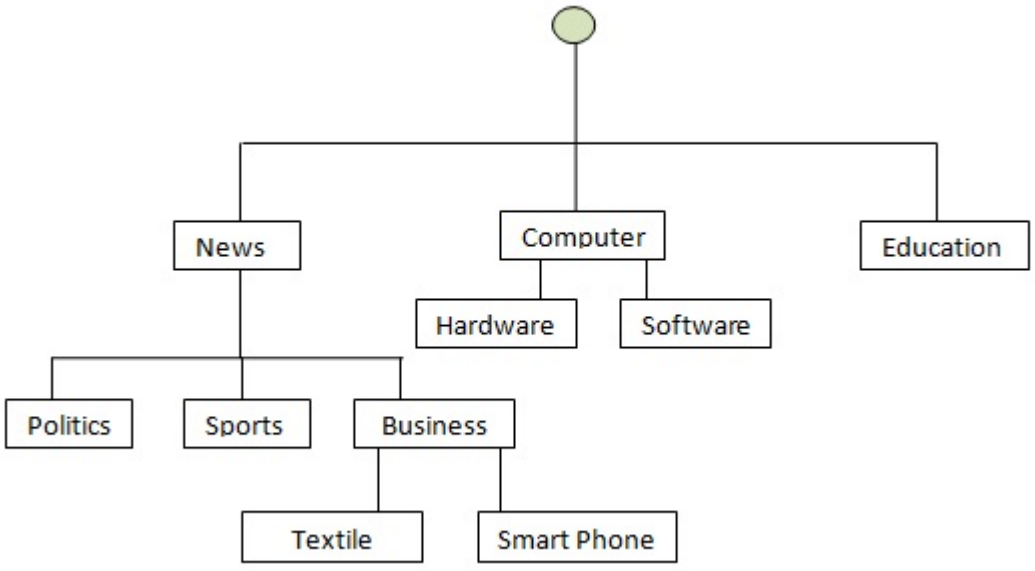


Figure 1.7: Hierarchical classification

1.3 Web Page Classification Processes

Classification process can be of two types based on the number of classes. Repository of web page can either be separated into either one of the two broad classes or one of the many predefined classes. Numbers of approaches are developed for both the processes. Classifiers are trained according to the need of classification. Features are chosen accordingly for the purpose of binary or multi-class process.

- Binary Categorization Process (Blum *et al.* [10]): It is relatively easier to classify web pages into exactly one of two categories. For example to arrange dataset to "Educational" and "Non Educational" pages or "Blog" and "Non Blog" pages as shown in figure 1.8.

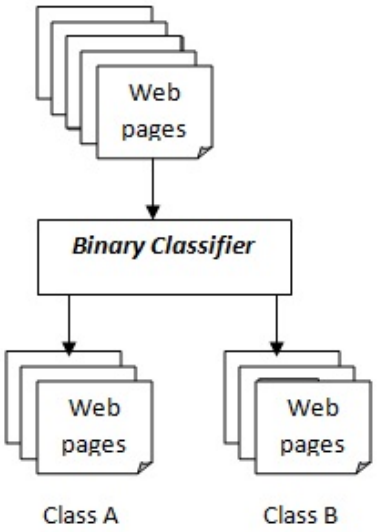


Figure 1.8: Binary classification

- Multi-class Categorization Process (Lin *et al.* [11], Deuk *et al.* [12]): Multi-class categorization requires many predefined classes. Web document repository is required to be arranged into one of many predefined categories. For example to arrange dataset to "News", "Health", "Research", "Education" etc as shown in figure 1.9.

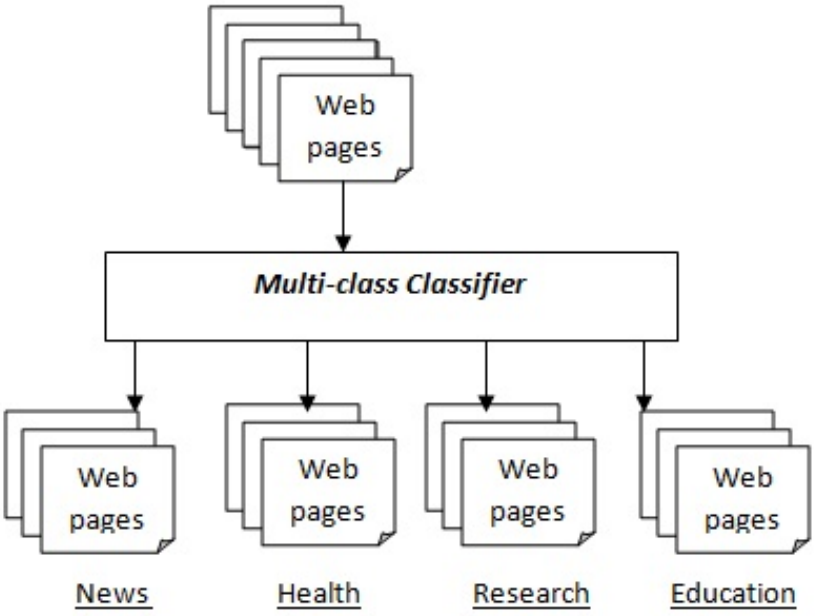


Figure 1.9: Multiclass classification

1.4 Web Page Categorization Classes

To accelerate socio-economic development in the world valuable knowledge and information needs to be organized in to more meaningful ways. By facilitating categorization of huge knowledge-base into critical ways with which growth in the developing world can be accelerated. Classification schemes can enable the publication, interlink and reuse of huge information in the form of web pages and valuable datasets. The general problem of Web page categorization can be further extended to more specific problems.

- Subject Categorization: Web pages can be broadly classified in to major subject categories to facilitate quick and easy retrieval of information. Subject categorization critical in managing billions of pages. Web page contents are analyzed by identifying the relevant feature which helps in deciding the topic of web document. It categorizes the web page according to its subject or topic. Examples of subject categorization are “software”, “hardware”, “poli-

tics", "sports" or "science". Subject classification enables web crawling easier and faster. Customized user queries can be efficiently satisfied by providing relevant results.

- Functional Categorization: The knowledge generated by the complex systems can be organized into functional categories according to its role. For example categorizing the web page as "research", "news", "information" or "entertainment" page is an instance of functional categorization. This type categorization enhances the clarity and the efficiency in knowledge representation. Functional categorization further facilitates the integrated use of various methods of spreading the contents and ways of learning.

- Sentiment Categorization: Web pages can be classified based on sentiments. Sentiment categorization distinguishes the web page according to the author's attitude about any particular topic. Sentiment analysis is performed to identify and extract subjective information in source web pages. Sentiment analysis is generally applied to social media and reviews for a large number of applications, ranging from survey, marketing, analyzing customer service reviews and movie reviews etc. Typically blogs are be classified using the sentiment categorization scheme to judge the mood of author such as "happy," " angry," and "sad". Sentiment analysis is applied on web pages to measure the emotional state of the author about the subject matter. Positive, negative, or neutral aspect is analyzed for feature selection.

- Genre Categorization: Genre categorization distinguishes the web page documents based on the genre of the document. Genre can be defined as a taxonomy that involves the content, form and style of a document which is inclined towards topic, with fuzzy classification to multiple genres. Specific features corresponding to genre are identified and selection methods examined for spreading documents based on genre types. Currently, search engines rely on keywords for matching documents to user queries. Possibilities are

continuously explored how to automate the classification of web documents according to their genres. Typically a newspaper articles genre include "sport news", "editorial", "entertainment news", and "political news". This means categorizes the web page with respect to its form or functional trait.

1.5 Text classification VS Web content classification

Web page content is different from traditional structured documents. First traditional text documents are written by following controlled structural and authoring styles. On the contrary, web pages are semi structured written in hypertext markup language (HTML). Web browsers render these documents for visual presentation to the users in impactful way. Semi-structured format of web documents are constructed using html tags. Finally, Web page hyperlinks connect to and from other documents. Hyperlink feature plays important and central role to the WWW definition but it is absent in typical text classification problems. This very nature of web enforces to apply nontraditional approaches for content classification. These all are good source of features which can be used for classification of web content in to various predefined categories. HTML tags can be explored to identify and select the unique special features present on page. Information from html tags can be extracted for the purpose of web page classification/categorization. Size of internet, dynamic nature of web documents and varied applications of web content makes web classification an important task, which is reasonably different from traditional text classification.

1.6 Features

Features play an important role in categorization. Generally classification of things is based on some specific and common features. The biggest challenge is to identify

domain specific feature by closely analyzing the domains. During this process some of the features are highly relevant and some are less useful. Careful examination of features helps in this distinction. Because too many features for a problem in hand may add complexities. So reiterate the list of features and select the most reliable ones and drop the others.

Features found to be useful in Web page classification research process are reviewed in this section on the basis of the following facts:

1. Web documents are developed in hypertext markup language (HTML), a script language.
2. Each web page is assigned a unique address called uniform resource locator (URL).
3. Textual content is presented using html tags.
4. Hyperlinks are integral part of web documents and used extensively.
5. Due to advancements in technology newer tags are added to HTML.
6. Web browsers render the HTML code for the view of end user.
7. Information on web pages is semi-structured.

1.7 Feature Sources

Features can be broadly classified into two categories. Features which are present on the web page are called “on-page features” and there are features which are present on the pages which are related with the page in some way are called “neighbor features”.

1.7.1 On-page feature sources

Textual content

Features that are directly located on the page in the form of textual content are the most prominent candidates for feature selection and extraction. Bag-of-words representation for all terms may not effectively achieve top performance due to inherent noises in the web documents. To efficiently use the textual content as feature researchers used N-gram representation. In this approach document is represented by a features vector, which includes not only single terms, but also up N consecutive words.

HTML Tags

HTML tags are other good source of feature extraction. HTML is predefined tag based language. Each tag is preprogrammed for the specific task. <Head> is used for page heading. <Title> tag is used for the title of web page. HTML <A> and </A> tags are used for setting hyperlinks along with anchor text. <img>, <audio> and <video> tags are used for displaying images and multimedia contents. and <li> tags are for displaying contents in form of lists. <Table> tag is used for presenting the information in form of tables. These HTML tags flag semantic content and are prominent candidate for feature extraction.

Web Page URL

URL (uniform resource locator) of web page can be another source of feature especially when the page contents are not available. Researchers successfully attempted the classification by tokenizing the URL string and extracting relevant information for classification. Feature set obtained through this method is computational less expensive. Though, the accuracy achieved through this method is not high but it eliminates the need of downloading the web content.

Visual Analysis

Web pages can be represented in two ways. Out of these, one way is the text representation weaved in HTML tags and the other way is the visual representation which is rendered by the code embedded in the web browsers. These both ways gives entirely different view of a web page. Large number of classification approaches stress on the textual representation and ignoring the visual information. Where as visual information is also very useful and it can play important role in deciding the category of a web page. Kovacevic *et al.* [13] developed a technique to use the visual information. Authors treated a page as hierarchical visual adjacency multi-graph and did visual analysis by applying heuristic rules. They were able to detect multiple logical areas corresponding to various sections of the web document. They used this information for classification and found improvement in the results.

Meta Tags

Web 2.0 onwards WWW is becoming more organized and professional. Web documents are being developed by experienced programmers. Web pages developed by skilled developers follow standards and specifications. Generally, latest web pages make use of meta tag in the form of meta <description>, meta<keywords>, meta<contents> etc. These meta tags can be another good source of features and may be explored for feature extraction.

Implicit Artificial Links

Web search engines can organize the result of user query according to the potential categories. It becomes convenient for web user to browse the query results. Shen *et al.* [30] proposed an approach based on connections among pages that appear in the same query result and user clicks those pages. They named these connections "implicit" links. They used query logs to explore both "implicit" and "explicit links" (hyperlinks) of pages to decide the category of pages. They used both type

of link information to compare the similarities generated by human insight and the ranking algorithm to demonstrate that implicit links can be useful for web page classification.

1.7.2 Neighbor's Features

Web pages are full of relevant information for generation of various feature sets for the purpose of classification. But sometimes, these features are incomplete, misleading and unrecognizable. Say, for example, in some pages textual contents is very less. These pages may have large images and multimedia objects. In these cases it is very difficult for the classifier to make the decision based on the on-page features. Shen *et al.* [14] proposed a scheme to use the information and class of neighboring pages. They used link graph to store such information and showed that linked pages were more likely to have terms in common. These features are considered as supplementary information for web document classification. Their research proved that (labels text, the surrounding text of anchor text, titles, headers and full content) along with on page feature can be used to reduce classification error. Neighbor pages features contribute towards classification in case pages contain less textual content.

1.7.3 Combining information from multiple sources

Qi and Davison [15], Calado *et al.* [16]; showed features can be extracted by combining information from various sources to generate a rich feature set. Researchers combined the information of <Title>, <Head>, <Body> html tags along with <Table>, <li>, <ol>, <form>, <text> term frequency. This information can be combined with the URL of document. Neighbor feature and link analysis can be added to feature set. <A ref> and <img> anchor text can be member of feature set. Furthermore, visual information, layout, structure of page, placement of links, styles and visual information rendered by the web browser can be combined to generate a detailed feature set. Analysis of synthetic and non synthetic images can also extend

the features. Researcher proved that every information source gives different point of view. Therefore, combined information has more potential and discriminative power.

1.8 Classification Algorithms

Automatic web page classification problem can be solved by training the classifier using the training datasets obtained through feature sets. Large number of classifiers and algorithms are available in the literature. Considering the goal, feature and type of classification scheme, one can choose the classification approach. Succeeding section explains about some of the available options.

1.8.1 Naive Bayes Classifier (NB)

Bayesian theorem based Classifier is known as Naive Bayes Classifier (NB). It can handle high dimensional input space and has ability to achieve higher accuracy. Naive Bayes classifiers can handle an arbitrary number of independent variables. For a given a set of variables, X = {x1,x2,x...,xd} and a set of possible classes, C = {c1,c2,c...,cd} posterior probability for the class Cj is calculated. using Bayes' rule (Zhang *et al.*[39]):

$$p(C_j|x_1, x_2, ..., x_d) \propto p(x_1, x_2, ..., x_d|C_j)p(c_j) \tag{1.1}$$

where p(Cj | x1,x2,x...,xd) is the posterior probability of class membership, means that the probability that X belongs to Cj. The probability can be decomposed as product of terms:

$$p(X|C_j) \propto \prod_{k=1}^{d} p(x_k|C_j) \tag{1.2}$$

and rewrite the as:

$$p(C_j|X) \propto p(C_j) \prod_{k=1}^{d} p(x_k|C_j) \tag{1.3}$$

Using Bayes' rule above, label a new case X with a class level Cj that achieves the highest posterior probability.

There is assumption in Naive Bayes that the conditional probabilities of the independent variables are statistically independent. But it is not always accurate. it can make classification work significantly simplifier. Naive Bayes reduces a high-dimensional density estimation task to one-dimensional kernel density estimation. it means the class conditional densities $p(xk|C_j)$ is calculated separately for each x_i. The assumption mentioned above does not seem to significantly affect the posterior probabilities, particularity, in the regions near to decision boundaries. Thus, classification task remains unaffected.

1.8.2 K Nearest Neighbor (K-NN)

Knn is considered as a lazy learning algorithm. The decision is generally postponed beyond the training examples till a new query is encountered. To classify a new class, locate its nearest neighbors from the training dataset. Euclidean distance, Mahalanobis distance, Minkowski distance equations can be used to calculate the distance. Standard Euclidean Distance defined as

$$d(x_i, x_j) = \sqrt{For\ all\ attributes\ a \sum (x_i, a - x_j, a)^2} \tag{1.4}$$

Steps followed in the KNN algorithm are following: for each training example, add the example to the list of training examples. Let xq is a given query instance and { x1 ,x2. . . .xk} denote the k instances from training examples that are nearest to xq . Choose the class that represents the maximum of the k instances. Kwon *et al.* [17] developed a technique using KNN algorithm that uses the term co-occurrence in

documents for improved similarity measure. Two documents have stronger relation if there are more co-occurred terms in common Their experiments showed performance improvements of the new similarity measure over cosine similarity and inner product measures. Yu *et al.* [18] gave new meaning to k-Nearest Neighbor algorithm with probability computation. The probability of a web document "d" being in class "c" is determined by its distance between its neighbours and itself and its neighbours' probability of being in class c.

1.8.3 Support Vector Machines (SVM)

The Support Vector Machines (SVM) method classifier was introduced by Vapnik *et al.* [19,20] The SVM method map the documents into a high dimensional feature space, and train a separating hyperplane, that generates the widest margins between two different types of class documents. SVM supports various kernel functions-linear, radial bias, polynomial and regression. Based on nature of problem, these kernels can be trained. Tuning of various SVM parameters can improve the width of separating hyperplane resulting in improved efficiency. SVMs are suitable for both categorization and regression. SVM employ non-linear class boundaries by mapping input vectors non-linearly into a high dimensional feature space by putting linear model into place. Support vectors are the training examples that are ceiling periphery of hyper plane as shown in figure 1.10. Other training set examples is not significant for setting up the class boundaries of the binary classification system.

SVM use Lagrange multipliers to translate the problem of finding this hyperplane into an equivalent quadratic optimization problem (QP). The SVM method is considered the state-of-the art in text categorization and highly competitive in classification accuracy among the several classification approaches available.

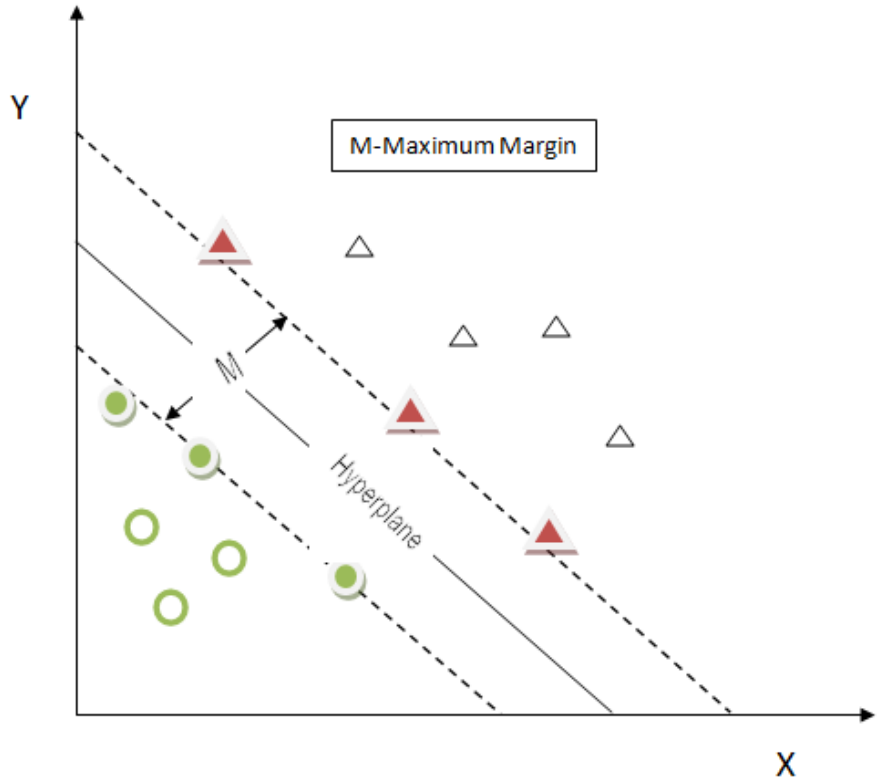


Figure 1.10: Maximum margin hyper-plane- a straight line separating classes.

1.8.4 Genetic Algorithms (GA)

Application of GAs to Web page classification starts with Calado *et al.* [16] who has proposed a Web page classifier of "if condition then classi" rules. Pietramala *et al.* [9] have introduced a GA, called Olex-GA, for the induction of rules of the form "classify document d under category c if ($T_{1\ 2\ d}$ *or...or* $T_{n+1\ 2\ d}$) and not ($T_{m+1\ 2\ d}$ *or...or* $T_{n+m\ 2\ d}$) holds, where each T_i is a term". In the study, only inclusion or exclusion of a term is considered, and weight computation for terms or HTML tags are not performed. Bai *et al.* [21] have used rough sets and GA to reduce the dimension of feature vectors and then applied support vector machine (SVM) to classify news dataset. Experimental results indicated that GA-based feature selection improves the performance of traditional SVM classifier. Zhang *et al.* [22] proposed a semi-supervised fuzzy clustering algorithm based on a GA. Both labeled and unlabeled documents are taken together to obtain a classifier. The GA fitness function developed by researcher is based on the minimization of a combination of

"fuzzy within cluster variance on unlabeled documents" and "misclassification error of labeled documents".

1.8.5 Neural Network (NN)

A neural network is a machine learning technique and comprised of processing elements called neurons. They can be used to solve categorization. The weight vector $w_i = [w_{i1}\ w_{i2}\ \ldots\ w_{in}]^t$ increases in proportion to the product of input x and learning signal. Neural networks have been widely applied by many researchers to classify the text documents with different types of feature vectors. For example, Kwon *et al.* [17,23] have proposed a neural network for document classification based on a linguistic feature selection with a fuzzy learning technique. Liu and Zhang *et al.* [22] have proposed a term frequency method to select the feature vectors for document categorization using neural networks. Furthermore, Enhong *et al.* [24] have proposed a feature extraction method based on a word semantic and the categories of words co-occurrence analysis for document classification using neural networks.

1.8.6 Ant mining

Ant mining approach is inspired by the nature of ants during search of food. Ants generally follow the shortest route to reach up to food. Ants are able to do this because ants lay down a scent called "pheromone". Each ant follows the strongest scent and tends to create the shortest path for fast and easy traversal. Amount of pheromone acts as positive feedback to other ants and they change their path to follow shorter path resulting in more scent on the shorter path. Nicholas *et al.* [25] applied ant colony approach to solve the problem of a web page content mining and discovery of knowledge in compact form. They attempted to reduce the large number of attributes required for web classification by applying ant colony algorithm

1.8.7 Relational Learning (RL)

Relational learning was first used in the field of image processing and vision analysis. Later researcher found its use in artificial intelligence. Relational learning technique is also applied to web mining problems. Sen *et al.* [26] proved that web page classification can be solved as a relational learning problem. They developed an approach based on relaxation labeling algorithm by assigning class probabilities to each page in text classifier. Trained classifier model, in turn re-evaluates class probabilities of each page in correspondence to the latest estimates of the class probabilities neighboring pages. Angelova *et al.* [27] developed slightly different approach in which all neighbors are not considered. They used content similarity to pick the neighbors.

1.8.8 Comparison of approaches

There are a number of approached discussed in the literature for the purpose of classification. To select the reasonable algorithm for solving the problem at hand, researchers need to take care of the following dimensions- Size of training data, Dimensionality of the feature space, whether problem is linearly separable, whether features are linearly separable, features are independent, over fitting is expected to be a problem and system's requirement in terms of speed/performance/memory usage.

Advantages and disadvantages of machine learning algorithms:

Generally, accuracy is the main concern for user. So he should try different classifiers by considering the advantages and disadvantages of algorithms in various situations and use cross-validation to select the best algorithm for problem at hand.

- Decision Trees are non-parametric, so don't have to worry about whether the data is linearly separable. They are easy to evaluate and interrupt. Moreover

they are fast to train. But, they can stuck in local minima. They can even over-fit and may need random forest to help reduce the variance.

- Support vector machine gives theoretical guarantees regarding over-fitting. It can easily handle high dimensionality of feature space. It is best suited for text categorization problems where very high-dimensional spaces are the norm. But selecting the appropriate kernel can be a challenge.

- Neural network(NN) may have any number of outputs, while support vector machines have only one. The most direct way to create an multiclass classifier with support vector machines is to create n support vector machines and train each of them one by one. Whereas n-ary classifier with NN can be trained in one go. But NNs are slow to converge and hard to set parameters. NNs can suffer from multiple local minima, the solution to an SVM is global and unique which is a significant advantage of SVM. Moreover, in case of NN choosing the correct topology is difficult; Training requires a lot of data and long duration in comparison to SVM.

- Bayesian classifiers are relatively easy to understand. Naive Bayes classifier requires less training data and may converge quicker in comparison to discriminative models. But it's very simple representation doesn't allow for rich hypotheses, Assumption regarding independence of attributes is too constraining.

- Genetic programming (GP) has some specific advantages that no analytical knowledge is required and yet it could produce higher accuracy of results. But it is hard to program and GP approach does scale with the problem size. And it also put restrictions on how the structure of solutions should be formulated.

- kNNs attract the machine learners because they are intuitive and simple. It can work well with small data sets. However, with large amount of data, it has significant computational overhead. It tends to work better in low-dimensional

spaces. SVM and NN has advantages over kNN. Both can perform better in high dimensional space in comparison to kNN.

- Ant colony can be advantageous for dynamic applications were positive feedback helps to fast discovery of good solutions. It advantage is distributed computation avoids premature convergence and convergence is guaranteed, but time to convergence is uncertain and coding is not straightforward.

1.8.9 Applications of Web Categorization

Researchers working on the problem of web page classification have identified many requirements and application areas of web content classification. It is needed for the following reasons:

- Web page classification is required for the efficient retrieval of web pages. Chekuri *et al.* [28] proposed a technique for automatic web document classification in order to improve the precision and quality of web search. Kaki et al [29] also proposed an approach to present a categorized view of search results to end users.

- It provides an aid to topical crawlers which search the web for a particular topic. Chakrabarti *et al.* [6] proposed an approach called focused crawling, in which only documents relevant to a predefined set of topics are of interest.

- It is required to construct, maintain and expand web directories. Nie *et al.* [8] developed another web ranking algorithm that is based on the topics of web pages. Kohlschutter *et al.* [30] performed analysis on open directory project (ODP) [53] categories, and demonstrated that ranking performance increases with the ODP level up to a certain extend. Huang *et al.* [31] also developed a technique based on web corpora and user defined hierarchies and showed that with advanced techniques customized or dynamic view of web directories can be automatically generated-ok

- It helps to increase the quality of search results. Categorized results present a good user interface than the search results which are presented in a ranked list. A question answering system may use classification techniques to improve its quality of answers. Chua *et al.* [7] proposed a technique to find answers to the list questions where a set of distinct entities are expected through web page functional classification.

- According to, Chen *et al.* [32] web page classification is also useful in web content filtering. They used both text and images present on the page for the purpose of content filtering

- Contextual advertising can also be achieved by web document classification. Broder *et al.* [33] proposed a technique for the same.

Next chapter discusses various existing related techniques for web page categorization. The relative advanatages and disadvanatages of various techniques are also discussed in the next chapter.

Chapter 2

Literature Review

Literature review illustrates large number of approaches [34–40] developed for solving the problem of web page classification. Researchers attempted the problem from varied prospective and purposes. Study of these approaches conclude that to solve web classification, there are three important aspects data in the form of domain web pages, features and weights and a classifier for training and testing as shown in figure 2.1.

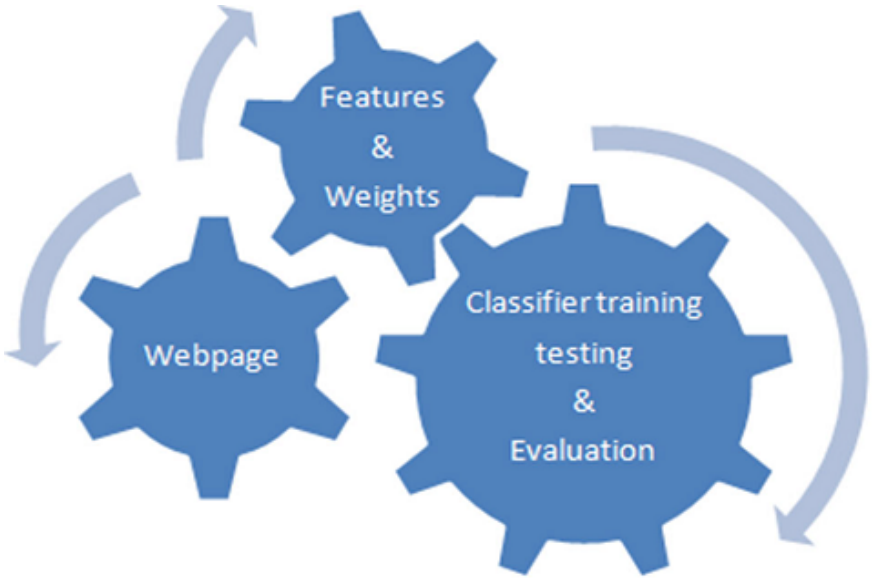


Figure 2.1: Web page classification aspects

Researcher attempted web page classification based on binary classification [17, 36, 39], multiclass classification [19, 40, 41] and hierarchical structure classification

[42–43]. Classifier used in some of these techniques are K-Nearest Neighbor approach [17, 37], Bayesian probabilistic models [39], Support vector machines [38, 40, 41], HMM [35], decision trees, Genetic algorithms [9, 22], neural network [36, 44] and Ant mining [28] with variety of features. A number of text categorization algorithms also have been applied to the problem of web page categorization. Feature used by researcher are URL[45], link analysis [45–47], <Body text> [45, 48, 49], <Title> text [45, 48, 50], <Head> text [49, 51], <Meta> tag data [46, 48, 50] in the form of meta tags <meta keyword>, <meta description> or combination of few or all. Some researchers used neighboring page features along with on page features [16]. Many of these techniques are found to be costly, time consuming and does not produce prominent results [53–56].

Researchers in the past attempted the web classification problem by considering a variety of features from the web pages. These feature sets are then used to classify various web pages in to different classes using various machine learning techniques. Various attributes considered in these techniques are explained as follows. Summary of the same is listed in the table 2.1 and 2.2. Riboni *et al.*[48] proposed a new method for web page categorization by careful selection of more than 100 documents from 10 different categories from the yahoo directory. They have introduced a novel method for hypertext categorization for real-time applications.

Sun *et al.*[38] introduced a new method for web page categorization using text and context-aware feature set. The classification efficiency has been increased to a large extent using the proposed scheme given by the authors. Yang *et al.*[50] proposed a technique in which they have analyzed the impact of five hypertext properties presence of which greatly influences the web page categorization. They have also provided a detailed analysis of various classifier such as-Naive bayes, k-NN, and first order learner. With respect to the size of the dataset, the relative performance of these three types of classifiers are identified in the proposal.

Benbrahim *et al.* [45] described an empirical study of hypertext categorization of web pages. To improve the classification task, authors have introduced a unique technique for information retrieval and classification at various levels. Lim *et al.* [46] have proposed a new scheme in which multiple set of features are included for automatic generation and classification of various web pages. Various features are included for web pages classification and the performance of the proposed scheme was found satisfactory with respect to various evaluation metrics.

Cakrabarti *et al.* [57] proposed an enhanced hypertext categorization using hyperlinks. Authors have designed novel algorithms for web page classification and performance of the designed scheme was evaluated using various evaluation metrics where its performance was found satisfactory. Chen *et al.* [58] proposed a SVM-based scheme using weighted voted schema for web page categorization. Authors have used an efficient weighted schema for feature selection which are manually used for putting the web pages in a specific category. The proposed scheme was found efficient with respect to the selection of various metrics.

Attardi *et al.* [47] and Choen *et al.* [59] have proposed a new web page categorization technique using link and contextual information. This reduces the overhead and complexity of the proposed solution in this environment. Their schemes were found satisfactory with high level of accuracy, and precision. But, the relevant tags were not used for classification of web pages in these schemes. Some other techniques are also discussed in (Hodgson *et al.* [51]; Quek *et al.* [49]) in which semantic information about the web pages was used to classify the web pages.

Yang *et al.* [60] proposed a technique to make web page categorization more effective in which text, contents, and hyperlinks are used as the features for context-aware information retrieval. The proposed scheme performs better than the other existing schemes in literature.

Table 2.1: Various Researchers and techniques

Researcher Tag	Researcher	Technique
A	Riboni *et al.* [48]	Perceptron classifier
B	Sun *et al.* [38]	SVM
C	Yang *et al.* [50]	Study of approaches
D	Benbrahim and Bramer [45]	Naïve Bayes, Knn, SVM C4.5 classifier comparision
E	Lim *et al.* [46]	TIMBL tool- Memory Based Learning-URL & HTML
F	Cakrabarti *et al.* [57]	Hyperlink analysis
G	Chen and Hseih [58]	SVM- Weight vote Schema
H	Chen *et al.* [3]	Fuzzy ranking analysis-Discriminative power measure
I	Attardi *et al.* [47]	Link and content Analysis
J	Hodgson [51]	Internet computing
K	Quek *et al.* [49]	SVM, Naïve Bayes
L	Cohen [59]	Machine Learning
M	Yang [60]	Fusion approach-combinning multiple sources
N	Brin and Page[61]	URL, Link anchor, Page rank, SEO
O	Furkranz [52]	Structural Information
@	Proposed Work	SVM, Wordstemming, Domain keywordlist

Brin *et al.* [61] has used an efficient classifier for large scale hypertext information retrieval from the web pages. The proposed scheme has enhanced performance than the other schemes of its category. Structural information was used for web page categorization in (Furnkranz *et al.* [52]), the performance of which was found satisfactory with respect to various evaluation metrics. Herndez *et al.* [62] proposed a new unsupervised URL-based web page categorization technique called as CALA. Authors have varied the learning rate and then evaluated the performance of the designed scheme where its performance was found satisfactory in comparison to the other schemes of its category.

Uzun *et al.* [63] proposed a hybrid approach for information extraction from the web pages. The proposed scheme was evaluated with respect to various evaluation metrics by mixing the contents to be searched out. Recently, Murthy *et al.* [64] proposed URL classification based on the semantic structure for web mining applications. Du *et al.* [65] proposed multi-view semi-supervised web page categorization using web mining applications. Authors have introduced a multi-view of the web page classification algorithm and evaluated it in different network scenarios where its performance was found satisfactory.

Moreover, more related work exists in the literature addressing various aspects of web classification. These proposals are summarized as follows. Levering *et al.* [73] proposed the usage of visual features for extraction of visual features of the web pages. The results obtained confirm the effectiveness of the proposed scheme by including the visual features in the web page.

Gao *et al.* [97] proposed a SVM-based tool for web page classification along with genetic algorithm (GA) to improve the convergence rate and accuracy of the web page classification. The results obtained confirm the effectiveness of the proposed solution by taking various evaluation parameters. Chen *et al.* [3] proposed two-level scheme using fuzzy ranking mechanism. They have computed relevance measure, discriminating power measure (DPM) to increase the efficiency of web page classification. Two different data sets are selected to measure the effectiveness of the proposed solution where its performance was found satisfactory.

Burget *et al.* [74] investigated that how various features such as advertisement, copyright notices aspect the overall performance of any web page classification. In the proposed solution, firstly basic visual blocks are detected in the page and in the second phase, the purpose of these blocks is guessed based on their visual appearance.

Eickhoff *et al.* [96] proposed web content classification by considering the aspect of topical and non-topical web page for child suitability. Child's psychology and cognitive sciences are considered while taking the decision about the web page classification. The results obtained in this approach were found satisfactory in comparison to the other existing schemes. Vaughan, *et al.*[75] proposed a method which takes parameters based upon the patterns of various types of web sites. The proposed method performed an accurate web page classification. Using the proposed method, the business model was constructed in the proposed scheme and its performance was found satisfactory.

Unler *et al.*[76] proposed a discrete particle swarm optimization (PSO) algorithm for the feature classification and selection for web page selection. Authors have tested their proposed scheme on the publicly available datasets and obtained results are comparable to the tabu search and the scatter search algorithms for classification accuracy. The performance of the proposed scheme was found better than similar existing solutions in literature.

Ozel, *et al.* [77] proposed a Web page classification using genetic algorithm. In the proposal, HTML tags or terms are considered for feature selection. By the usage of both HTML tags and terms in each tag improve the accuracy and reaches up to 95%. Sun *et al.* [78] proposed an entity-based co-training (EcT) algorithm. A new learning module is proposed in the designed algorithm such that unlabeled data can also be classified which improves the overall performance of the designed solution. The proposed solution requires no prior knowledge about class distribution for web classification.

Sabbah, *et al.*[79] proposed dark web classification hybridized feature selection method using a weighting technique. The proposed technique can be applied for

accurate terrorism activities detection in textual contexts. The experimental results prove the effectiveness of the proposed solution. Kim *et al.*[80] proposed an effective method to mitigate the effects of various intrusion detection activities in the network. Authors proposed novel techniques for identification of false webpage by using an efficient classifier. The proposed solution has good detection rate of 90.4% with low false positive rate of 0.2%.

Wang *et al.* [81] presented a classification method for less popular web pages. Authors have used two different aspect, i) latent semantic analysis (LSA), and ii) density-relation-based rough set model. The proposed solution performance was found good compared to the other solutions. Lee *et al.* [17, 23] have worked on fuzzy learning and used linguistic feature selection. They classified document based on selection and proposed a neural network based on the features. In another work, frequency method was used by Liu and Zhang [22] to select the feature vectors for document categorization using neural networks. Furthermore, Enhong *et al.* [24] also used neural networks for document classification and feature extraction. The authors categorized the words and used co-occurrence analysis and word semantic for extracting features.

Freitas [25] proposed a web page classification technique which is based on Ant Colony Algorithm. A number of text preprocessing techniques based on linguistics were investigated for their benefits and dangers to reduce the large numbers of attributes associated with web content mining and discover knowledge in a much more compact form. Pietramala *et al.* [9] have proposed Olex-GA, which is another version of GA, for the induction of rules for classification of documents. The authors did not perform weight computation and only considered inclusion and exclusion of terms.

Bai, Wang, and Liao [21] classified news dataset using rough sets and GA. The

authors applied support vector machine (SVM) and reduced the dimensions of feature vectors. The experiments of the scheme resulted in improvement over traditional SVM classifier and proved the superiority of GA-based feature selection technique. Ozel *et al.* [77] proposed technique which used GA and best weights were implemented for each feature. Results prove that there is considerable improvement in classification accuracy when tagged terms are used as features. The classifier worked better than other classifiers even when the training data set had negative documents and achieved high accuracy.

Calado *et al.* [16] and Qi and Davison [15] showed an improvement even in the presence of orthogonal information by combining different sources of information. The combination of link and content information is mostly used in web classification. Multiple classifiers are trained after combining information from multiple sources where each information is treated as disjoint feature set. Final decision is produced after combining the classifiers.

Blum and Mitchell [10] achieved higher accuracy by using both labeled and unlabeled data. To classify unlabeled instances a binary classification is used where training on different data sets is done for two classifiers. The training of one classifier is done based upon the prediction of other classifier. This approach significantly reduced the error rate as compared to approach that used only labeled data. The approach was generalized by Yang and Ghani [50] to multi-class problems.

Effectiveness of document classification into large scale taxonomies using SVM was studied by Rung-Ching *et al.* [77]. Neither hierarchical SVMs nor flat SVMs had satisfying results for classification into large taxonomies. Kwon and Lee [24] developed an improved similarity measure based on k-Nearest Neighbour classifiers. The co-occurrence of terms had a constraining effect of semantic concept and documents having more such terms had stronger relationship amongst them. Ex-

Table 2.2: On-page Candidates considered for feature selection refer table 2.1.

Feature set	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Title <title>	x	x	x	x	x							x				
Meta Tags <meta>	x	x		x	x						x	x				
Head <head>	x				x						x	x				
Headings <h1..h6>	x					x					x	x				
Link tag	x		x	x	x	x	x			X	x	x	x	x	x	x
<body> Text		x	x		x	x		x	x	X		x		x	x	
Meta Tag <keyword>	x	x		x	x						x	x				
Meta Tag ¡description¿	x	x		x	x						x	x				
<img src> anchor text	x															
<a href> anchor text	x											x				
Multimedia <video> anchor text	x															
Webpage URL	x					x										
Lists , ,	x															
<dt>	x															
Table headings <table>	x															
Genre Attributes Counts –links, images, video, audio, table, list(ol,ul) etc.	x															
Image name, video name, audio name at <src>	x															

periments proved the superiority of proposed concept as compared to inner product similarity and cosine similarity.

Furthermore, comparison and summarization of various web page categorization techniques and approaches is done in the following part of literature survey as shown in table 2.3.

Yong-Bae *et al.* [85] used inverse document frequency and term frequency to reveal the genre and subject of the web page. They applied naïve Bayesian classifier to classify the web pages into genre and subject. They achieved the accuracy of 73 %. Xiaogang *et al.* [86] applied hierarchical feature propagation and single path classification algorithm to classify the web apges in dynamic and hierarchical way. They used unique feature tags which are weighted and propagated upwards. These propagated tags become the feature of parent category. Accuracy of their algorithm came out nearly 78%. Schenker *et al.* [87] proposed a technique in which they used document similarity rather than a set of extracted features by applying extension of the k-Nearest Neighbor method to work with data represented using graphs rather than numerical feature vectors. Approximately 75% accuracy is obtained through this technique.

Shen *et al.* [88] proposed an approach based on bag of words and document frequency. They applied Adapted Luhn's Summarization Method and SVM as classifier. Their results show 72 % accuracy.Kan *et al.* [89] developed a technique which is based on the concept of maximum entropy. They relied on the url of web page instead of other kind of feature. They obtained low accuracy which is nearly 52 %. This technique can be good if web page contents are not available but it is not quite reliable technique. Liu *et al.* [94] developed a graph based technique. They used html tags as features and applied wweight learning method, label propagation algorithm which resulted in appr. 65% accuracy.

Devi *et al.* [90] proposed an approach which is similar to Kan *et al.* [89] in a sense they also used url of web page as feature. They applied Naïve Bayesian classifier and support vector machine as machine learning methods. They also could not obtained higher accuracy. But it is computationally less expensive. Yin *et al.* [91] proposed an approach based on social tagging graph. They used social tags as feature and a new link structure between objects and tags is explored for classification. They

Table 2.3: Comparison of Various Web page Classification Techniques

S.No	Researcher	Classifier	Features	Technique	Accuracy ~ %
1	Yong-Bae *et al.* [85]	naïve Bayesian classifier	Doc freq term freq(ID/TF)	Genre-Revealing and Subject-Revealing	73
2	Xiaogang *et al.* [86]	Hierarchical Feature Propagation Single path classification algorithm	Unique feature tags - weighted and propagated upwards and become the features of the parent category	Dynamic and Hierarchical Way	78
3	Schenker *et al.* [87]	Graph Model	Document similarity rather than a set of extracted features	Extension of the k-Nearest Neighbor method to work with data represented using graphs rather than numerical feature vectors.	75
4	Shen *et al.* [88]	Summarization& SVM	Bag of words, Document frequency	Adapted Luhn's Summarization Method	72
5	Kan *et al.* [89]	SVM & ME	URL as Features	Maximum entropy (ME)	52
6	Liu *et al.* [94]	Graph-based	HTML tags	Weight learning method, label propagation algorithm	65
7	Devi *et al.* [90]	Naïve Bayes & SVM	URL as Features	Machine learning methods	38
8	Yin *et al.* [91]	Social Tagging Graph	Social tags	A new link structure between objects and tags is explored for classification	79

succeeded in obtaining 79 % accuracy.

Further, on investigating various support vector machine (SVM) based techniques following information is obtained and summarized in table 2.4.

Sun *et al.* [38] proposed an approach using support vector machine as classifier. They used selective html tags for feature extraction by filtering body tag text, title tags text and anchore text. Weights are assigned to the feature. Their approach resulted in approx. 58 % accuracy. Liang *et al.* [41] proposed an approach which is based on word frequency ID/TF as feature. This approach scans web page by keyword dictionary and calculates the frequency of each keyword. Researchers used SVM to train the multiclass classifier and obtained accuracy of 82%. Zou *et al.* [56] developed an SVM based technique for classifying chinese web pages using ID/TF as feature. Statistics(CHI) based on IDF and the LI-normalization are then applied on the frequency vector to produce the feature word vector. They were successful in getting accuracy nearly 86%.

Punera *et al.* [92] developed a technique for hierarchical classification using support vector machine as a classification tool. Html tags, Document Taxonomies and Binary tree T are used for hierarchical classification and achieved approx. 76% accuracy. Ching *et al.* [58] proposed a scheme using SVM which is based on latent semantic analysis(LSA) to find frequency of words using a weighted vote schema. Accuracy of their scheme is approximately 74%. Xue *et al.* [95] developed SVM based classification technique using Html tags-Title, body, head, meta tags as features. They performed comparison on the polynomial kernel function and the radius basis function (RBF). Through this technique they were successful in obtaining 84 % accuracy in some cases.

Sun *et al.* [78] developed a unique technique which is based on entities. They applied entity-based co-training and SVM on the dataset and were able to achieve approx 80 % accuracy.

Table 2.4: Comparison of Various SVM based Web page Classification Techniques

S.No	Researcher	Classifier	Features	Technique	Accuracy ∼ %
1	Sun *et al.* [38]	SVM	Html tags Text, text+title, Anchore	Tag extraction and weight assignment	58
2	Liang *et al.* [41]	SVM Multi classifier	word frequency ID/TF as feature	Scans web page by keyword dictionary and calculates the frequency of each keyword	82
3	Zou *et al.* [56]	SVM	Chinese Web Page ID/TF as feature	Statistics(CHI) based on IDF and the LI-normalization are then applied on the frequency vector to produce the feature word vector	86
4	Punera *et al.* [92]	SVM	Html tags, Document Taxonomies	Binary tree T Hierarchical Classification	76
5	Ching *et al.* [58]	SVM	Latent semantic analysis used to find frequency of words	Using a weighted vote schema	74
6	Xue *et al.* [95]	SVM	Html tags-Title, body, head, meta tags	comparison on the polynomial kernel function and the radius basis function (RBF)	84
7	Sun *et al.* [78]	SVM	Entities	Entity-based co-training	80
8	Godoy *et al.* [93]	SVM	Social tags	Personalized tag-based resource classification	73

Godoy *et al.* [93] proposed an approach for resource classification using support vector machine as learning algorithm. Their approach is based on social tagging. Personalized tags are used as features for resource classification. They obtained 76 % accuracy for categorizing the resources.

Researchers in the past extensively applied two commonly used Naive Bayes (NB) and Support Vector Machines (SVM) algorithms in machine learning to solve the problem of webpage classification. NB algorithms are probability based classifiers with strong assumptions that all features are completely independent and relatively easy to implement. While algorithms using this assumption are unable to do complex associations between features, they can still be highly effective classifiers. In comparison to the simplistic NB algorithms, SVM are exceedingly difficult to implement. SVMs work by translating non-linearly classifiable feature data into a higher-dimensional hyperplane where it is possible to classify the data in a simple, linear manner. Less training data is required in case of NB classifier and may converge quicker in comparison to discriminative models. But it's very simple representation doesn't allow for rich hypotheses, Assumption regarding independence of attributes is too constraining. SVMs work by translating non-linearly classifiable feature data into a higher-dimensional hyperplane where it is possible to classify the data in a simple, linear manner. SVMs are less prone to overfitting. Both algorithms are known to be highly effective classifiers, and are able to achieve impressive accuracy in spam email, blog, web page classification. Further, Weather forecasting, Speech recognition, Character recognitions, Stock market prediction, Medical diagnosis problems can be solved both mathematically and in a nonlinear fashion using SVM and NB classifier. The difficulty of solving such problem mathematically lies in the accuracy and distribution of data properties and model capabilities.

After the critical analysis as depecited from the literature survey discussed as above, there is a great need for designing a new solution for the web page categorization to improve the accuracy and response time of any implemented solution in this domain. The next chapter discussed the binary web page classification.

Chapter 3

Binary web page classfication

3.1 Introduction

With an evolution of Internet and related technologies, the number of the Internet users grows exponentially. These users demand access of relevant webpages from the Internet within fraction of seconds. To achieve this goal, there is a requirement of an efficient categorization of web pages contents. Manual categorization of these billions of web pages with respect to the high accuracy is a challenging task. Most of the existing techniques reported in the literature are semi-automatic. Using these techniques, the higher levels of accuracy can not be achieved. To achieve these goals, this chapter proposes an automatic web pages categorization into domain category. The proposed scheme is based on identification of specific and relevant features of the web pages. In the proposed scheme, first extraction and evaluation of features are done followed by filtering the feature set for categorization of domain web pages. A feature extraction tool(FET) based on HTML-DOM of web page is developed in the proposed scheme. Feature extraction and weight assignment are based on the collection of domain specific keyword list developed by considering various domain pages. Moreover, keyword list is reduced on the basis of *ids* of keywords in keyword list. Further stemming of keywords and tag text is done to achieve higher accuracy. An extensive feature set is generated to develop a robust classification technique.

The proposed scheme was evaluated using a machine learning method in combination with feature extraction and statistical analysis using support vector machine kernel as classification tool.

Huge corpa of web pages present on Internet poses a challenge to extract the relevant information. Users of web can use this relevant information present on web for large number of applications. Classification of web documents can help to reduce the efforts and computation time to find the necessary information. For that matter, an efficient automatic classification scheme is required, so that the database repository should be able to satisfy the queries raised by the end users for particular web page's contents (Boser *et al.* [19]; Cortes *et al.* [20]; Chang *et al.* [82]; Chapelle [84]; Hsu *et al.* [83]).

Users can find the relevant web pages from a large distributed database repository using various popular search engines such as Yahoo, Google, and Bing. But, to identify the relevant features of the web pages for a particular domain is a challenge and especially when the repository is too big to handle manually. Though these search engines have made the task easy for searching the relevant contents but, these are generalized in terms of content searching. These search engines find the relevant contents based upon the keyword search which may result in irrelevant information retrieval at some point of time. Also, these search engines use various classification methods before crawling and searching. In view of the above, there is a requirement of automatic tools to perform the targeted categorization of web contents for desired level of accuracy.

3.1.1 Motivation

The domain web pages have some unique set of features (Riboni [48]) that distinguishe them from others. To perform the feature selection process, a known set of approximately 100 domain specific web pages are closely examined to identify the

feature set. In this process, few common keywords related to domain web pages are separated. This research is focused on specific tags that may contain information regarding the domain category (Riboni [48]). Firstly, some prominent tags are suggested that may be suitable for this task, e.g., <head> Text </head>, <Title> Text </Title>, <li> Text , <ol> Text , <img>, <Table> headings </Table>, <ahref> (Attardi *et al.* [47]; Cakrabarti *et al.* [57]), <img src> anchor text, number of synthetic images on web page, number of un-synthetic images, number of forward links on the web page (Frnkranz [52]).

Although, there are many existing solutions in literature for web page classification as discussed above but, these solutions are inadequate for classification of web pages with respect to multi-attributes criteria. Also, as new tags are introduced in the latest version of HTML and web pages are developed in a professional manner. These new tags also flag relevant information for classification. Hence, there is a requirement of more enhancements in the existing solutions so that desired level of accuracy can be obtained. Motivated from these facts, this chapter proposed a multi-attribute based classifier for classification of web pages using which the reliability and accuracy of the existing methods for web page classification can be improved. All these factors lead us to develop a robust technique for classification to enhance the precision and accuracy.

3.1.2 Contributions

Based upon the above discussion, following contributions are presented in the chapter.

i) A multi-attribute -based criteria is selected for web pages classification by choosing various features as given in Table 2.2.

ii) Feature set considered in the proposed scheme is extensive in comparison to existing schemes. Taking into consideration the latest tools and technologies with respect to the advancements in the html tags, more tags need to be

explored which were not either part of html tag list or hardly used especially images, multimedia content, tables, menus, forms, number of links, extra meta data. Introduction of these tags as feature can help in increasing the confidence and accuracy of machine learning model.

iii) An efficient algorithm for feature extraction and weight assignment is designed.

iv) The performance of the designed algorithm is evaluated on a large training data set where its performance was found satisfactory in comparison to the other existing schemes of its category.

3.2 Classification Types

Mainly, there are two types of web page categorization (Eickhoff *et al.* [96]) defined as follows.

i) Binary categorization (Educational/non-educational or commercial/non-commercial)
It categorizes page into one or more categories as follows.
Education(1.0)/Course(1.0)/ Student(1.0)/Faculty(1.0)/Staff(1.0)/Department(1.0).

ii) Multiclass categorization (Sports page-Cricket/Football/Hockey/Baseball)
It categorize the page into particular category and then further into specific class.
News(1.0):- Politics (0.3) + Entertainment (0.4) + Sports (0.2) + Health (0.1)

Generally, binary categorization is done for major categories and multiclass categorization is done for minor categories. Advantage of binary classification is that it can broadly classify the contents but it does not address the subcategories in fine grained manner, e.g., binary classifier can categorize a page to a commercial or non-commercial category. On the other hand, multiclass classification can fine grain the contents into sub categories, e.g., news page can be categorized whether it is sports, politics or an entertainment page.

3.3 Classification Approaches

Apart from the above two broad categories, there are some other methods also available for web page classification in literature. These methods include -decision tree, Bayesian classifier, k-nearest neighbors, and SVM. Although, decision tree method is the most effective method for web page classification but it may generate false invalidation as the size of the data increases. It is easy to follow by the users using IF-ELSE constructs. Bayesian classifiers are also used in many web page classification techniques to find the conditional probability of related web pages so that these can be extracted to a higher level of accuracy. But, its complexity is higher as compared to the other classifiers of its category. k-nearest neighbor and SVM methods are also used for classification of web pages. SVM is the supervised machine learning method for web page classification in which machine is trained with respect to known pattern using a predefined learning rate. With variation in the learning rates, the desired level of accuracy can be achieved in SVM. K-nearest neighbors use the likelihood of page classification with respect to neighborhood of the data. Using which pages are classified in to different categories. Both these methods are the most popular techniques for web page classifications. It has been found in literature that if the training data size is small and restricted to only few classes, then SVM and k-nearest neighbor techniques are most effective in comparison to Bayesian network due to their higher complexity in extraction and searching. But, if the training data is uniform then all these techniques perform almost equally. But, as per the popularity is concerned then SVM-based classifier is one of the most popular techniques to be used for classification.

3.4 Features

Features play an important role in categorization. Generally classification of things is based on some specific and common features (Benbrahim et al. [45]). The biggest

challenge is to identify domain specific feature by closely analyzing the domains. During this process some of the features are highly relevant and others are not. Careful examination of features helps in this distinction. Because too many features for a problem in hand may add complexities. So reiteration of the list of features and selection of the most reliable one plays a crucial role in determining the overall performance of any designed solution in this environment. The web pages are further exploited to identify the feature sources by examining the following attributes as shown in figure 3.1.

- Multimedia tags and anchor text
- Graphical structure on the web through hyperlinks
- Images and hyperlink anchor text
- Meta tag description
- Meta tag keywords
- URL of the web page

By mixing all these information, a feature set is generated. In the existing proposals reported in the literature (Attardi *et al.* [47]; Benbrahim *et al.* [45]; Brin *et al.* [61]; Frnkranz [52]; Cakrabarti *et al.* [57],; Chen *et al.* [58]; Chang *et al.* [3]; Cohen [59]; Du *et al.* [65]; Hernndez *et al.* [62]; Hodgson *et al.* [51]; Lim *et al.* [46]; Murthy *et al.* [64]; Quek *et al.* [49]; Riboni *et al.* [48]; Sun *et al.* [38]; Uzun *et al.* [63]; Yang *et al.* [60]; Ghani *et al.* [50]), various researchers have used different features for classification by selecting the parameters such as-used link analysis, visual features, and meta tags. Others researchers have also used neighboring page information to classify the web contents. But, all these features are not sufficient to produce a high accuracy. Moreover, these techniques do not work well for heterogeneous web pages during different interval with different html tags and styles. The problem becomes more complicated with an introduction of new tags where web pages are

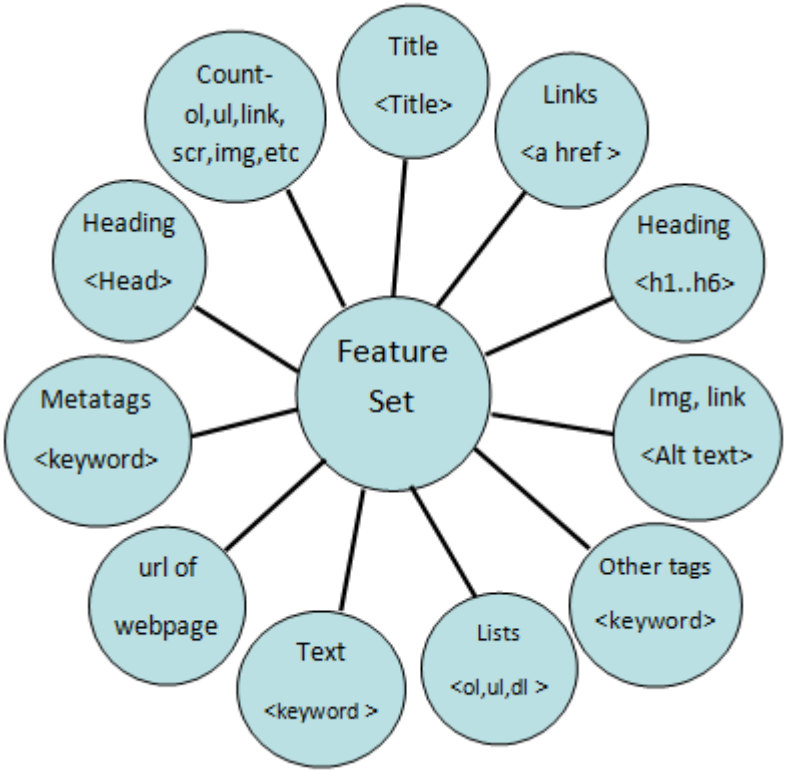


Figure 3.1: Features selected for web page classification.

developed in a professional manner. These new tags also flag relevant information for classification. All these factors lead us to develop a robust technique for classification to improve the precision and accuracy. The motivation and contributions of this chapter are discussed in the subsequent Sections.

3.5 Background about the Classification Tool (SVM)

Support Vector Machine (SVM) is a tool which is used for classification. SVM is used to design both linear and nonlinear classifiers for many applications. Based upon the needs of a particular application, different versions of kernels can be selected in SVM. Each kernel in SVM has a learning algorithm which acts as per the parameters selected for an application [19][20]. SVMs are machine learning-based systems in which system is trained using a learning algorithm which is used to classify a given class of data which is later on used for regression analysis. The support vectors in

SVM are computed by solving a quadratic programming (QP) problem. Following are the main features of SVM-based classifiers for any scientific applications.

(a) Even if small training data set, SVMs classifiers perform well to classify the given data sets in to different classes.

(b) Probability distribution of the data is not presumed on prior knowledge of dataset.

(c) Mathematical analysis of SVM is simple as compared to other machine learning methods.

Support Vector Machine (SVM) provides the power of flexibility from kernels. Kernel allows us to do stuff in infinite dimensions. Sometimes going to higher dimension is not just computationally expensive, but also impossible. f(x) can be a mapping from n dimension to infinite dimension. Then kernel gives us a wonderful shortcut. Kernel is to make the calculation process faster and easier, especially when the feature vector is of very high dimension.

SVM Kernels:

Mainly three different types of SVM-Kernels are available (Boser, Guyon, and Vapnik, 1992; Cortes, and Vapnik, 1995). SVM can result in high accuracy and with an appropriate kernel they can work well even if you're data isn't linearly separable.

Linear kernel:

$$f(x) = w^T x + b; \tag{3.1}$$

which finds the optimum separating hyperplane. where w is called the weight vector and b is known as the bias. where w is the weight, x is the feature vector, and b is

the bias. if f(x)¿ 0, then we classify datum to class 1, else to class 0. Need is to find a set of weight and bias such that the margin is maximized.

Linear kernel has some advantages but probably the most significant one is the fact that generally is way faster to train in comparison with non-linear kernels such as RBF.

Polynomial Kernel:

$$K(x, y) = (x^T y + c)^d; \tag{3.2}$$

where x and y are vectors in the input space, i.e. vectors of features computed from training or test samples and $c \geq 0$ is a free parameter trading off the influence of higher-order versus lower-order terms in the polynomial. When c = 0, the kernel is called homogeneous.

Radial Bias Kernel (RBF): The RBF kernel on two samples **x** and **y**, represented as feature vectors in some input space, is defined as

$$k(x, y) = exp(-\frac{||x - y||^2}{2\sigma^2}) \tag{3.3}$$

$||x - y||^2$ may be recognized as the squared Euclidean distance between the two feature vectors. σ is a free parameter.

The polynomial and RBF are especially useful when the data-points are not linearly separable. Although the RBF kernel is more popular in SVM classification than the polynomial kernel, the latter is quite popular in natural language processing (NLP). The most common degree is d = 2 (quadratic), since larger degrees tend to overfit on NLP problems. One problem with the polynomial kernel is that it may suffer from numerical instability. One more thing to add: linear SVM is less prone to overfitting than non-linear. Choice of kernel depends on the situation. If your

number of features is really large compared to the training sample, just use linear kernel; if your number of features is small, but the training sample is large, you may also need linear kernel but try to add more features; if your feature number is small, and the sample number is intermediate, use Polynomial kernel will be better.

More details about the SVM can be found in (Boser, Guyon, and Vapnik, 1992; Cortes, and Vapnik, 1995).

3.6 Proposed approach

As per the above discussion, there is a requirement of an efficient automatic web page categorization technique. The proposed scheme consists of various phases for web page categorization which are explained as follows.

3.6.1 HTML parsing

The first step in the proposed scheme is the HTML parsing. It consists of steps such as generating document object model (DOM), tag extraction, extract relevant data from the tags, and then generate data set file. These steps are shown in the figure 3.3.

3.6.2 Feature extraction

Web pages are developed using html scripting language. Our approach for feature extraction is based on DOM of web page. The relevant features extraction steps are shown in the figure 3.2. The HTML DOM is a programming interface for a HTML document and is defined as follow.

- The HTML elements are considered as objects in a web document
- The properties of various HTML elements need to be considered
- The procedures used to find various HTML elements need to be considered

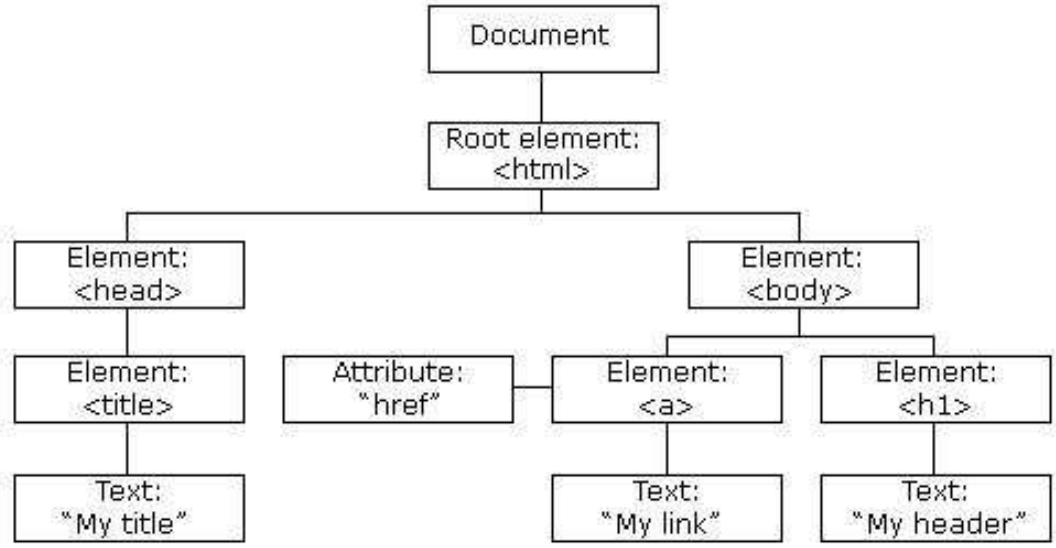


Figure 3.2: DOM for feature extraction (Eickhoff et al. [96])

- Various triggers and events for various HTML elements need to be considered

3.6.3 Dimension reduction

This step is performed to reduce the complexity of the proposed scheme. In this step, Zero weighted feature columns from the collected dataset are removed. Because either those tags as feature are not used in the domain pages or these tags are not contributing to information regarding domain. This process reduced the complexity of the problem.

The detailed block diagram of the proposed scheme is illustrated in figure 3.3

3.6.4 Algorithms

There are different algorithms designed for the web page classification in the proposed scheme. Algorithm 1 describes the steps performed for collection of keywords. In Algorithm 1, the domain specific web pages are selected using the popular keywords in step 1. In step 2, a keyword list is prepared. In step 3, inverse document frequency for each keyword selected in step 2 is made. In step 4, size of the elements is reduced based upon the inverse document frequency classification in step 3. Then

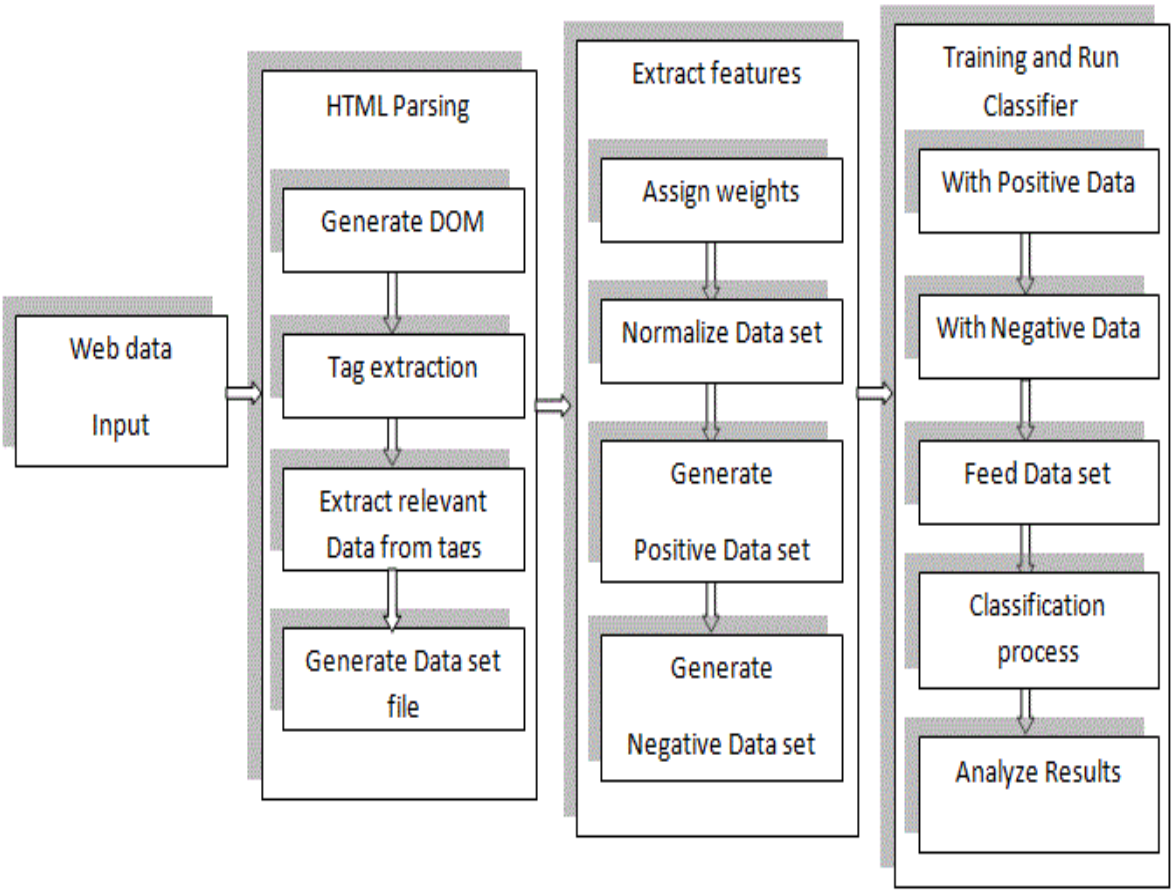


Figure 3.3: Block Diagram of Methodology

word stemming is performed for each keyword in the keywords list constructed in the earlier steps. In step 6, a final list of all the stemmed keywords is constructed. The detailed description of all these steps is given in Algorithm 1. The complexity of this algorithm is $O(n\_Webpages \times n\_kw)$.

Algorithm 1 Collect keywords

Input: Domain WebPages

Output: Stemmed Keywords List (KWL)

Step 1- Manually study the domain specific N web pages and selecting the popular keywords kwi describing the domain. (Refer Yahoo! Directory, Google Directory, Dmoz Open Directory)

Step 2-Prepare keyword list as $kwl = \{kw_1, kw_2, kw_3, ..., kw_n\}$.

Step 3-Calculating inverse document frequency(idf) for each kwi from N webpages.

$idf(kwi, D) = log\frac{N}{|\{d \in D : kwi \in d\}|}$

Step 4- Reduce the size of kwl elements on the basis of idf of kwi

Step 5- Applying word-stemming to each keyword in kwl.

Step 6- Make a final KWL of stemmed keywords, KWL = $\{KW_1, KW_2, KW_3, ..., KW_N\}$.

Algorithm 2 describes the detailed execution of the proposed web page categorization technique. The final stemmed keywords list which is an output of the Algorithm 1 becomes input to the Algorithm 2. An array of feature list is constructed in the next step of Algorithm 2. Then, all the features to be included are put in the URL and a domain specific URLs are opened. These URLs are then connected to DOM then elements to be searched are linked with initialization of number of elements count. Frequency of the elements to be included in the final list are identified using inverse document frequency. Finally, weight is assigned to the features to be included for web pages categorization. Keyword's counting with respect to the above weight assigned to the features become the criteria for categorization of the web pages.

Algorithm 2 Feature extraction and Weight Assignment Algorithm

Let F = html tag as feature
FL = F1,F2,F3 ... FN
$KW_L = \{KW_1, KW_2, KW_3 \ldots KW_m\}$ ▷ from algorithm1
M = $|KWL|$, N = $|FL|$
float FeatureWeight[N] ▷ Array corresponding to features in FL
Integer FeatureCount[N] ▷ keep the count of feature in url
Open.infile("WebAddresses.dat") ▷ Domain specific urls
while (eof()) **do**
    String url=infile.readline()
    doc = Jsoup.connect(url).get() ▷ doc is DOM (document object model)
    for $(i = 1, i \leq N; i++)$ **do**
        String S=Null
        FeatureWeight [i]=0
        Elements Eles = doc.select(F[i]) ▷ Elements links = doc.select("a[href]")
        Elementcount=0; ▷ Element count e.g no. of links/images/video
        for $(ElementE = 1; ElemenE < N; ElementE++)$ **do** ▷ (each ElementE in Eles)
            S= wordstem(E.text());
            Term-Frequency=0;
            for $(j = 1; j \leq M; j++)$ **do**
                ▷ $TF - idf = (\sum_{KW_1}^{KW_n} TF_i * idf(KW_i)$ in feature text
                FeatureWeight $[i]$ = FeatureWeight $[i]$+ count $(KWL[j], S)$
            end for ▷ end for KWL
            Elementcount++
        end for ▷ end for element
        FeatureCount [i]= Elementcount ▷ may be used as feature
        Append relevant data to dataset file in classifier format
    end for ▷ end for FL
end while
Close infile

3.7 Performance evalution

The performance of the proposed scheme was evaluated using simulation on SVM (Boser, Guyon, and Vapnik, 1992; Cortes, and Vapnik, 1995; Hsu, Chang, and Lin, 2003). Following steps are performed for evaluating the performance of the designed scheme.

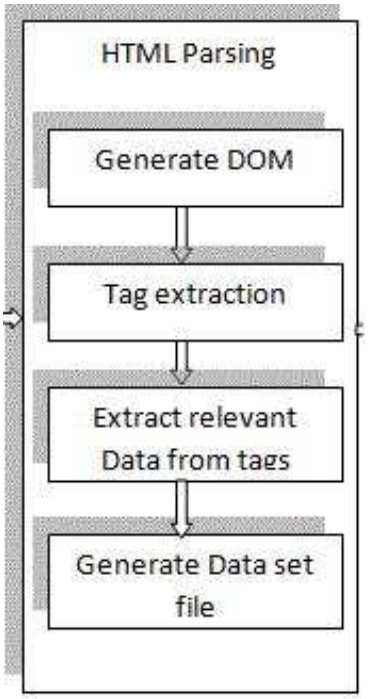
3.7.1 Experimental settings

A tool is developed using Jsoup java package in Java programming language. Using feature extraction tool, features are extracted from each page and stored in the excel file and then converted to support vector machine readable format as shown in figures 3.4, and 3.5. Also, figure 3.6 descibes the format readable by the SVM for classification of web pages.

There are various other settings performed in the proposed scheme such as data set classification, construction of training data, and finally testing data. Training data is used to train the SVM with respect to the given data set in the standard format which is readable to SVM. Learning rate is set for web page classification using grid.py python script built in libsvm tool. This script automatically adjust various parameter for learning and cross validation by recursively running the training dataset for particular kernel function till it obtains optimum result. SVM performance is monitored by two key parameters- C called as the penalty and d, a threshold set on the precision. An increase in the value of d results an increase in the precision but a decrease in the value of recall. On the other hand, a decrease in the value of d results a decrease in the precision but it increases the value of recall. Various settings done during the experimental validation are explained as follows.

```
Source  Design
522         String fav = "";
523
524          Element element = doc.head().select("link[href~=.*\\.(ico|png)]").first();
525          if(element==null){
526
527                  element = doc.head().select("meta[itemprop=image]").first();
528                  if(element!=null){
529
530  fav= fav + "\n "+element.attr("content");
531                  }
532          }
533          else{
534                  fav =fav+ "\n"+ element.attr("href");
535          }
536          jTextArea6.setText("fav"+fav);
537         fw.write(fav+"\n");
538
539
540         s=" ";
541         Elements hTags = doc.select("h1, h2, h3, h4, h5, h6");
542
543          Elements h1Tags = hTags.select("h1");
544          s=h1Tags+"\n";
545          Elements h2Tags = hTags.select("h2");
546          s=h2Tags+"\n";
547          Elements h3Tags = hTags.select("h3");
548          s=h3Tags+"\n";
549          Elements h4Tags=hTags.select("h4");
```

(a)



(b)

Figure 3.4: (a) Snippets of code (b) Steps in code

3.7.2 Dataset

WebKB dataset and 7sector dataset are used to cross check the performance of methodology and accuracy of SVM trained classifiers corresponding to each domain.

WebKB dataset is the collection of education domain web pages of four universities. Each university web pages are further divided into various categories such as-course, faculty, student, and project. 7sector dataset is already divided into web pages of seven sectors like banking, health, finance, insurance, technology, energy, and basic material. For each of the above category, randomly 1000 plus known categories especific web pages are examined. Separate dataset corresponding to each category is prepared by selecting domain specific web pages as true positive and non domain web pages as true negative.

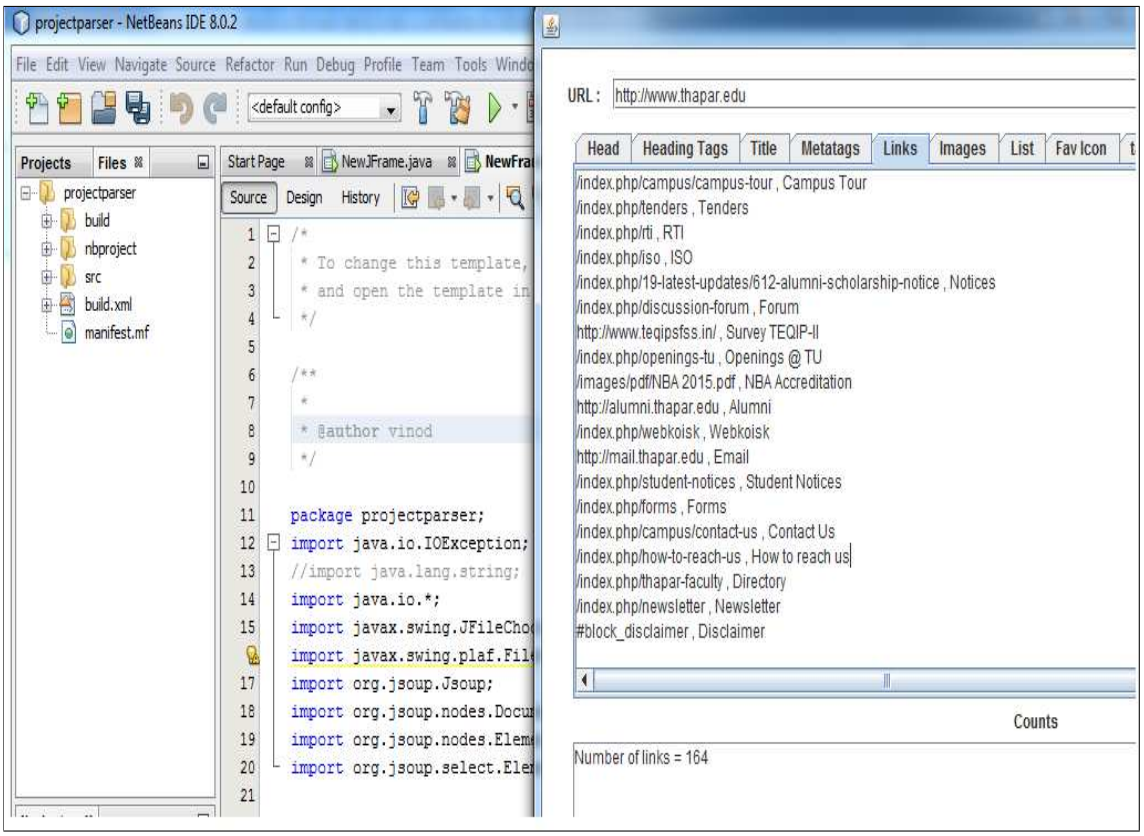
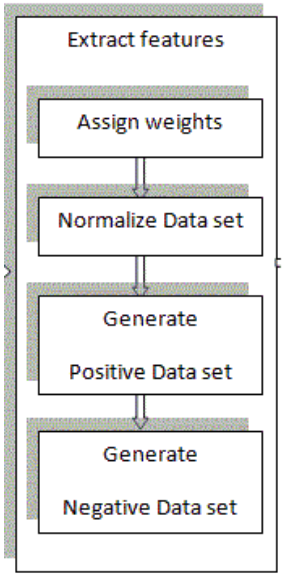


Figure 3.5: Feature extraction and weight assignment tool for sample site thapar.edu

These positive and negative pages are feed to feature extraction and weight assignment tool along with domain specific KWL generated using Algorithm 1. Further dataset is divided into training dataset, and test dataset. Category specific training feature dataset acts as a feed to SVM classifier for that category.

3.7.3 Dataset format

Training dataset and testing dataset are converted to SVM readable format (Chang, and, Lin [82]) as shown in figure 3.6(a). The dataset file format for SVM is also shown in the figure 3.6(b). Document label is represented by [label], specific to a particular class (ci) of a document. The Feature of a feature vector, and integer value are represented by [index] and [value] respectively. This format is used for both positive, and negative documents



(a)

[Label] [index1] : [value1] [index2] : [value2] . . .

[Label] [index1] : [value1] [index2] : [value2] . . .

(b)

Figure 3.6: (a) Dataset classification steps (b) SVM readable format dataset

.

3.7.4 Execution

The proposed scheme was based on the binary classification of the document. Hence, a linear kernel is chosen for the execution. It is relatively simple and has less overhead. Linear kernel function (Cortes, and Vapnik [20]) is defined as $\{g : f(g) = w^T g + b = 0\}$. where w is called the weight vector and b is known as the bias. Let us assume that $b = 0$. At $W^T g = 0$, all points of g are perpendicular to W passing to origin. The value of b shifts $f(g)$ away from the origin. The sign of the function $f(g)$ decides that on which side of the hyperplane a point lies (Hsu, Chang, and Lin [83]). Moreover, the maximum margin classifier maximizes the geometric mean margin $\frac{1}{||w||}$, i.e., $||w||^2$ resulted in the optimization problem as follows.

Min (w,b) $\frac{1}{2}||w||^2 + C\sum_{1}^{n} e_i$ with constraints such that $y_i(w^T x_i + b) \geq 1 - e_i; i = 1, , \ldots, n$; The slack variables ($e_i > 0$) allow the example to be in the margin ($0 < e_i < 1$). This formula is given by Vapnik [19,20] and is named as soft-margin SVM. $C > 0$ is computed for taking the maximum value of the margin with minimum slack. Hence, margin errors are penalized with $C\sum_{1}^{n} e_i$ term.

3.7.5 Scaling dataset

LIBSVM3.20 is used as a SVM tool to generate the training model classifier for a class (Chang *et al.* [82]). Training dataset and Test dataset feature vector are scaled in range [-1 , 1].

>>svm-scale –l -1 –u 1 –s range traindataset > traindataset.scale >>svm-scale –r range testdataset > testdataset.scale

Scaled Training data set is feed to the SVM for training purposes.

3.7.6 Cross-validation training and testing

>> {#svm-train -s0 -t0 -v5 [libsvm-options] traindataset.scale traindataset.model

To avoid over fitting and accuracy of test data results, in-built five-fold cross validation technique is applied to the training set (Chapelle, [84]). In this, the training dataset is divided in to five subsets. SVM is trained on four set data and fifth set is used for checking the accuracy. In next fold, subset acting as test data becomes part of training data and some other sub set act as test data. This process is repeated for five folds and corresponding to each fold accuracy is computed. Then, the average accuracy of model is predicted. Once the SVM is trained for a particular category, Test dataset for that category classified is feed to SVM for processing. Trained SVM classifer is applied to the scaled test data set for prediction as follows.

>>svm-predict testdataset.scale traindataset. model testdataset.predict

At the completion of execution, linear kernel filtered the data set based on binary categorization.

3.7.7 Parameters selected for evaluation

Final results are evaluated for the confidence building. F1 and P measures are standard metrics to qualitatively examine the results performance of experimental set up. Value of F1 should be in the range of 0 – 1. More the values of F near to 1, more accurate are the results [9,12,17,22] . Following definitions are used in experiments performed in the proposed scheme.

- Precision(Pr): It is computed by the ratio of number of correct categories to the total number of categories and is defined as follows.
  Precision = $\frac{TruePositive}{TruePositive+FalsePositive}$
- Recall(Re): It is computed by the ratio of correct categories assigned to the

total number of known correct categories and is defined as follows.

$Recall = \frac{TruePositive}{TruePositive+FalseNegative}$

- Accuracy $= \frac{TP+TN}{P+N}$
- F1 measure: It is computed by taking the harmonic mean of precision and recall and is defined as follows.

  $F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall}, \qquad (0 <= F1 <= 1)$

Precision and recall are calculated after running the trained classifier on test dataset using optimum parameters obtained through grid.py python script. These parameters put the limit on the threshold values. Moreover, the initial keyword selection list plays important role during the experimental validation. Hence, a large number of keywords in keyword-list (KWL) may lead to over fitting and short list may lead to misclassification.

3.8 Results and discussion

To justify the choice of liner kernel function $x : f(x) = w^T x + b = 0$, Cross validation-Grid parameter selection is performed on scaled training dataset of each category by linear, polynomial, and radial basis kernel. The parameters selected above are evaluated to check out the effectiveness of the proposed scheme as follows.

$\# >> grid.py - log2c - 5, 15, 2 - log2g3, -15, -2 - v5 - s0[libsvmoptions]$
$traindataset.scale.\#$

Figure 3.7 shows the results obtained with respect to RBF kernel-based cross validation grid parameter selected for the course category. As shown in the figure, the accuracy obtained using the proposed RBF kernel-cross validation is more than 99%. This proves the effectiveness of the proposed scheme with respect to cross validation.

Figure3.8 shows the accuracy of the proposed scheme for cross validation with respect to polynomial cross validation. As evident from the results obtained, the polynomial cross validation is better than the RBF kernel-based cross validation.

Figure 3.9 shows the accuracy of the proposed scheme with respect to linear kernel cross validation for the category courses. As evident from the results obtained, among all three types of kernel, linear kernel has the highest accuracy among all three types of categories. Reverse operating curve (ROC) as shown in figure 3.10(a) for WebKB dataset- course category and figure 3.10(b) cross validation ROC for training dataset WebKB- student category also indicate the performance of proposed solution. This proves the effectiveness of the proposed scheme for the category courses. Also, the results obtained indicate that linear kernel-based cross validation has higher accuracy cost and gamma which is a clear indication that proposed scheme is successful in achieving higher level of accuracy which increases the reliability of proposed web page categorization scheme.

Table 3.1 shows the results obtained from the WebKB dataset. The dataset contains three categories namely as- course, student, and faculty. Test, and training dataset parameters are selected. Based upon these three categories, the values of the parameters Precision (Pr), Recall(Re), and F1 measure are computed. Accuracy (Acc), cost (C), and gamma functions are computed for training dataset also. Convincing results are obtained using the proposed scheme as shown in the Table 3.1.

Table 3.2 shows the results obtained for 7sector Dataset for Banking, Health, and Technology. Again the data is divided in to test, and training datasets. The performance of the dataset is evaluated with respect to the parameters such as Pr, Re, and F1 measures. Acc, C, and gamma function are evaluated with respect to the training dataset in the proposed scheme. The results obtained in the Table 3.2 indicates that the proposed scheme achieves a higher level of accuracy with respect

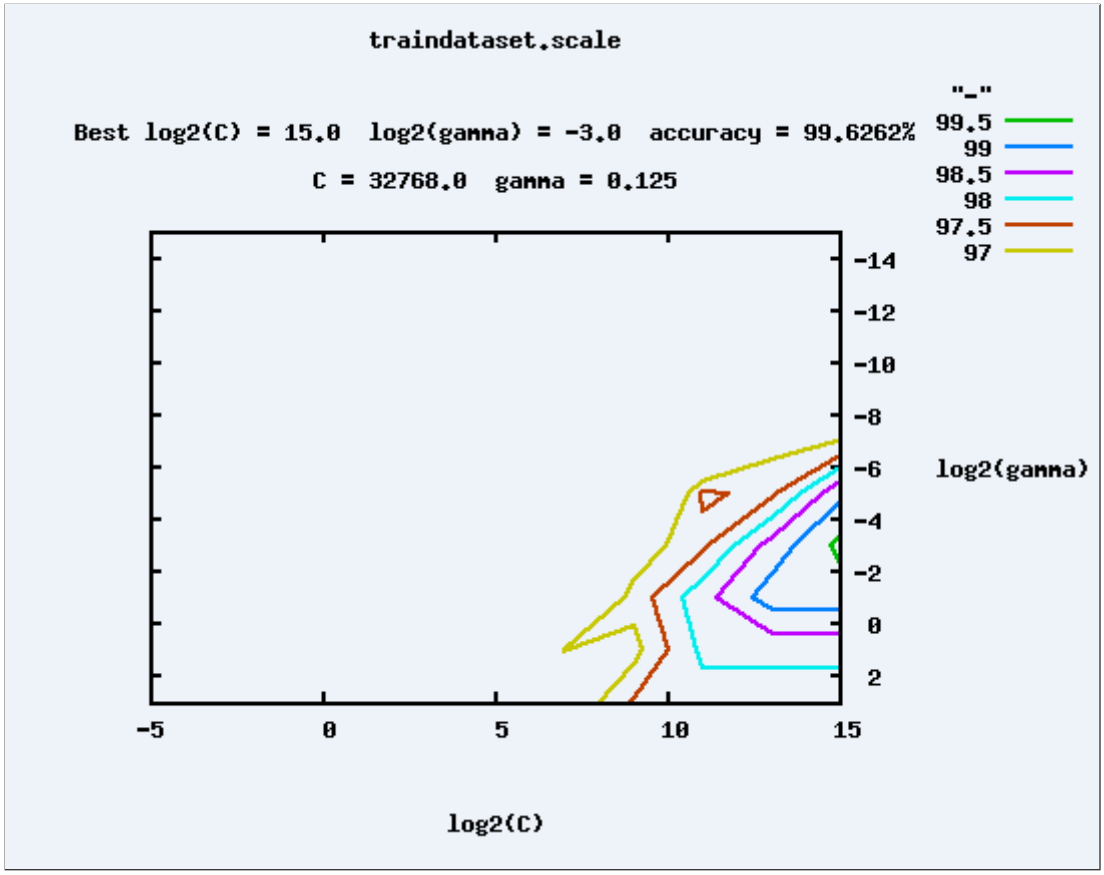


Figure 3.7: RBF kernel-cross validation-Grid parameter selection for course category

to these parameters. Hence, the proposed scheme can be used in web page classification for large number of categories of web pages. Tables 3.1, and 3.2 clearly indicate the superior performance of the proposed scheme with respect to the selected parameters which is supported by the results shown in the figures above.

Precision and recall are calculated after running the trained classifier on test dataset using optimum parameters obtained through grid.py python script. These parameters put the limit on the threshold values. Moreover, the initial keyword selection list plays important role in this. Large number of keywords in keyword-list (KWL) may lead to over fitting and short list may lead to misclassification. Hence learning rate is set for web page classification using built in grid tool of libsvm tool. This script automatically adjust various parameter for learning and cross validation by recursively running the training dataset for particular kernel function till

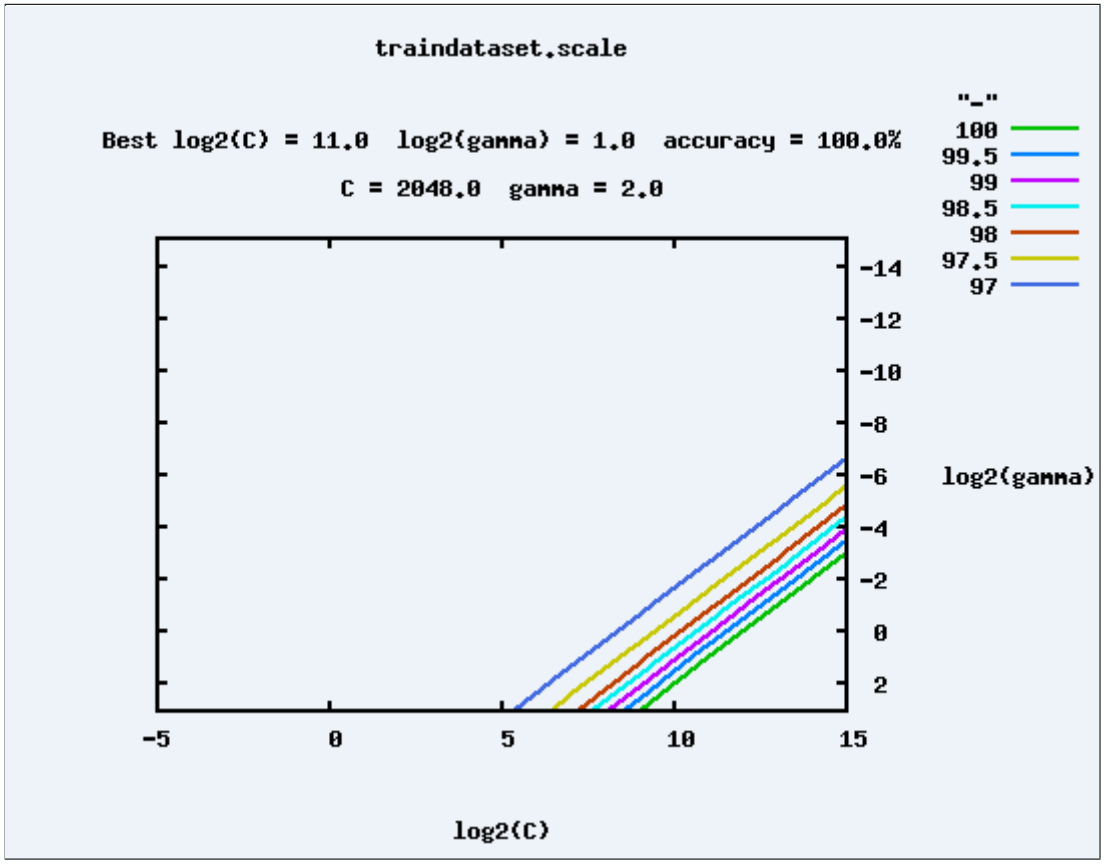


Figure 3.8: Polynomial kernel-cross validation-Grid parameter selection for course category

it obtains optimum result. SVM performance is governed by two parameters, C (the penalty imposed on training examples that fall on the wrong side of decision boundary) and d (the decision threshold). Increasing d, result in fewer test items meeting the criterion and this usually increases precision but decreases recall. Conversely, decreasing d typically decreases precision but increases recall. d is chosen to optimize performance on F1 measure on training validation set.

Table 3.1: Results of WebKB data set formula $\{x : f(x) = wTx + b = 0\}$

WebKB DataSet (Chen, Lee, and Chang, 2009)	Test dataset			Training Dataset Parameters		
	Pr	Re	F1	CV-Acc	Cost (C)	gamma (γ)
Course	.97	.82	.88,	100	8192	.0078
Student	.93	.86	.89	99.65	8248	.0082
faculty	.87	.79	.82	98.57	2048	.0078

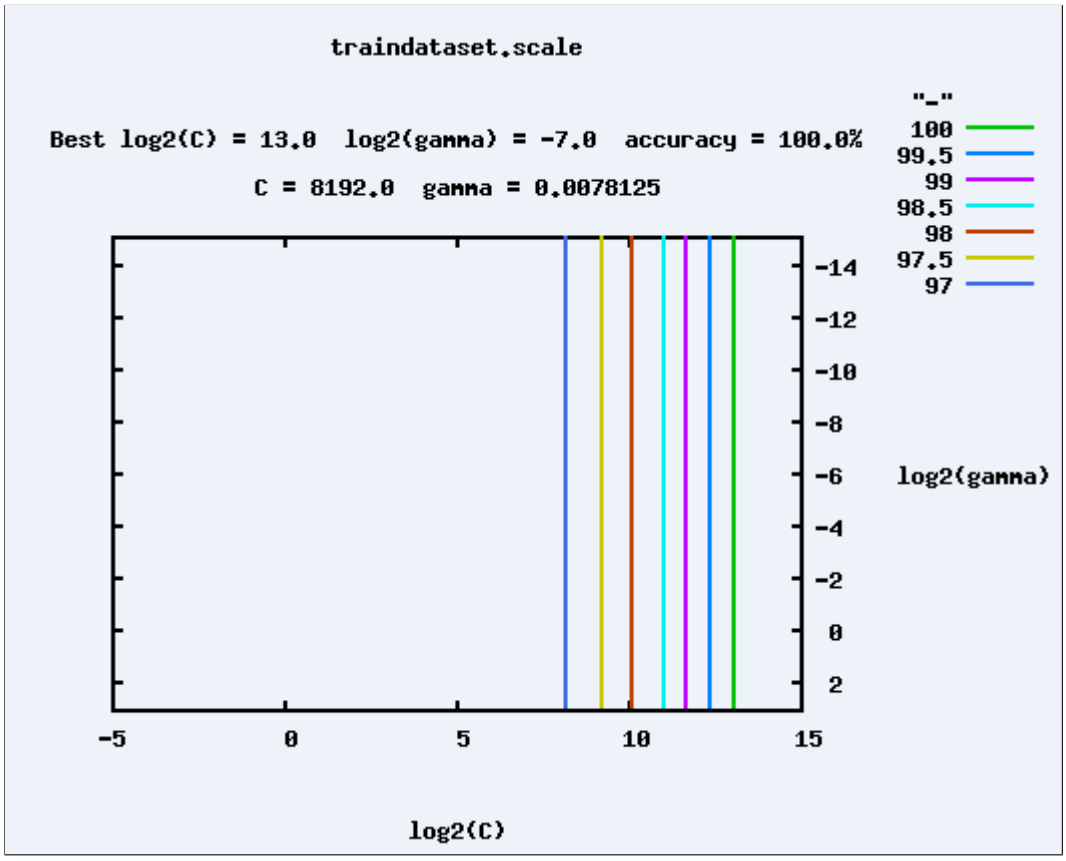
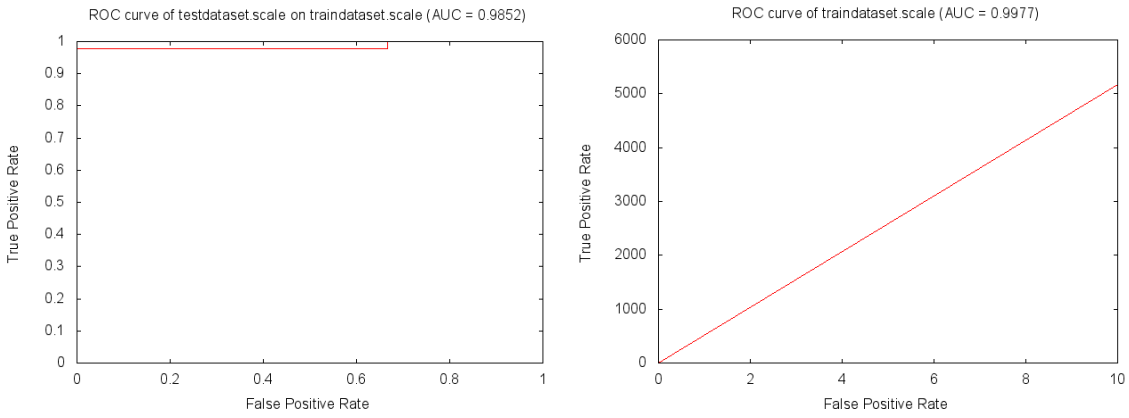


Figure 3.9: Linear kernel -Cross validation-Grid parameter selection for course category

Table 3.2: Results of 7sector data set formula $\{x : f(x) = wTx + b = 0\}$

7sector DataSet (Burget, and Rudolfova, 2009)	Test dataset			Training Dataset Parameters		
	Pr	Re	F1	CV-Acc	Cost (C)	gamma (γ)
Banking	.87	.79	.82	96.31	9280	.0048
Health	.88	.76	.81	98.18	2048	.0084
Technology	.81	.86	.81	94.29	32	.0078



(a) (b)

Figure 3.10: (a) Test Data ROC Curve for WebKB Course Category (b) Cross validation ROC curve-training dataset WebKB Student Category

.

3.9 Summary

Web page categorization is one of the most challenging tasks to be achieved due to the rapid increase in the number of users who demand a quick access of web pages with respect to their expectations. In this chapter, we proposed a novel SVM-based web pages categorization. Experimental evaluation showed that tag-based classifiers outperformed the other existing classifier such as-Bayesian, and k-NN. It is evident from the results that accuracy of the model depends on two major factors- (i) Accuracy of feature selection-extraction technique. (ii) Initial selection of domain specific keyword list ($kL = \{kw_1, kw_2, \ldots kw_n\}$) which is used for weight assignments to the features. Various evaluation metrics such as-precision, accuracy, and F call are chosen to test the effectiveness of the proposed scheme. With respect to these metrics, the proposed scheme outperforms the other state-of-the art schemes in literature.

In future, this work can be extended to include N-grams. Results could be improved by adding or dropping number of feature for relatively new web pages developed using HTML latest version and problem can further be extended to multi-class classification of web pages. Other kernels of SVM can be explored for automatic multi-class classification for the improvement of F, and P values.

Chapter 4

Multi-class Web page classification

4.1 Introduction

From the past few years, we have witnessed a tremendous growth in wireless communication technology. During this era, there is a gradual increase in the web applications users using Internet technology. This revolution has lead to the design and development of different types of applications which are mainly used for the benefits of common peoples. A large number of portable devices have been developed during this era using which Internet can be accessed by the end users from anywhere. Users can access various resources from anywhere even on-the-fly which makes resources sharing and data communication, an easy task. This is possible only with the advancements in wireless and cloud computing technologies which has changed the whole world. With the advancements in communication infrastructure, a large number of smart devices such as Smart watches, Internet-enabled coffee makers, smart city, Smart Homes, Long distance health monitoring, inventory control, automotive industry etc. are developed from different organizations. These smart devices are interconnected with other smart devices located across different geographical locations. This type of environment is called as the Internet of Things (IoT), where a large number of devices share information with one another. These devices share information with one another using various short, medium and

long range transmissions technologies and protocols such as ANT, NFC, ZigBee, Bluetooth, Wi-Fi, LoRa etc. Large number of devices coming on internet posses challenge for quality of service (QoS). Web classification can play important role by providing relevant data at fast rate resulting in reducing the communication bottle neck. Web document classification help in searching and sorting the data at faster rate suited for IoT and hence reduce the traffic on internet. Users access various services from the smart devices using the underlying communication infrastructure and protocols in IoT environment. The web interface acts as an intermediate data transmission medium between the users and database repository from where the data can be accessed as per the user's choice. Usually, users demand that only relevant information should be retrieved by the users so that congestion on the network can be minimized due to unnecessary data transfer between source and destination.

Hence, novel approaches and techniques need to be devised to reduce the manual efforts in web page classification. Keeping focus on these points, this chapter proposes a novel approach for multiclass classifier based on unique personality features of the web page of particular domain category for the next generation wireless networks. Personality features are collected and assigned weights in the proposed scheme. Then, the proposed classifier is trained based on these special features. Results obtained depict that proposed classifier successfully classified news domain pages, education, resume, online shopping, and research web pages from large database repository.

4.1.1 Motivation

Over the years, wireless communication networks have been widely used by a large community of users in wide variety of applications such as intelligent transportation systems, energy management, safety, and security etc. But, during this era, due to large number of user's request, there may be a performance bottleneck in some

part of the network with respect to various QoS parameters such as congestion and network delay. Web document classification helps in searching, sorting, retrieval, and querying of a document for the wireless networks. World Wide Web (WWW) contains huge repository of information in the form of web pages. However, size of Internet is growing day-by-day. The huge repository of information poses challenge to collect and process the relevant related information of a particular domain. So, traditional text classification techniques are difficult to apply on the rapidly growing web-based contents.

In light of the above facts, there is a requirement of an efficient classification technique to reduce congestion in the network so that throughput of various applications can be increased.

4.1.2 Contribution

Based upon the above research issues and challenges, major contributions of this chapter are summarized as follows.

1. A novel multi-class classifier using on-page positive personality features is proposed for web page classifications.

2. An algorithm for feature selection, extraction and weight assignment is designed, the performance of which is found satisfactory with respect to various parameters in comparison to the other state-of-the art existing techniques.

3. Training and testing of the multiclass classifier is done using five-fold cross validation and it is applied to wide variety of heterogeneous data better results on the tested data samples.

4. Accuracy of the proposed classifier is found to be satisfactory from a large data set of different categories. Also, there is a 10–15 % overall performance gain using the proposed scheme in comparison to the other existing schemes.

4.2 Advantages of the proposed scheme

In summary, following are the advantages of the proposed scheme.

1. It has a multiclass classification.
2. Less computational cost in terms of feature extraction and weight assignment in the proposed scheme.
3. There is no need of negative training dataset.
4. Small feature set with fast training and testing of the data set with higher accuracy.

4.3 Proposed solution

The technique described in this chapter relies on the personality of the web page. Unique personality features distinguishes one web page from others. Proposed approach suggests personality dependent features derived by studying and analyzing a particular domain of web pages. This chapter is mainly focused on five domains- Resume pages, Educational Web pages Research pages, E-newspaper web pages and online shopping web pages. The commonality among web pages of particular domain is identified based on personality of web pages of that domain which is used to extract and generate positive feature set. Then weights are assigned to these feature sets and the classifier is trained with the relevant training data. Once the classifier is trained with the positive feature set, actual data is feed to the classifier for the desired domain specific results. Results are then manually cross check for Precision (P) and Recall (R) values as suggested by Hsu *et al.* [40]. Proposed technique produced the prominent results. The difference is evident from Fig. 4.1 b, c which shows a newspaper web page represents news domain and research web page represents research domain respectively. The number of links, images, words,

and videos differs clearly in these figures. Following constructs are used for feature selection in the proposed scheme.

4.3.1 On Page information sources

Web pages are semi structured and developed in tag based html scripting language. Tags on the web page flag semantic content [25]. This information can be exploited in search for feature selection from different prospective.

Text: Amount of words in form of text is a good indicator to distinguish web pages. Typically Resume page/personal home page has less text in comparison to newspaper or educational page.

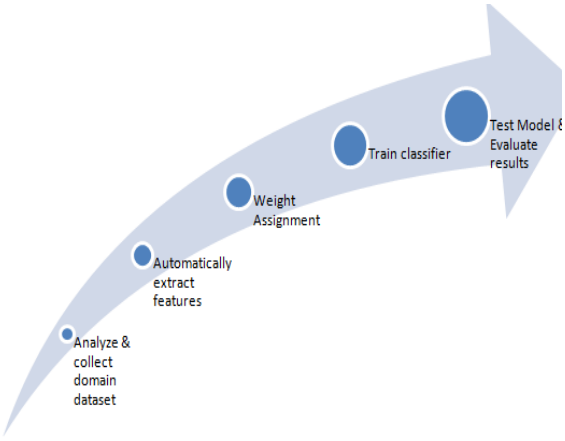
Links: Placement of links on a page can contribute towards the category of page. Generally newspaper and education pages are rich in terms of links on page.

Images: Number of images on page is also good indicator of category. News page has more images, whereas resume page has one or no image.

Synthetic images: Synthetic image can be distinguished by the histogram of image. Usually synthetic images are placed on research page in the form of graphs, equation or to represent a formula.

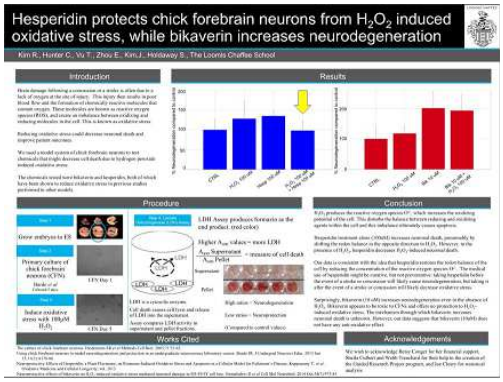
HTML tags: Presence of html tags like <Table>, Lists <dl>,
, <hr>, <p>, <b>, <i>,<u> can contribute in distinguishing the class of page.

Multimedia: Use of multimedia content in the form of video/audio can act as an indicator in deciding the class.



(a) Summary of proposed approach

(b) Sample Newspaper Page



(c) Sample of Research page

Figure 4.1: Scenario (a) Steps in proposed scheme (b) Sample newspaper page (c) Sample research page

4.4 Domain Personality feature set

4.4.1 Resume page/Personal Home Page

Resume page has relatively less textual contents as compare to research page and newspaper page. It may have one or no image on it. Resume page may have two to three tables describing the qualification, experience. List tag <ol>, or <dl> are used to specify skill sets. Resume page may use bold<b>,<u> and <i> tags to highlight the achievements,
,<hr> tags are used to separate out the blocks of resume. It may use three four links <a href> as other sources of information

regarding person in the form research publications or references.

Feature Set F1= count { words, <ol>, <ul>, <dl>, <img>, <b> ,<i>,<u> ,
,<hr>, <ahref>, <table> }

4.4.2 Research pages

Research pages are rich in textual content as compare to other categories. Research page may contain few natural images in the form of pictures and few synthetic images in the form of graphs/equations. Synthetic images can be recognized by studying histogram of images on grey scale. It may have one/two videos describing the research. Number of paragraphs <p> are usually high. Generally usage of <table> tag is high.

Feature Set F2 = count {words, <img > natural, <img> synthetic , <p>,<table>}.

4.4.3 E-newspaper web pages

Newspaper web pages have large number of images, videos, audios. It also has large number of links. Text present on the page is also high in terms of words. Use of list<ol>/<ul>/<dl> is also high. Heading <h1..h6> are used quit often.

Feature Set F3= count { words,<img>,<ahref>,<video>, <audio>,<p>, <h1.. h6> }

4.4.4 Education Web pages

Newspaper web pages have large number of images, videos, audios. It also has large number of links. Text present on the page is also high in terms of words. Use of

list / / <dl> is also high. Heading <h1..h6> are used quit often.

Feature Set F4= count { words,<img>,<ahref>,<video>, <audio>, <p>,<ol>, <ul>, <dl> }.

4.4.5 Online Shopping web pages

Shopping page has large number of images that describe the products. Typically it has less text, moderate number of links, video. Use of <h1..h6> and list tags are high.

Feature Set F5= count { words,<img>, <ahref>, <video>, <ol>, <ul>, <dl> , <h1..h6> }

4.5 Feature extraction and weight assignment

Corresponding to feature vector F, a tool is developed using NetBeans IDE in java language. Tool is named as feature extraction tool (FET) which parses the web pages in the form of document object model(DOM). Feature vector elements are extracted and counted by querying the DOM. Count of elements are assigned as weights to feature elements. Zero weight is assigned to missing features. Further weights are scaled in the range [-1 1]. Feature Tool has inbuilt facility to convert data to LIBSVM [3] format as

[label index1:value1 index2:value2 ... indexn:valuen].

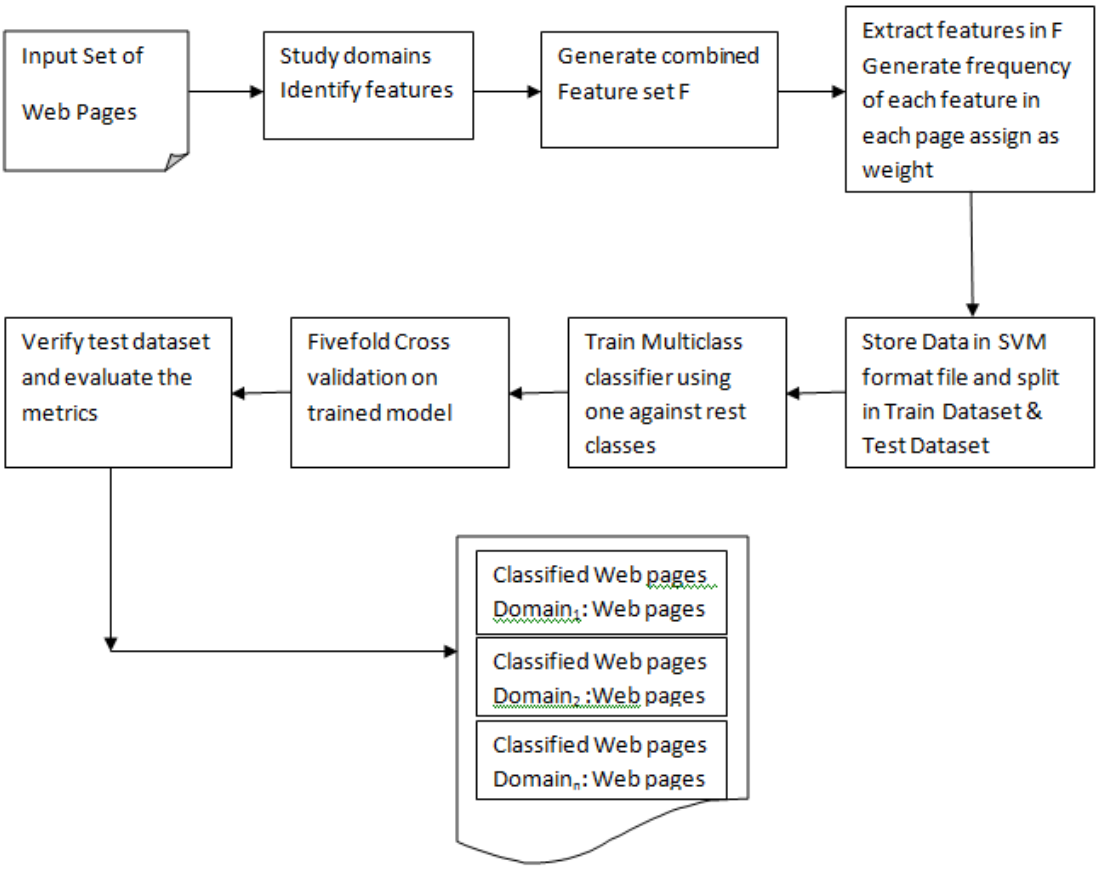


Figure 4.2: Flowchart of the proposed scheme

4.6 Data set

Dmoz open directory project (ODP), Yahoo! Directory, various university sites, E-newspaper sites online shopping sites were explored for generating database of URLs. Approximately, 200 URLs corresponding to each domain are collected. Feature extraction tool traversed URLs database to extracted the features, assigned the weights and appended the data to DataSet file. There is no need of negative training data set because the proposed approach classifier works on the principle of one against rest. DataSet was divided into training dataset and test dataset. Another small test dataset2 of domain web pages was collected randomly from WWW to cross check the accuracy of multiclass classifier (Figs. 4.2, 4.3).

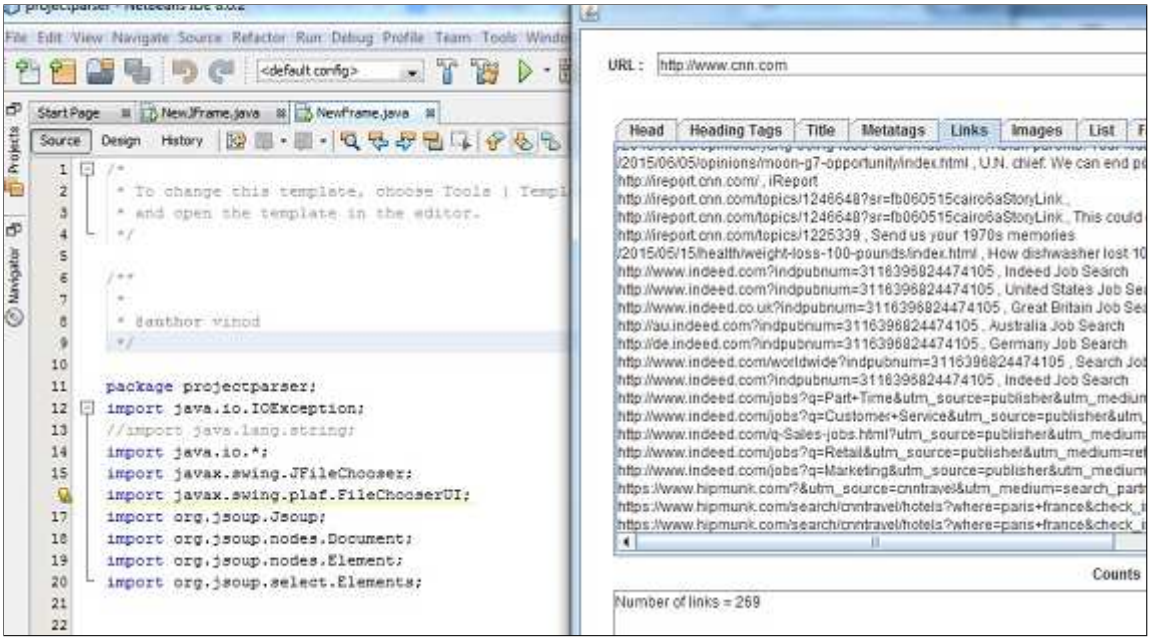


Figure 4.3: Feature extract and weight assignment tool (FET)

4.7 Algorithms

Two algorithms are designed to test the effectiveness of the proposed scheme. Both algorithms are explained in details as follows. Algorithm 3 selects the features of web pages in data file. The algorithm collects the web pages as an input. After collecting the web pages, D is selected from domain (line 1) and feature F_i is selected from feature set (line 2). For all the features, the feature that has most frequently used tag in domain is identified (lines 3-5). From all the features, the web page is selected to calculate the frequency of the feature appearing on the web page. From all the features of combined Feature set, the web page is selected to calculate the frequency of feature appearing on the web page.

Algorithm 3 Feature selection, extract and weight assignment

Input: Collection of web pages
Output: Features of web pages in data file

1: Let $D \leftarrow \{D_1, D_2 \ldots D_N\} \ \forall \ D_i \in$ Domain
2: Let $F_i \leftarrow \{f_1, f_2 \ldots f_n\} \ \forall \ F_i \in$ Feature set ▷ where f_i represent feature
3: **for** $\forall \ i \ \in$ Domain **do**
4: Identify and collect F_i as most frequently used tags in D_i web pages in F_i
5: **end for**
6: $F = \{F_1 \cup F_2 \cup \ \ldots F_n\}$
7: **for** i=1 ... N **do** ▷ $N \leftarrow$ number of domains
8: **for** j=1 ... n **do** ▷ $n \leftarrow$ number of domain web pages
9: Feed $W_j \ \in \ D_i$ to FET; ▷ FET←Feature Extraction Tool
10: $Weightf_i = 0 \ \forall \ f_i \in F$ ▷ W_j represents web page
11: **end for**
12: **for** k=1 ... N **do** ▷ $N \leftarrow$ Total number of domain web pages
13: **if** $f_i(W_k) \in F$ **then**
14: $Weightf_i \leftarrow Weightf_i + 1$ ▷ $Weightf_i$ represents frequency of feature that appeared on web page
15: **end if**
16: Add $Weightf_i$ to datafile ▷ $\forall \ f_i \in F; \ \forall \ W_j \in D$
17: $Weightf_i = 0; \ \forall f_i \in F$
18: **end for**
19: **end for**
20: Select next domain and goto step 7
21: Separate Train dataset and Test dataset in SVM readable format

The selection of web pages is done using feature extraction tool which lies among the domain (lines 7-12). For all domain pages if the feature of web page exists in feature set, then the frequency of feature is incremented by unity. The weight of data file is updated and frequency of each feature in the feature set F appearing on web page is set to zero before the loop execution is repeated. The for loop is repeated for all the domain pages (lines 13-19). The next domain is selected and steps number 7 to 19 are repeated. Training and testing of the data set are separated in SVM format (line 21).

The overview of training and testing of multi-class classifier is provided in algorithm 4. Initially the multi-label set is represented in the form of X_iY_i and denoted as T, where the domain of i is from 1 to n. The input space X, is assigned S^T and y are possible labels that varies from 1 to C (lines 1-2). A variable y_i which is a

subset of Y is associated with x_i which belongs to X (line 3).

Algorithm 4 Training and Testing the multiclass classifier

Input: Multi label training set is represented as $T = \{(Xi\ Yi)|1 \leq i \leq n\}$
Output: To predict a set of labels for each unseen instance

1: Let input space $X \leftarrow\ S^T$
2: Let $y = \{1, 2, 3\ ...C\}$ ▷ Possible labels
3: $x_i\ \in\ X$ is a single instance and $y_i \subseteq Y$ is the label set associated with x_i
4: Construct $B \leftarrow \{B_1, B_2\ ...B_n\}$ $B_i \in$ Binary classifier
5: Train each $B_i\ \forall\ i$ to separate class from rest
6: Combine B_i to get a multi-class classification according to the maximal output
7: **for** j=1 ... C **do**
8: Applying the sgn function argmax g^j where $g^j(x) = \sum_{i=1}^{c} y_i \alpha_i^j k(x, x_i) + b^j$
9: **end for**
10: Allot the point x to the class whose confidence value is largest for this point
11: Apply five-fold Cross validation
12: Predict classes of test dataset

A binary classifier B is constructed which takes values from $B_1, B_2, ...B_n$ (line 4). All Binary classifiers are trained to separate class from the rest. For each binary classifier, a combination is applied to get multi-class classification according to maximal output (line 6). After this sgn function and argmax function are applied for g^j where, g^j is defined as follows.

$$g^j(x) = \sum_{i=1}^{c} y_i \alpha_i^j k(x, x_i) + b^j \tag{4.1}$$

The above equation is applied for all j varying from 1 to C (lines 7-9). After that a point x is alloted to the class whose confidence value is largest for that point. A five fold cross validation is performed to predict the class of test data set (lines 11-12). Finally, classes of test data set are predicted and their values are returned (line 13).

4.8 Complexity analysis

Complexity of algorithm1 is O(n_Webpages × n_features) as it is hard to characterize the complexity of algorithm2 correctly. First, there are two complexities involved: at training time and at test time. For linear SVMs, at training time, we estimate the

vector w and bias b by solving a quadratic problem. It's O(max(n,d) × min (n,d)), where n is the number of points and d is the number of dimensions. For multiclass classifier it becomes O(max(n,d) × min (n,d) ×C), where C is number of classes

4.9 Performance evaluation

4.9.1 Simulation settings

LIBSVM Support Vector Machine [3] is used as a classifier for evaluation of the proposed scheme. It supports various kernel functions for classification and regression. Linear kernel was selected to construct $B \leftarrow$ (B1, B2 ... Bn), Bi is the binary classifier for experimentation. Since multiclass problem at hand is solved by converting it to binary classification problem and so it is solved it one class against the rest of classes.

Linear kernel function is defined as $\{x : f(x) = w^T x + b = 0\}$, parameter b : bias translates the hyperplane away from the origin. The constrained optimization problem is similar to as explained in [1,2,5].

$$Min(w, b) = 1/2||W||^2 + C \sum_{1}^{n} e_i \tag{4.2}$$

subject to the following

$$y_i(w^T x_i + b) > 1 - e_i, i = 1, 2, ..., n; \tag{4.3}$$

$e_i > 0$; $C \sum_{i}^{n} e_i$ is a penalty term for misclassification and margin errors. Where $e_i > 0$ are the slack variables used for the margin of error.

4.9.2 Converting Multiclass to Binary class

The proposed approach is based on multi-label classification which works on the principle of one against all. Binary classification problem is built corresponding to

each label. The instances associated with that label are in one class and the rest are in other class Python script in LIBSVM software, developed by[3] trans_class.py is used for converting multiclass problem to binary class problem.

>>python trans_class.py TrainingDataSet TestDataSet.

Program generates three files tmp_train, tmp_test, tmp_class respectively training dataset , testdataset and class labels for mapping.

4.9.3 Cross validation, Training and Testing

Training of classifier was done using standard five-fold cross validation technique. During this process various options and parameter values were tried to get the maximum cross validation accuracy. After cross validation classifier model tmp_train.model was generated using training data set as tmp_train. Classifier model works as one class verses rest classes as mentioned above.

>>svm-train –v 5 – t 0 [other libsvm options] tmp_train tmp_train.model

Next step was to test the model by predicting classes of test dataset as tmp_test and measure the accuracy of model.

>>svm-predict tmp_test tmp_train.model tmp_predict.

4.9.4 Evaluation metrics

Precision(P), Sensitivity(Sens) also known as recall (R), Accuracy(Acc), F Score(F1) metrics were used to measure the performance of proposed scheme[5,28]. These parameters are defined as follows.

$$P = \frac{TP}{TP+FP},\ Sensitivity = \frac{TP}{TP+FN}$$

$$Accuracy = \frac{TP+TN}{P+N},\ F1 = \frac{2P \times R}{2TP+FP+FN}$$

TP= Number of true positives, TN= Number of true negatives

FP= Number of false positives, FN= Number of false negatives

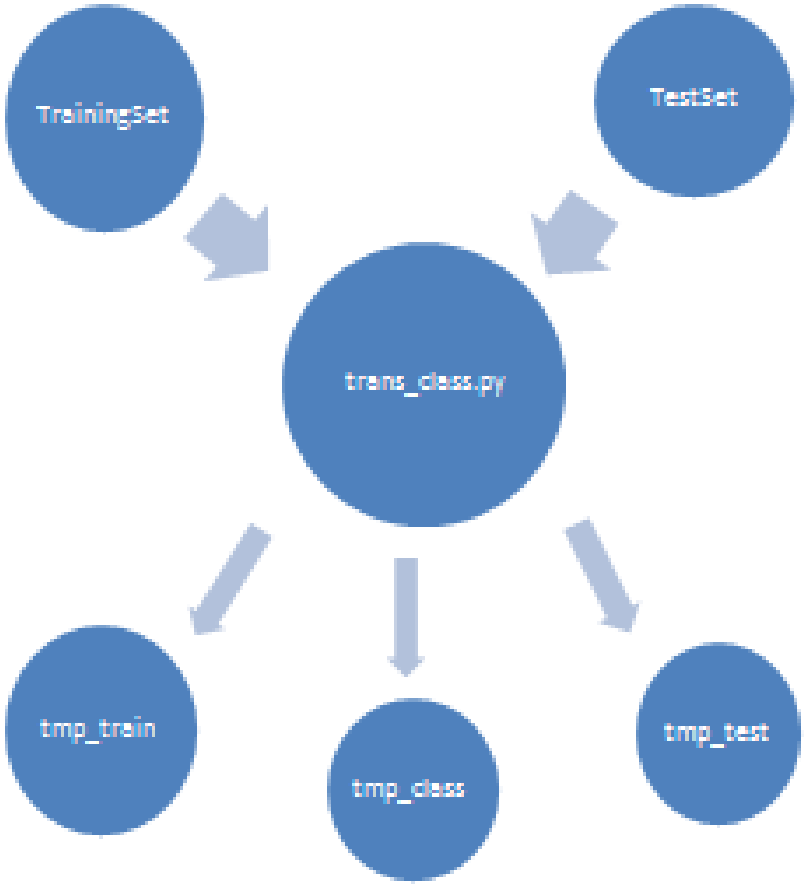


Figure 4.4: Working of python script

4.10 Results & Discussion

Results are shown in table 4.1 and table 4.2. Proposals in the past have used various other metrics on page and neighboring page features like meta-tags, text, link analysis, URL analysis, <title>, <head> , other html tags or combination of these features for classification using different classifiers as Bayesian probabilistic models, SVM, K-means, Nearest neighbor, genetic algorithm, or ant colony classifier to classify web pages.

In comparison to those existing proposals, we have developed a reliable, low cost classifier for an efficient web page classification. Generation of dataset and weight assignment are computationally less expensive and fast in the proposed scheme.

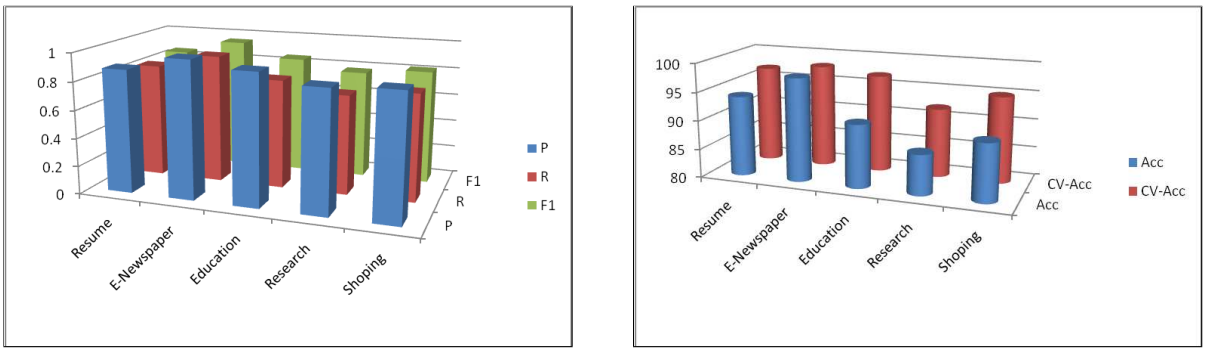
Moreover, proposed approach is independent of negative training samples and has higher precision than the existing schemes. This proves that proposed technique is reliable and has higher accuracy with less computation cost and time.

Table 4.1: Result of Test DataSet1 (Generated by dividing dataset to training and testing)

Domain class	Test Dataset1				Accuracy
	P	Sens	Accuracy	F1	
Resume	0.88	0.82	94	0.848	97
E-Newspaper	0.98	0.92	98	0.949	98
Education	0.93	0.78	91	0.848	97
Research	0.86	0.71	87	0.777	92
Shopping	0.88	0.76	90	0.815	95

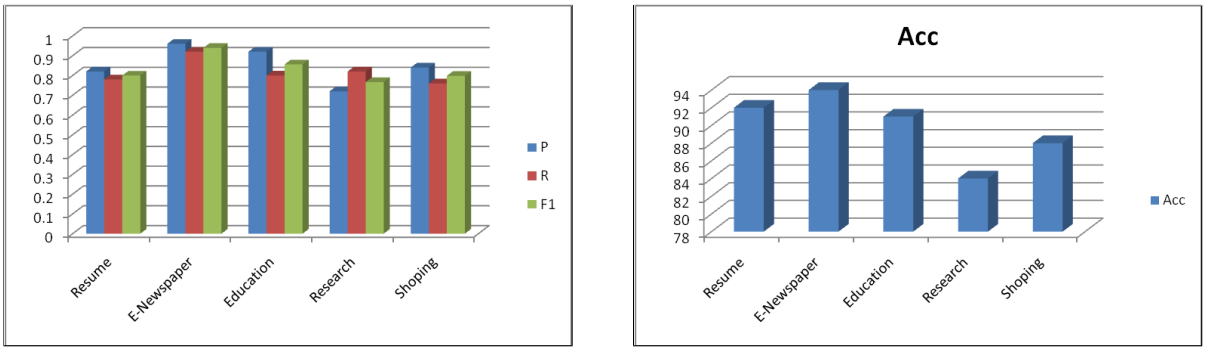
Table 4.2: Result of Test DataSet2 (Generated randomly from the Internet)

Domain class	Test Dataset2			
	P	Sens	Accuracy	F1
Resume	0.82	0.78	92	0.799
E-Newspaper	0.96	0.92	94	0.939
Education	0.92	0.8	91	0.855
Research	0.72	0.82	84	0.766
Shopping	0.84	0.76	88	0.798



(a) **(b)**

Figure 4.5: Results with (a) Test DataSet1 Chart- Categoriwise P,R,F1 (b) Test DataSet1 Chart- Category wise Accuracy and CV-Acc



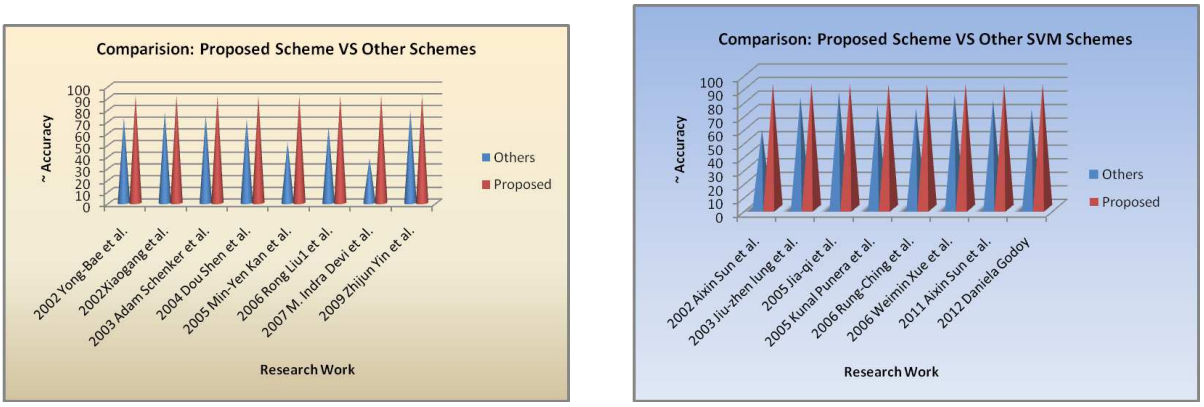
(a) **(b)**

Figure 4.6: Results with (a) Test DataSet2 Chart- Categoriwise P,R,F1 (b) Test DataSet2 Chart- Categoriwise Accuracy

Tuning parameters are set during the experiments for web page classification using grid.py python script built in libsvm tool. This script automatically adjust various parameter for learning and cross validation by recursively running the training dataset for particular kernel function till it obtains optimum result. Precision and recall are calculated after running the trained classifier on test dataset using optimum parameters obtained through grid.py python script. These parameters put the limit on the threshold values.

Data sets have been tested for various categories namely-resume, e-newspaper, education, research, shopping. The values for P, R and F1 is depicted in Fig. 4.5(a).

The value of P is 0.98 and .93 for e-newspaper and education respectively. The value of P for rest of cases is below 0.9. The R and $F1$ also follow similar trend and their values are maximum for e-newspaper. Research parameter incurs minimum value for P,R and $F1$ as visible from the Fig. 4.5(a) The value of accuracy and CV-Accuracy for DataSet1 is illustrated in Fig. 4.5(b). The accuracy is below 85% for research and is least, while the e-newspaper experiences the highest accuracy of above 95%. There is improvement in CV-Accuracy as compared to accuracy for all the parameters. The CV-Accuracy of only shopping and research parameters are below 95% and for rest of parameters, it is above 95%.



(a) **(b)**

Figure 4.7: Relative comparison of the proposed scheme (a) % accuracy of data set1 with non-SVM schemes (b) % accuracy of data set1 with other SVM schemes

The results for DataSet2 for all the parameters is depicted in Fig. 4.6. The comparative graph illustrating the variation in P, R, and $F1$ for Resume, e-newspaper, education, research and shopping is provided in Fig. 4.6(a). The value of P for education is 0.9, while for resume and shopping the P has 0.8 value each. P is least for research and maximum for e-newspaper incurring values of below 0.7 and above 0.9 respectively. The trend in value of R is different, where the maximum is still incurred by e-newspaper while shopping exhibits least value for R. The value of R for shopping reduces to below 0.75. However, the R value for research increases to 0.8. There is an increase in $F1$ values as compared to R for all the parameters except for research, where $F1$ value reduces in comparison to R value. The value of

R is maximum for e-newspaper and least for research. The accuracy for DataSet2 is depicted in Fig. 4.6(b). The accuracy for resume, education and shopping are 0.91, 0.9 and 0.87 respectively. E-newspaper has highest accuracy of 0.93 while research has lowest accuracy of 0.83. Also, figure 4.7 shows the relative comparison of the proposed scheme with respect to the other existing schemes in literature. The results obtained clearly show the effectiveness of the proposed scheme in comparison to the other existing schemes in literature. Hence, the proposed scheme is effective in web page classification as compared to the other schemes in literature.

Tables 4.3, 4.4, and 4.5 show how the proposed scheme performs better than the existing schemes of its category with respect to various selected parameters. Table 4.3 shows the major gains of using the proposed scheme in comparison to the other existing schemes in literature. Type of classifiers, feature set selection, training data sets requirements, computing costs, and performance are the key parameters which are selected for performing a relative comparison of the proposed scheme with respect to other existing schemes in literature. In Table 4.4, the level of accuracy and technique used along with the type of classifier and feature selection set are the parameters which are selected for performing the relative comparison of the proposed scheme. Simimlar parameters are selected in Table 4.5 also. From Tables 4.3, 4.4, and 4.5, it is clear that the proposed scheme performs better than the other existing schemes of its category with respect to the selected parameters. Hence, the effectiveness of the proposed scheme in comparison to other existing schemes is proved using the Tables 4.3,4.4,and 4.5.

Table 4.3: Major gains in comparison to other schemes

S.No.	References	Classifiers	Features	Class	Negative training data required	Computing cost	Performance
1	Yong-Bae *et al.* [85]	naive Bayesian	Doc freq term freq	Single	Yes	High	Medium
2	Xiaogang *et al.* [86]	Hierarchical Feature Propagation	Feature weight propagation	Single	Yes	High	Medium
3	Schenker *et al.* [87]	Graph Model	Document similarity	Single	No	High	Medium
4	Shen *et al.* [88]	SVM & Summarization	Bag of words	Single	Yes	High	Medium
5	Kan *et al.* [89]	SVM & ME	URL	Single	Yes	Low	Low
6	Devi *et al.* [90]	Naïve Bayes & SVM	URL	Single	Yes	Low	Low
7	Yin *et al.* [91]	Social Tagging Graph	Social tags	Single	No	Medium	High
8	Punera *et al.* [92]	SVM	Html tags, Document Taxonomies	Hierarchical	Yes	High	Medium
9	Liang *et al.* [41]	SVM Multi classifier	word frequency ID/TF as feature	Multi-class	Yes	High	High
10	Sun *et al.* [78]	SVM & co-training	Entities	Single	Yes	High	High
11	Godoy [93]	SVM	Social tags	Single	No	Medium	High
12	**Proposed Scheme**	**SVM-Multiclass**	**HTML tags, frequency of tags**	**Multi-class**	**No**	**Low**	**High**

Table 4.4: Comparison of Proposed Scheme with Various other Web page Classification Techniques

S.No	Reference	Classifier	Features	Technique	Accu-racy ~ %
1	Yong-Bae *et al.* [85]	naïve Bayesian classifier	Doc freq term freq(ID/TF)	Genre-Revealing and Subject-Revealing	73
2	Xiaogang *et al.* [86]	Hierarchical Feature Propagation Single path classification algorithm	Unique feature tags - weighted and propagated upwards and become the features of the parent category	Dynamic and Hierarchical Way	78
3	Schenker *et al.* [87]	Graph Model	Document similarity rather than a set of extracted features	Extension of the k-Nearest Neighbor method to work with data represented using graphs rather than numerical feature vectors.	75
4	Shen *et al.* [88]	Summarization& SVM	Bag of words, Document frequency	Adapted Luhn's Summarization Method	72
5	Kan *et al.* [89]	SVM & ME	URL as Features	Maximum entropy (ME)	52
6	Liu *et al.* [94]	Graph-based	HTML tags	Weight learning method, label propagation algorithm	65
7	Devi *et al.* [90]	Naïve Bayes & SVM	URL as Features	Machine learning methods	38
8	Yin *et al.* [91]	Social Tagging Graph	Social tags	A new link structure between objects and tags is explored for classification	79
9	**Proposed Scheme**	**SVM-Multiclass**	**On page personality features - HTML tags frequency**	**Multiclass classifier using one class against rest-No negative data**	**92**

Table 4.5: Comparison of Proposed Scheme with Various other SVM based Web page Classification Techniques

S.No.	**References**	**Classifier**	**Features**	**Technique**	**Accuracy %**
1	Sun *et al.* [38]	SVM	Html tags Text, text+title,Anchore	Tag extraction and weight assignment	58
2	Liang *et al.* [41]	SVM Multi classifier	word frequency ID/TF as feature	Scans web page by keyword dictionary and calculates the frequency of each keyword	82
3	Zou *et al.* [56]	SVM	Chinese Web Page ID/TF as feature	Statistics(CHI) based on IDF and the LI-normalization are then applied on the frequency vector to produce the feature word vector	86
4	Punera *et al.* [92]	SVM	Html tags, Document Taxonomies	Binary tree T Hierarchical Classification	76
5	Ching *et al.* [58]	SVM	Latent semantic analysis used to find frequency of words	Using a weighted vote schema	74
6	Xue *et al.* [95]	SVM	Html tags-Title, body, head,meta tbm, tbmh	comparison on the polynomial kernel function and the radius basis function (RBF)	84
7	Sun *et al.* [78]	SVM	Entities	Entity-based co-training	80
8	Godoy *et al.* [93]	SVM	Social tags	Personalized tag-based resource classification	73
9	**Proposed Scheme**	**SVM-Multiclass**	**On page personality features - HTML tags frequency**	**Multiclass classifier using one class against rest-No negative data**	**92**

4.11 Summary

Information retrieval and data mining are two most powerful techniques used for efficient operations such as- data dissemination and communication in next generation wireless communication networks. With an increase in the user's requests from different geographical locations, there may be a performance bottleneck in some part of the networks. Also, with passage of time, many users want the service availability with respect to the web page access from anywhere with respect to the parameters such as-fast response time and increased accuracy. To address these issues, this chapter proposed a novel technique for web page classification using on-page personality features with multiclass classifier. Algorithms for feature extraction and multiclass classification are designed in the proposed solution. A detailed complexity analysis of the proposed scheme is also provided in the text to evaluate the designed scheme. The performance of the proposed scheme is evaluated using different evaluation metrics where its performance was found satisfactory with respect to the selected parameters. The overall increment in accuracy with respect to selected parameters of the proposed scheme in comparison to the other existing schemes is found out to be 10-15 %. Overall gain in accuracy is mainly due to nature of data, relevant feature set selection that is most suited for particular domain web pages such as E-newspaper, research, online shopping, education etc. It also depends on selection of the kernel by comparing cross validation results of various SVM kernels and kernel parameters obtained through grid technique.

In the future, various other features for multiclass classifier would be explored for other domains for getting the better results.

Chapter 5

Conclusion and future scope

5.1 Conclusion

Web classification or categorization is a challenging, interesting and important task for the effective usage of huge amount of information in the form of web documents. Many researchers worked on web document classification. Researchers used various feature sets, classification approaches and algorithms. Some researchers got promising results. Still there was large space for improvement in automatic web page classification accuracy. In the proposed work, large number of schemes present in the literature was studied from various angles. Comparative study of classification approaches and feature selection methodologies were done to identify the areas of improvement. Some researchers in the past used URL as feature, some used link analysis as feature, some used <head> and <title> as feature, some used <body> text as feature. Even external links, neighbor page feature and social tags were also tried to classify the web document. Some researcher extracted plan text from html tags and removed stop words to try the classification problem like simple text categorization. Existing approaches were not successful to produce the higher accuracy and some of these techniques were computationally extensive and costly. In the proposed work web page classification problem is re-looked as fresh problem. Careful selection and extraction of feature set is done. Exhaustive features are ex-

plored to generate the feature sets. In proposed work many html tags are consider for feature selection. Even some tags were not present in earlier html versions like <audio>, <video> and other multimedia related tags. Use of meta tags which were rarely present in earlier pages but present in more recent pages and professionally developed pages are also explored for probable candidate for feature set. <img> alternate text is also consider for feature set. <A href> hyperlink tag and alternate text are also part of feature set. Proposed approach also relied on <table> , <ol>, <li> etc. tags. Further number of links, number of images, count of multimedia tags and word frequency are given equal importance in deciding the category of web page.

In proposed approach two significant schemes are developed. One is corresponding to binary classification and the other corresponding to multi class classification. After examining the various classification approaches support vector machine is selected to train the classification model. Linear kernel, Radial Bias Kernel and Polynomial kernel function are tried to compare the performance of various kernels in order to achieve higher accuracy. To improve the performance of trained classification model various kernel parameter and their values are tried. Weights are assigned to feature based on tf-idf and frequency of feature on page. During this process the preliminary feature set is reduced by eliminating less important features corresponding to domains. This is achieved by applying dimension reduction technique. Further data is scaled in the range of [-1,1] and converted to SVM readable format.

In binary classification approach exhaustive feature set is constructed by combining the information from various sources. Word stemming is also applied on the extracted feature text to reduce misclassification. Experimental results show higher accuracy as compare to other existing techniques in the literature.

In multi class classification approach on page personality features of the web page

are considered. Various domains like education, online shopping, E-news paper etc. are studied to uniquely identify the personality features to generate a feature set corresponding to each domain. Further, a combined feature set is generated by joining the feature set of each domain. In multiclass classification technique instead of text, on page personality feature frequency is considered. This helped in reducing the computation cost. Further multiclass classification problem is solved by converting the multi class problem to binary class problem and solving the problem as one class against the rest. Experimental results produced by this technique are quite promising.

Finally, this work also provided a comparison between the most used html tags and other sources such as URL of web page, link analysis and external links as probable features for the web page classification. This comparison led to the conclusion that the combinations of features from various sources can improve the results obtained by the individual feature source, even when they initially showed a good performance, and that the improvement also depends on the difficulty level of the dataset that was used.

5.2 Future Scope

The proposed work has lots of scope for extension and improvement. In future, more features from the neighboring pages can be considered to improve the results. N-grams can be considered for targeted classification of the web documents. Implicit and explicit links can be explored as feature to improve the accuracy level.

The proposed technique can be extended to classify the web pages developed in the regional languages. The Proposed research work can be extended to build domain ontology based web page classification system. This work can also be extended to build hierarchical structure based classification scheme. These types of classifica-

tion schemes may not only improve the search engine results but it may also improve the result of question–answer systems and query based information systems. Cataloging and Knowledge base management systems of web resources can be developed by extending the proposed work schemes. Huge information in the form of web pages can be effectively used for various analytical purposes through web mining by application of proposed research work. Some prominent research areas can also be explored such as mentioned in [98-105] where the proposed scheme can be applied. These research areas are from hetetrogeneous domains and are expected to expedite in the years to come.

Web content classification can be extended further to specifically cater the need of fast and relevant data for Internet of things (IoT). Proposed solution can be relooked for improvements due to increase in the size of the data, types of devices and upcoming html tags used in the future. This is because Internet enabled devices such as smart vending machines, smart TV, smart watches, smart homes, smart automobile and smart city and related industrial applications usage poses further challenge to its quality of service.

The proposed research work can be used to support many semantic web based applications and software tools. Furthermore, the proposed work can be extended to accommodate the targeted web based advertising system. Blog classification and Email classification can be achieved through extension of the proposed work.

LIST OF PUBLICATIONS

(1) Vinod Kumar Bhalla, Neeraj Kumar, An efficient scheme for automatic web pages categorization using the support vector machine, *The New Review of Hypermedia and Multimedia*, vol. 22, no. 3, pp. 223-242, 2016. [**SCIE**]

(2) Vinod Kumar Bhalla, Neeraj Kumar, An Efficient Multiclass Classifier Using On-Page Positive Personality Features for Web Page Classification for the Next Generation Wireless Communication Networks, *Wireless Personal Communications*, In Press, DOI:10.1007/s11277-016-3173-4. [**SCIE**]

Bibliography

[1] Pierre, J.M., (2000). Practical Issues for Automated Categorization of Web Pages, Proceeding ECDL 2000 workshop on the Semantic Web.

[2] Pinkerton, B. (1994). Finding what people want: Experiences with the WebCrawler. In Proceedings of the First World Wide Web Conference. Geneva, Switzerland, pp. 17-20.

[3] Chih-Ming Chen, Hahn-Ming Lee, Yu-Jung Chang (2009). Two novel feature selection approaches for web page classification, Expert Systems with Applications 36, pp. 260–272.

[4] John M. Pierre (2001). On the Automated Classification of Web Sites, Linkoping Electronic Articles in Computer and Information Science, Vol. 6(2001): nr 0. http://www.ep.liu.se/ea/cis/2001/000/. February 4, 2001.

[5] Lin Li, Luo Zhong, Guandong Xu, Masaru Kitsuregawa (2012). A feature-free search query classification approach using semantic distance, Expert Systems with Applications 39, pp. 10739–10748.

[6] Chakrabarti, S. (2003). Mining the Web: Discovering Knowledge from Hypertext Data, San Francisco, CA: Morgan Kaufmann.

[7] Yang, H. and T.-S. Chua (2004a). Effectiveness of web page classification on finding list answers. In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, pp. 522–523. ACM Press.

[8] Nie, L., B. D. Davison, and X. Qi (2006, August). Topical link analysis for web search. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval, New York, NY, pp. 91–98. ACM Press.

[9] Pietramala, A., Policicchio, V. L., Rullo, P., & Sidhu, I. (2008). A genetic algorithm for text classification rule induction, Lecture Notes in Artificial Intelligence, 5212, pp. 188–203.

[10] Blum, A. and T. Mitchell (1998). Combining labeled and unlabeled data with co-training. In COLT' 98: Proceedings of the 11th Annual Conference on Computational Learning theory, New York, NY, pp. 92–100. ACM Press.

[11] Hung-Yi Lin (2012). Efficient classifiers for multi-class classification problems, Decision Support Systems 53, pp. 473–481.

[12] Deuk Hee Park, Hyea Kyeong Kim, Il Young Choi, Jae Kyeong Kim (2012). A literature review and classification of recommender systems research, Expert Systems with Applications 39, pp. 10059–10072.

[13] Kovacevic, M., Diligenti, M., Gori, M., And Milutinovic, V. (2004). Visual adjacency multigraphs—a novel approach for a Web page classification. In Proceedings of the Workshop on Statistical Approaches to WebMining (SAWM). pp. 38–49.

[14] Shen, D., J.-T. Sun, Q. Yang, and Z. Chen (2006). A comparison of implicit and explicit links for web page classification. In Proceedings of the 15th International Conference on World Wide Web, New York, NY, pp. 643–650. ACM Press.

[15] Qi, X. and B. D. Davison (2006). Knowing a web page by the company it keeps. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM), New York, NY, pp. 228–237. ACM Press.

[16] Calado, P., Cristo, M., Moura, E., Ziviani, N., Ribeiro-Neto, B., And Goncalves, M. A. (2003). Combining link-based and content-based methods for Web document classification. In Proceedings of the 12th International Conference on Information and Knowledge Management (CIKM). ACM Press, New York, NY, pp. 394–401.

[17] Kwon, O.-W. and J.-H. Lee (2003, January). Text categorization based on k-nearest neighbor approach for web site classification. Information Processing and Management 29 (1), pp. 25–44.

[18] Yu, H., Han, J., And Chang, K. C.-C. 2004. PEBL:Web page classification without negative examples. IEEE Trans. Knowl. Data Eng. 16, 1, pp. 70–81.

[19] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, 5th Annual ACM Workshop on COLT, pp. 144-152, Pittsburgh, PA, 1992. ACM Press.

[20] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), pp. 273–297.

[21] Bai, R., Wang, X., & Liao, J. (2007). Combination of rough sets and genetic algorithms for text classification. In Proceedings of second international workshop, AIS-ADM 2007, pp. 256–268. St.Petersburg, Russia.

[22] Z.Q. Liu, Y. Zhang (2001). A competitive neural network approach to web-page categorization, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 9 (6), pp. 731–741.

[23] Kwon, O.-W. and J.-H. Lee (2000). Web page classification based on k-nearest neighbor approach. In IRAL '00: Proceedings of the 5th International Workshop on Information Retrieval with Asian languages, New York, NY, pp. 9–15. ACM Press.

[24] C. Enhong, W. Shangfei, Z. Zhenya, W. Xufa, (2001). Document classification with CC4 neural network, in: Proceedings of ICONIP 2001, Sanghai, China.

[25] Nicholas Holden and Alex A. Freitas (2004), Web Page Classification with an Ant Colony Algorithm Parallel Problem Solving from Nature - PPSN VIII Lecture Notes in Computer Science Volume 3242, 2004, pp. 1092-1102.

[26] Sen, P. and L. Getoor (2007). Link-based classification. Technical Report CS-TR-4858, University of Maryland.

[27] Angelova, R. and S. Siersdorfer (2006). A neighborhood-based approach for clustering of linked document collections. In CIKM '06: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, New York, NY, pp. 778–779. ACM Press.

[28] Chekuri, C., Goldwasser, M., Raghavan, P., And Upfal, E. 1997. Web search using automated classification. In Proceedings of the Sixth International World Wide Web Conference (Santa Clara, CA). Poster POS725.

[29] KAKI, M. 2005. Findex: Search result categories help users when document ranking fails. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI). ACM Press, New York, NY, pp. 131–140.

[30] Kohlschutter, C., P.-A. Chirita, and W. Nejdl (2007). Utility analysis for topically biased PageRank. In WWW '07: Proceedings of the 16th International Conference on World Wide Web, New York, NY, pp. 1211–1212. ACM Press.

[31] Huang, C.-C., Chuang, S.-L., And Chien, L.-F. (2004a). Liveclassifier: Creating hierarchical text classifiers through Web corpora. In Proceedings of the 13th International Conference on World Wide Web (WWW). ACM Press, New York, NY, pp. 184–192.

[32] Chen, Z., O. Wu, M. Zhu, and W. Hu (2006). A novel web page filtering system by combining texts and images. In WI '06: Proceedings of the 2006

IEEE/WIC/ACM International Conference on Web Intelligence, Washington, DC, pp. 732–735. IEEE Computer Society.

[33] Broder, A., Fontoura, M., Josifovski, V., And Riedel, L. 2007a. A semantic approach to contextual advertising. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, New York, NY, pp. 559–566.

[34] Stuckenschmidt, H., Hartmann, J., & Van Harmelen, F. (2002). Learning structural classification rules for web-page categorization. In FLAIRS conference, pp. 440–444.

[35] Denoyer, L., Zaragoza, H., & Gallinari, P. (2001, March). HMM-based passage models for document classification and ranking. In Proceedings of ECIR-01, 23rd European colloquium on information retrieval research seattle, pp. 126–135. WA, USA.

[36] Selamat, A., & Omatu, S. (2004). Web page feature selection and classification using neural networks. Information Sciences, 158, pp. 69–88.

[37] Tan, S. (2005). Neighbor-weighted k-nearest neighbor for unbalanced text corpus. Expert Systems with Applications, 28(4), pp. 667–671.

[38] Sun, A., Lim, E. P., & Ng, W. K. (2002, November). Web classification using support vector machine. In Proceedings of the 4th international workshop on Web information and data management, pp. 96–99.

[39] Zhang, M. L., Pea, J. M., & Robles, V. (2009). Feature selection for multi-label naive Bayes classification. Information Sciences, 179(19), pp. 3218–3229.

[40] Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. IEEE Transactions on Neural Networks, 13(2), pp. 415–425.

[41] Liang, J. Z. (2004). SVM multi-classifier and web document classification. In Proceedings of international conference on machine learning and cybernetics, 2004, vol. 3, pp. 1347–1351. Shanghai.

[42] Koller, D., & Sahami, M. (1997). Hierarchically classifying documents using very few words. In Proceedings of 14th international conference on machine learning ICML-97, pp. 170–178. San Francisco, Nashville, USA.

[43] Dumais, S., & Chen, H. (2000, July). Hierarchical classification of web content. In Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 256–263. New York, USA.

[44] Liang, J. Z. (2003). Chinese web page classification based on self-organizing mapping neural networks. In Proceedings fifth international conference on computational intelligence and multimedia applications, ICCIMA 2003, pp. 96–101. Wan, China.

[45] Benbrahim, H., & Bramer, M. (2004, October). An empirical study for hypertext categorization. In IEEE international conference on systems, man and cybernetics, pp. 5952–5957.

[46] Lim, C. S., Lee, K. J., & Kim, G. C. (2005). Multiple sets of features for automatic genre classification of web documents. Information Processing & Management, 41(5), pp. 1263–1276.

[47] Attardi, G., Gulli, A., & Sebastiani, F. (1999). Automatic Web page categorization by link and context analysis. In Proceedings of THAI-99, European symposium on telematics, hypermedia and artificial intelligence, pp. 105–119.

[48] Riboni, D. (2002). Feature selection for web page classification. In Proceedings workshop, pp. 473–478.

[49] Quek, C. Y., & Mitchell, T. (1997). Classification of world wide web documents. Master's thesis, School of Computer Science Carnegie Mellon University.

[50] Yang, Y., Slattery, S., & Ghani, R. (2002). A study of approaches to hypertext categorization. Journal of Intelligent Information Systems, 18(2–3), pp. 219–241.

[51] Hodgson, J. (2001). Do HTML tags flag semantic content. Internet Computing, 5(1), pp. 20–25.

[52] Frnkranz, J. (1999). Exploiting structural information for text classification on the WWW. In IDA '99 proceedings of the third international symposium on advances in intelligent data analysis, pp. 487–497. London, UK: Springer.

[53] Internet source. http://www.dmoz.org/ Open Directory Project (ODP).

[54] Internet source yahoo! Directory.

[55] Aliakbary, S., Abolhassani, H., Rahmani, H., & Nobakht, B. (2009). Web page classification using social tags. In Computational science and engineering, 2009. CSE '09. International conference on, vol. 4, pp. 588–593.

[56] Zou, J., Chen, G.-L., & Guo, W.-Z. (2005). Chinese web page classification using noise-tolerant support vector machines. In Natural language processing and knowledge engineering, 2005. IEEE NLP-KE '05. Proceedings of 2005 IEEE international conference, pp. 785–790.

[57] Cakrabarti,S., Dom, B., and Indynk, P.(1998) Enhanced Hypertext Categorization Using Hyperlink. In: L, Haas and A, Tiwary (eds.), Proc. SIGMOD-98, ACM International Conference on Management of Data, pp. 307-318.

[58] Chen, R. and Hseih, C. (2006) Web page classification based on a support vector machine using a weighted vote schema. Expert Systems with Applications, 31, pp. 427-435.

[59] Cohen, W.W. (2002) Improving a Page Classifier with Anchor Extraction and Link Analysis. Advances in Neural Information Processing Systems, 15, pp. 1505-1520 .

[60] Yang, K. (2002) Combining Text, Link and Classification based Retrieval Methods to Enhance Information Discovery on the Web. University of North Carolina, United States.

[61] Brin, S., and Page, L. (1998) The anatomy of a large-scale hyper textual Web search engine. Computer Networks and ISDN Systems, 3, pp. 107-117.

[62] Hernndez, I., Rivero, C.R., Ruiz, D., Corchuelo, R. (2014) CALA: An unsupervised URL-based web page classification system, Knowledge-Based Systems. 57, pp. 168-180.

[63] Uzun, E., Agun, H.V., Yerlikaya, T. (2013) A hybrid approach for extracting informative content from web pages. Information Processing & Management. 49, 4, pp. 928-944.

[64] Murthy, K.A. Suresha (2015) XML URL Classification Based on their Semantic Structure Orientation for Web Mining Applications. Procedia Computer Science. 46, pp. 143-150.

[65] Du, Y., Li, Q., Cai, Z., Guan, X. (2013) Multi-view semi-supervised web image classification via co-graph. Neuro computing, 122, 25, pp. 430-440.

[66] Olson, D.L., and Dursun, D. (2008) Advanced Data Mining Techniques, Springer, 1, 138, ISBN 3-540-76916-1.

[67] Sebastiani, S. (2002) Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34, pp. 1-47.

[68] Gawande, K., Webers, C., Smola, A., Vishwanathan, S.V.N., Gunter, S., Teo, C.H., Shi, J.Q., McAuley, J., Song, L., and Le. Q.(2007). ELEFANT user manual (revision 0.1). Technical report, NICTA.

[69] Joachims, T. (2006) Training linear SVMs in linear time. In ACM SIGKDD Inter- national Conference On Knowledge Discovery and Data Mining (KDD), pp. 217-226.

[70] Internet source, http://google.indicateur.biz/ google source

[71] WebKB dataset. Available: http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-51/www/co-training/data

[72] 7Sectors dataset. Available: http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo 11/www/wwkb/index.html

[73] Levering, R., Cutler, M., and Yu, L. (2008). Using visual features for fine-grained genre classification of web pages. In Proceedings of the 41st Hawaii International Conference on System Sciences, pp. 131-140, Waikoloa, HI.

[74] Burget, R., and Rudolfova, I. (2009). Web page element classification based on visual features. In 2009 First Asian Conference on Intelligent Information and Database Systems, pp. 67-72, Dong Hoi.

[75] Vaughan, L., Tang, J., and Du, J. (2010). Constructing business profiles based on keyword patterns on Web sites. Journal of the American Society for Information Science and Technology, 61(6), pp. 1120-1129.

[76] Unler, A., and Murat, A. (2010). A discrete particle swarm optimization method for feature selection in binary classification problems. European Journal of Operational Research, 206(3), pp. 528-539.

[77] Ozel, S. A. (2011). A web page classification system based on a genetic algorithm using tagged-terms as features. Expert Systems with Applications, 38(4), pp. 3407-3415.

[78] Sun, A., Liu, Y., and Lim, E. P. (2011). Web classification of conceptual entities using co-training. Expert Systems with Applications, 38(12), pp. 14367-14375.

[79] Sabbah, T., Selamat, A., Selamat, M. H., Ibrahim, R., and Fujita, H. (2015). Hybridized term-weighting method for Dark Web classification. Neurocomputing, doi:10.1016/j.neucom.2015.09.063.

[80] Kim, D. W., Yan, P., and Zhang, J. (2015). Detecting fake anti-virus software distribution webpages. Computers & Security, 49, pp. 95-106.

[81] Wang, J., Peng, J., and Liu, O. (2015). A classification approach for less popular webpages based on latent semantic analysis and rough set model. Expert Systems with Applications, 42(1), pp. 642-648.

[82] Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. ACM Transaction Intelligent Systems and Technology, 2(3), 27. http://www.csie.ntu.edu.tw/cjlin/libsvm.

[83] Hsu, C. H., Chang, C. C. & Lin, C. J. (2003). A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.

[84] Chapelle, O. (2007). Training a support vector machine in the primal. Neural Computation, 19(5), pp. 1155–1178.

[85] Lee, Y.-B., & Myaeng, S. H. (2002). Text genre classification with genre-revealing and subject-revealing features. In Proceeding SIGIR '02 Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval, pp. 145–150.

[86] Peng, X., & Choi, B. (2002). Automatic web page classification in a dynamic and hierarchical way. In Data mining, 2002. ICDM 2003. Proceedings. 2002 IEEE international conference, pp. 386–393.

[87] Schenker, A., Last, M., Bunke, H., & Kandel, A. (2003). Classification of web documents using a graph model. International Journal of Pattern Recognition and Artificial Intelligence, 18(03), pp. 475–496.

[88] Shen, D., Chen, Z., & Yang, Q., (2004). Web-page classification through summarization. In Proceeding SIGIR '04. Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval, pp. 242–249.

[89] Kan, M.-Y., & Hoang Oanh Nguyen, T. (2005). Fast webpage classification using URL features. In Proceeding CIKM '05. Proceedings of the 14th ACM international conference on Information and knowledge management, pp. 325–326.

[90] Devi, M. I., Rajaram, R., & Selvakuberan, K. (2007). Machine learning techniques for automated web page classification using URL features. In Conference on computational intelligence and multimedia applications, 2007. international conference, vol. 2, pp. 116–120.

[91] Yin, Z., Li, Z., Mei, Q., & Han, J. (2009). Exploring social tagging graph for web object classification. In Proceeding KDD '09. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 957–966.

[92] Punera, K., Rajan, S., & Ghosh, J., (2005). Automatically learning document taxonomies for hierarchical classification. In Proceeding WWW '05. Special interest tracks and posters of the 14th international conference on world wide web, pp. 1010–1011.

[93] Godoy, D. (2012). One-class support vector machines for personalized tag-based resource classification in social bookmarking systems. Concurrency and Computation: Practice and Experience, 24(17), pp. 2193–2206.

[94] Liu, R., Zhou, J.,& Liu, M. (2006). A graph-based semi-supervised learning algorithm for web page classification. In Intelligent systems design and applications, 2006. ISDA '06. Sixth international conference, vol. 2, pp. 856–860.

[95] Xue, W., Bao, H., Huang, W., & Lu, Y. (2006). Web page classification based on SVM. In Intelligent control and automation, 2006. WCICA 2006. The sixth world congress on, vol. 2, pp. 6111–6114.

[96] Eickhoff, C., Serdyukov, P., and de Vries, A. P. (2010). Web page classification on child suitability. In Proceedings of the 19th ACM international conference on Information and knowledge management, pp. 1425-1428, New York, NY, USA.

[97] Gao, M., Tian, J., and Zhou, S. (2009). Research of web classification mining based on classify support vector machine. In 2009 ISECS International Colloquium on Computing, Communication, Control, and Management, pp. 21-24, Sanya.

[98] Shyam, G. K., Manvi, S. S. (2015). Modelling resource virtualisation concept in cloud computing environment using finite state machines. IJCC 4(3), pp. 258-278.

[99] Singh, R., Sharma, T.P.,(2015). On the IEEE 802.11i security: a denial-of-service perspective. Security and Communication Networks, 8(7), pp.1378-1407.

[100] Almuairfi, S., Veeraraghavan, P., Chilamkurti, N. (2013). A novel image-based implicit password authentication system (IPAS) for mobile and non-mobile devices. Mathematical and Computer Modelling, 58(1-2), pp. 108-116.

[101] Tian, F., Wu, F., Chao, K.M., Zheng, Q., Shah, N., Lan, T., Yue, J. (2016) A topic sentence-based instance transfer method for imbalanced sentiment classification of Chinese product reviews. Electronic Commerce Research and Applications, 16, pp. 66-76.

[102] Sharma, S., Lather, J.S., Dave, M. (2016). Semantic approach for Web service classification using machine learning and measures of semantic relatedness. Service Oriented Computing and Applications, 10(3), pp.221-231.

[103] Lin, C. C., Deng, D. J., Wang, S. B. (2016). Extending the Lifetime of Dynamic Underwater Acoustic Sensor Networks Using Multi-Population Harmony Search Algorithm. IEEE Sensors Journal, 16(11), pp. 4034-4042.

[104] Akuma, S., Iqbal, R., Jayne, C., Doctor, F. (2016). Comparative analysis of relevance feedback methods based on two user studies. Computers in Human Behavior, 60, pp. 138-146.

[105] Shrivastava, V., Misra, R.B., (2010). Development of Bayesian belief network model for electrical load demand. Int. J. Systems Assurance Engineering and Management, 1(2), pp.170-177.