Web Page Categorization based on Characteristics of Web Page

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Khushboo Taneja
801032012

Under the supervision of:
Mr. Vinod Kr. Bhalla
Assistant Professor, CSED

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2012

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "*Web Page Categorization based on Characteristics of Web Page*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in *Computer Science and Engineering* Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Mr. Vinod Kr. Bhalla* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

(Khushboo Taneja)

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

Mr. Vinod Kr. Bhalla
Computer Science and Engineering Department
Thapar University
Patiala

Countersigned By:

(Dr. Maninder Singh)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

First of all, I am thankful to God for his blessings and showing me the right direction. With His mercy, it has been made possible for me to reach so far.

It is a great privilege to express my gratitude and admiration toward my respected supervisor **Mr. Vinod Kr. Bhalla**. He has been an esteemed guide and a great support behind achieving the task. Without his able guidance, kind efforts and encouragement, the work would not have been what it is. I am truly grateful to him for extending his total co-operation and understanding whenever I needed help and guidance from him.

I wish to express my heartiest thanks to Dr. Maninder Singh, Head, Computer Science and Engineering Department, Thapar University, Patiala for providing me the opportunity and all necessary facilities to accomplish this thesis successfully.

I would also like to say thanks to my friends who were always there at the need of hour and provided help for the completion of my thesis work.

I am grateful to my parents who soulfully provided me their constant support and encouraging attitude to undertake the challenge of this proportion. They believed in before I believed in myself. To them I owe my wonderful today and dream filled future.

Khushboo Taneja
(801032012)

Abstract

Internet is the source of enormous amount of information accessed by large number of people every day. Contemporary web is comprised of trillions of pages and everyday tremendous amount of requests are made to put more web pages on the WWW. It has been difficult to manage information present on web than to create it. Web page categorization can be defined as an approach to categorize the web pages based on a set of predefined categories to manage large web content. Yahoo! [3] and ODP [4] are the examples of web directories in which pages are categorized manually or semi automatically, but it is a very time consuming task. There are many ways of categorizing web pages using different techniques. This thesis proposes an approach to categorize web pages automatically on the basis of characteristics of web pages using neural network based single discrete perceptron training algorithm which is extended by selecting web page specific features to categorize web pages of predefined categories with high accuracy. The idea is presented with the help of two specific and major categories of web pages chosen for categorization, that are newspaper and education. The approach can be effectively used to categorize web pages into broad categories. The whole approach can be described in three steps. In the first step, features are extracted automatically after analyzing the source web pages. The second step includes the implementation and training of the algorithm. Finally, the third step will categorize the source web pages into one of the two categories.

The proposed approach can be used to categorize commercial and non commercial sites, blogs and non blogs, social networking sites and non social networking sites.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

1. WWW: World Wide Web
2. Yahoo!: Yahoo Web Directory
3. URL: Uniform Resource Locator
4. ODP: Open Directory Project
5. SVM: Support Vector Machine

Chapter-1

Introduction

The growing number of applications on the web leads to rapid increase in number of web pages. The data available on the web can be in the form of text, images, audio, video, graphics and many other forms. Web pages present on the web can be static or dynamic. The content of dynamic web pages keeps on changing time to time. Web is considered as a large repository of information which is accessed by millions of users' everyday through internet. The dynamic nature of web and large scale explosion of web pages may put a threat to efficient information retrieval tasks. Web can be considered as an information resource, therefore it is important to describe and organize the huge content present on the web in order to realize web's full potential [1]. Thus web page categorization is an intellectual task, important and indeed essential for organizing and understanding web content for different applications, efficient information retrieval and other tasks related to web mining.

This chapter will discuss some facts about web page categorization including the types of web page categorization, need of web page categorization and various characteristics of web pages. At the end of this chapter the proposed idea of the new approach to *categorize web pages based on characteristics of web page* is presented.

1.1 Web Page Categorization

Web page categorization also known as web page classification is the process of assigning a web page to one or more predefined category labels [2]. Categorization is often considered as a supervised learning problem in which a labeled data set is used to train a classifier which can be applied to classify and label the test data. The training and testing data can be collected from different sources in order to achieve high performance of the categorizer. Figure 1.1 shows the basic categorization method.

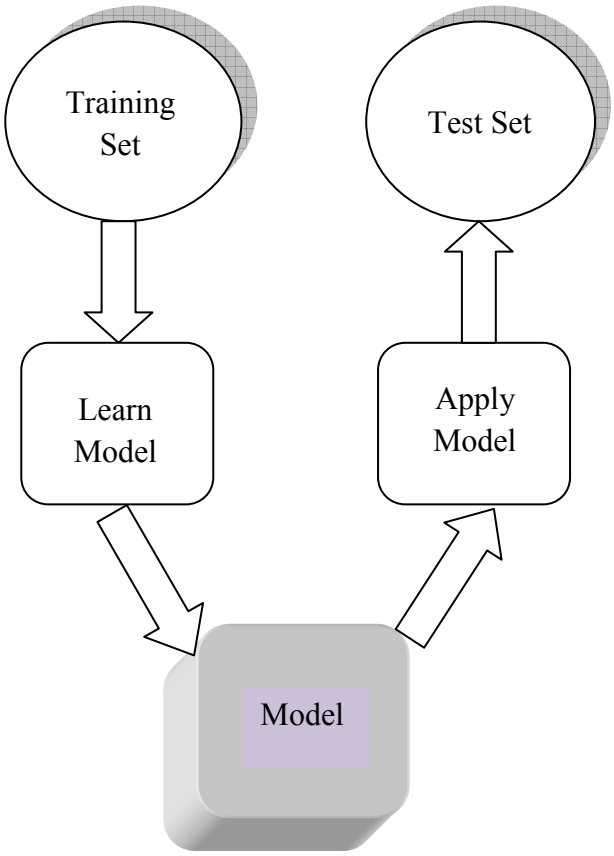


Figure 1.1: Basic Categorization Method

Many of the web page categorization techniques are based on the concept of supervised learning. In supervised learning, desired output is known. During training the desired output is compared with the actual output. If any error is found, it is fed back to make corrections, in this way training will continue until the actual output becomes equal to the desired output. The proposed approach in this thesis is also based on supervised learning.

There are different types of categorization schemes as needed by different applications. Some of the categorization schemes are described in the following section. However this thesis will focus mainly on binary categorization and subject categorization.

1.2 Types of Web Page Categorization

There are two types of categorizations based on the number of categories:

- **Binary Categorization:** It categorizes the web page into exactly one of two categories.
- **Multi-class Categorization:** It categorizes the web page into one of many categories.

Other types of categorizations are:

- **Subject Categorization:** It categorizes the web page according to its subject or topic. For example, categorizing the web page as "science", "sports" or "politics" is an instance of subject categorization.
- **Functional Categorization:** It categorizes the web page according to its role. For example categorizing the web page as "research page", "homepage" or "information page is an instance of functional categorization.
- **Sentiment Categorization:** It categorizes the web page according to the author's attitude about any particular topic.
- **Genre Categorization:** It categorizes the web page with respect to its form or functional trait. For example when analyzing newspaper articles typical genres include "editorial", "letter", "reportage" and "spot news".

On the basis of organization of categories web page categorization can also be divided into two types as explained below:

- **Flat Categorization:** In flat categorization, categories are considered parallel. The categories like "business", "sports", "health" forms a flat categorization because no category can supersede the other category.
- **Hierarchical Categorization:** In hierarchical categorization one category can supersede the other categories.

Figure 1.2 and 1.3 gives a brief idea about flat categorization and hierarchical categorization respectively.

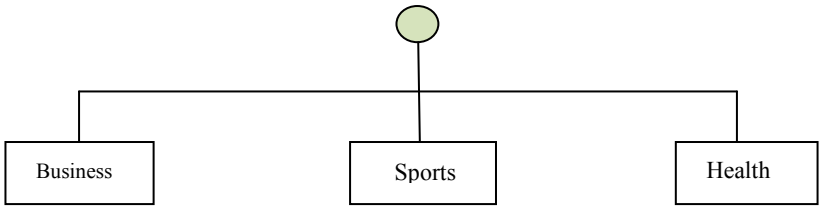


Figure 1.2: Flat Categorization

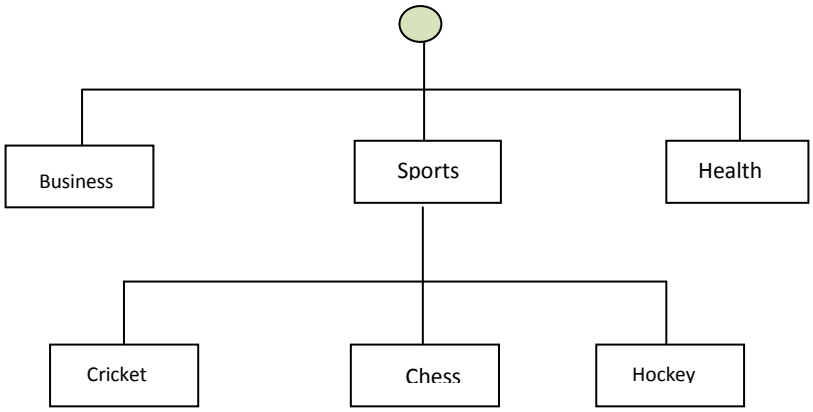


Figure 1.3: Hierarchical Categorization

1.3 Need of Web Page Categorization

Web page categorization is needed due to the following reasons:

- It helps in efficient retrieval of web pages.
- It provides an aid to topical crawlers which search the web for a particular topic.
- It helps in maintenance and development of web directories.
- It helps in topic specific web link analysis.
- It helps in increasing the quality of search results. Categorized results present a good user interface than the search results which are presented in a ranked list [2].

1.4 Characteristics of Web Page

A web page has the following characteristics:

- It is a semi structured document in HTML.
- It consists of text, images, links, videos and other multimedia content.
- It is connected to other pages through hyperlinks thus forming a graphical structure on the web.
- It is rendered to user by the web browser.

From the very beginning categorization was done manually by domain experts. Yahoo! [3] and ODP [4] are the examples of web directories which are developed manually. But with the rapid increase of web pages it became extremely difficult to categorize web pages manually. Therefore categorization began to be done semi automatically or automatically. There are a number of approaches which have been applied in the field of web page categorization including K-Nearest Neighbor approach [11], Bayesian probabilistic models [12][13][14], inductive rule learning [15], decision trees [14], neural networks [16] and support vector machines [17]. All the above mentioned approaches are based only on the text content of the web pages. Besides text content other features like images, links, videos etc can also be used for categorization of web pages.

In the proposed approach which is described in this thesis, the characteristics of web pages like number of links, number of images and number of words or the amount of text have been used to categorize the web pages into one of the two categories. The idea is presented using source web pages of two major categories or domains: Newspaper and Education. After analyzing the web pages belonging to newspaper sites and education sites, it has been found that newspaper web pages contain more number of links, images and words than education web pages. The difference in these characteristics is used for categorization. Figure 1.4 and 1.5 shows a newspaper web page and education web page respectively. The difference in the number of links, images and words can be seen clearly in the figures.

Figure 1.4: Newspaper Web Page

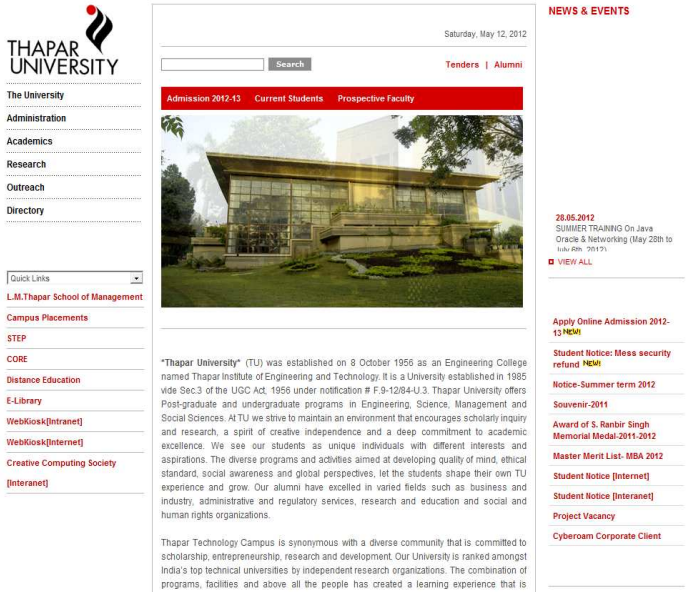


Figure 1.5: Education Web Page

These features are extracted and analyzed. Neural network based single discrete perceptron training algorithm is used to form a binary categorizer which can categorize the web pages into one of the two categories. Training and testing data are obtained from different websites and Yahoo! [3] web directory.

1.5 Structure of the Thesis

The rest of thesis is organized in the following order:

Chapter 2: This chapter will provide the overview of various approaches which have been used in the area of web page categorization and will discuss the concept of neural networks.

Chapter 3: This chapter will give the problem statement and the methodology used to solve the problem.

Chapter 4: This chapter will give solution to the problem described in chapter 3 using neural network based single layer perceptron training algorithm.

Chapter 5: This chapter will give testing and performance results of the algorithm given in chapter 4.

Chapter 6: This chapter will give the conclusion of the thesis with the future scope of the topic.

Chapter-2

Literature Review

Web page categorization is a fundamental problem these days due to rapid increase in the number of web pages. The need for automated categorization of web pages is for at least two reasons. One reason is the large number of resources present on the web and their ever-changing nature. It is not possible to manage such dynamic nature of web manually without a lot of human effort and time. The second reason is that categorization itself is a subjective activity; different applications depend upon different classification schemes. Therefore different types of categorization schemes, representing different facets of knowledge may need to be applied in an ongoing fashion due to large scale increase in applications [1]. A number of techniques have been used for the categorization of web pages based on different approaches as described below.

2.1 Web Page Categorization Techniques

The categorization techniques can be classified into the following broad categories:

- Categorization by domain experts
- Clustering approaches
- Meta tags based approach
- Text content based categorization
- Link and Content Analysis

In manual categorization approach, categorization is done by domain experts. However it is a very time consuming task and it takes a lot of human effort to categorize the large number of web pages.

Clustering algorithms have been used to form clusters of related web pages to make classification easier and faster. However these algorithms are static because most of the clustering algorithms like K- Means etc. require the number of clusters to be specified in advance.

Meta tags based approach relies on the use of meta tags in web pages like <META name="keywords"> and <META name="description">. However this approach fails in the cases where web pages don't contain meta tags.

In text content based categorization, a database of keywords is prepared by calculating the frequency of occurrence of words and phrases in a category. The commonly occurring words like "the", "and", "of" etc. are removed from database and the remaining keywords are then used for categorization.

The link and content analysis is based on the hyperlinks and anchor text present on the web page which gives enough hints about referred page.

Every web page categorization technique involves the following steps for web page categorization:

Step 1: Understand completely the domain to be categorized.

Step 2: Collect training data for the categorization.

Step3: Pre-process data by reducing the dimensions of feature set as required by the categorization algorithm.

Step 4: Put the categorizer on training.

Step 5: Apply the test data to the categorizer.

Step 6: Evaluate the results.

A number of researches have been done in the field of web page categorization. In [12] web pages are categorized on the basis of summaries of web pages, a summarization based

algorithm is proposed to solve the problem of categorization. In [19] web pages are categorized without the web page, URLs are used to categorize the web pages via a two-phase pipeline of word segmentation/expansion and classification. In [20] web pages are categorized in the hierarchical structure using SVM classifiers. In [21] the concept of domain ontology has been introduced in the field of automatic classification of web pages. It involves determining document features that represent the web documents most accurately, and classifying them into the most appropriate categories after analyzing their contents by using at least two predefined categories per given document features. In [22] Sini Shibu *et. al* proposed a unique method for categorization of web pages by applying feature selection technique along with page rank.

2.2 Neural Network

A neural network is a network comprising of processing elements called *neurons*. Neural networks can supplement the enormous processing power of the von Neumann digital computer with the ability to make sensible decisions and to learn by ordinary experience. They are used to solve categorization and many other problems. Neurons perform as summing and non linear mapping junctions. In some cases they can be considered as threshold units that fire when their total input exceeds certain bias levels. McCulloch and Pitts [24] outlined the first formal model of an elementary computing neuron.

2.2.1 McCulloch-Pitts Neuron Model

The McCulloch-Pitts neuron model is shown in figure 2.1[23]. The inputs x_i , for i = 1,2,…,n, are 0 or 1, depending on the absence or presence of the input impulse at instant k. The neuron's output signal is denoted as o. The firing rule for this model is defined as follows:

$$o^{k+1} = 1 \text{ if } \sum w_i x_i^k \geq T,$$

$$0 \text{ if } \sum w_i x_i^k < T$$

where superscript $k = 0,1,2,\ldots$ denotes the discrete-time instant, w_i is the multiplicative weight connecting the i'th input with the neuron's membrane and T is the neuron's threshold value , which needs to be exceeded by the weighted sum of signals for the neuron to fire.

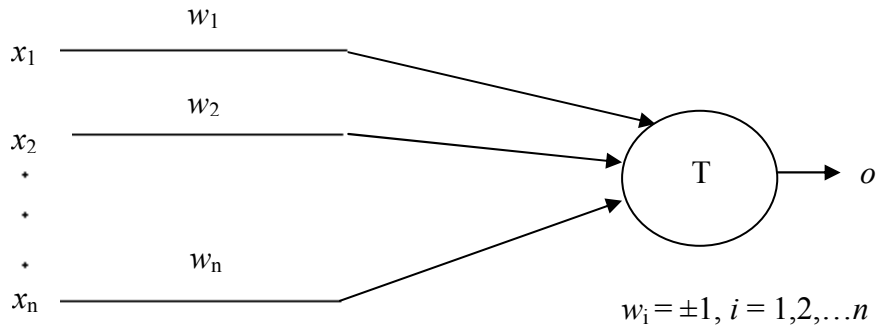


Figure 2.1: McCulloch-Pitts Neuron Model

2.2.2 General Neuron Model for Neural Networks

The McCulloch-Pitts model of a neuron is very simple. It allows binary 0, 1 states only, operates under a discrete time assumption, and assumes synchrony of operation of all neurons in a larger network. Weights and the neurons' thresholds are fixed in the model and no interaction among network neural takes place except for signal flow. Figure 2.2 depicts a general neuron model .

Every neuron model consists of a processing element with synaptic input connections and a single output. The signal flow of neuron inputs, x_i , is considered to be unidirectional as indicated by arrows, as is a neuron's output signal flow. Figure 2.2 shows a set of weights and the *neuron's processing unit,* or *node*.

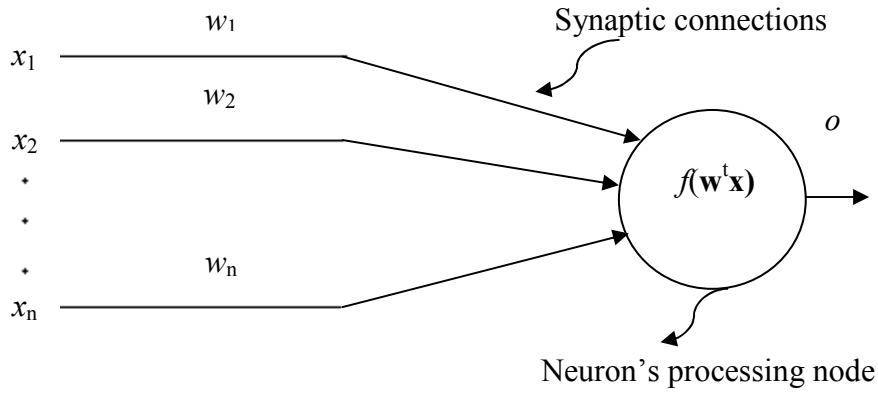


Figure 2.2: General Neuron Model

The neuron output signal is given by the following relationship:

$$o = f(\mathbf{w}^t\mathbf{x}), \text{ or}$$

$$o = f(\sum w_i x_i) \text{ where } i = 1,2\ldots n$$

where **w** is the *weight vector* defined as

$$\mathbf{w} = [w_1 \quad w_2 \quad \ldots \quad w_n]$$

and **x** is the *input vector*:

$$\mathbf{x} = [x_1 \quad x_2 \quad \ldots \quad x_n]$$

The function $f(\mathbf{w}^t\mathbf{x})$ is referred to as *activation function*. Its domain is the set of activation values, *net*, of the neuron model, hence this function can also be written as $f(net)$. The variable *net* is defined as a scalar product of the weight and input vector:

$$\text{net} = \mathbf{w}^t\mathbf{x}$$

2.2.3 Types of Activation Function

The typical activation functions used in different neural networks are:

$$f(net) = \{2/(1+\exp(-\lambda net))\} - 1 \qquad \textbf{(2.1)}$$

$$f(net) = \text{sgn}(net) = +1 \text{ if } net > 0 \text{ or } -1 \text{ if } net < 0 \quad \textbf{(2.2)}$$

where $\lambda > 0$ in (2.1) is proportional to the neuron gain determining the steepness of the continuous function $f(net)$ near $net = 0$. The activation function in (2.1) and (2.2) are called *bipolar continuous* and *bipolar binary functions*, respectively. The word "bipolar" is used to point out that both positive and negative responses of neurons are produced for this definition of the activation function.

Activation function (2.1) and (2.2) are also known as soft limiting activation function and hard limiting activation function respectively. Hard limiting activation function describes the discrete neuron model.

If the neuron's activation function has the bipolar binary form, it can be represented as in figure 2.3, which is actually a discrete neuron functional block diagram showing summation performed by the summing node and the hard limiting thresholding performed by the *threshold logic unit* (TLU). This model consists of the synaptic weights, a summing node, and the TLU element.

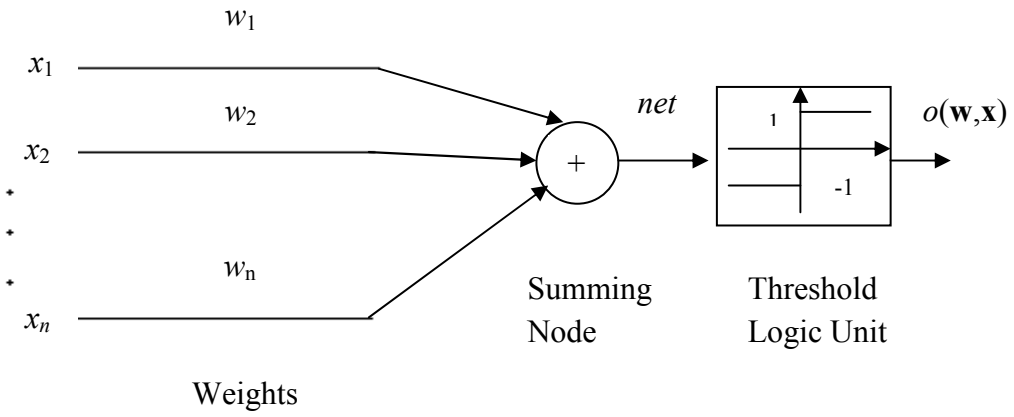


Figure 2.3: Hard Limiting Neuron (Binary Perceptron)

Hard limiting neuron can also be called as discrete or binary perceptron. The discrete perceptron, introduced by Rosenblatt [25], was the first learning machine. Continuous activation function can be represented by the neuron model shown in figure 2.4. Soft limiting neuron can also be called as continuous perceptron. The soft limiting activation

functions are often called *sigmoidal characteristics* as opposed to the *hard limiting activation functions.* In this thesis single discrete perceptron training algorithm is used with bipolar binary activation function. Single discrete perceptron training algorithm can be used for the problem of categorization and has been explained in section 2.5

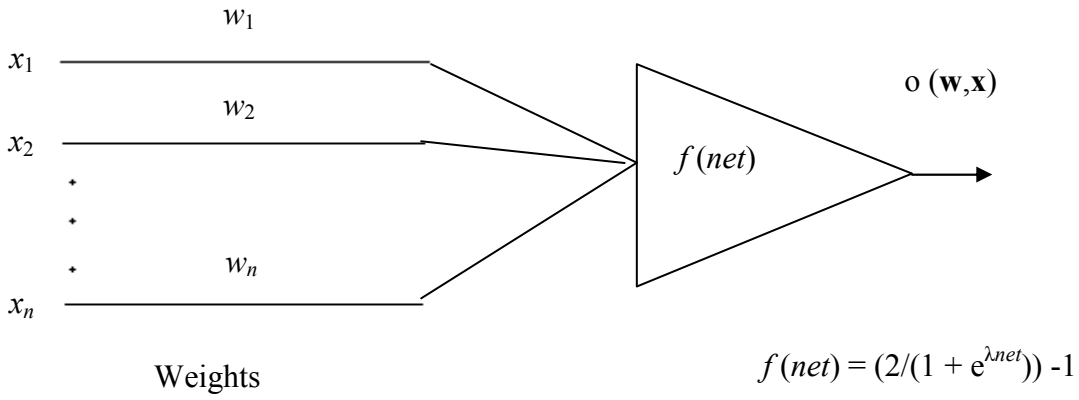


Figure 2.4: Soft Limiting Neuron (Continuous Perceptron)

2.3 Types of Neural Networks

Neural network can be defined as an interconnection of neurons such that neuron outputs are connected, through weights, to all other neurons including themselves; both lag- free and delay connections are allowed. There are basically two types of neural network as described below:

2.3.1 Feedforward Network

Feed forward network can be considered as a network consisting of *m* neurons receiving *n* inputs as shown in figure 2.5. Its output and input vectors are, respectively

$$\mathbf{o} = [o_1 \quad o_2 \quad \dots \quad o_m]^t$$

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]^t$$

Weight w_{ij} connects the i'th neuron with the j'th input. The double subscript convention used for weights is such that the first and second subscript denote the index of the destination and source nodes, respectively. The activation value for i'th neuron is

$$net_i = \sum w_{ij} x_j, \text{ for } i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n$$

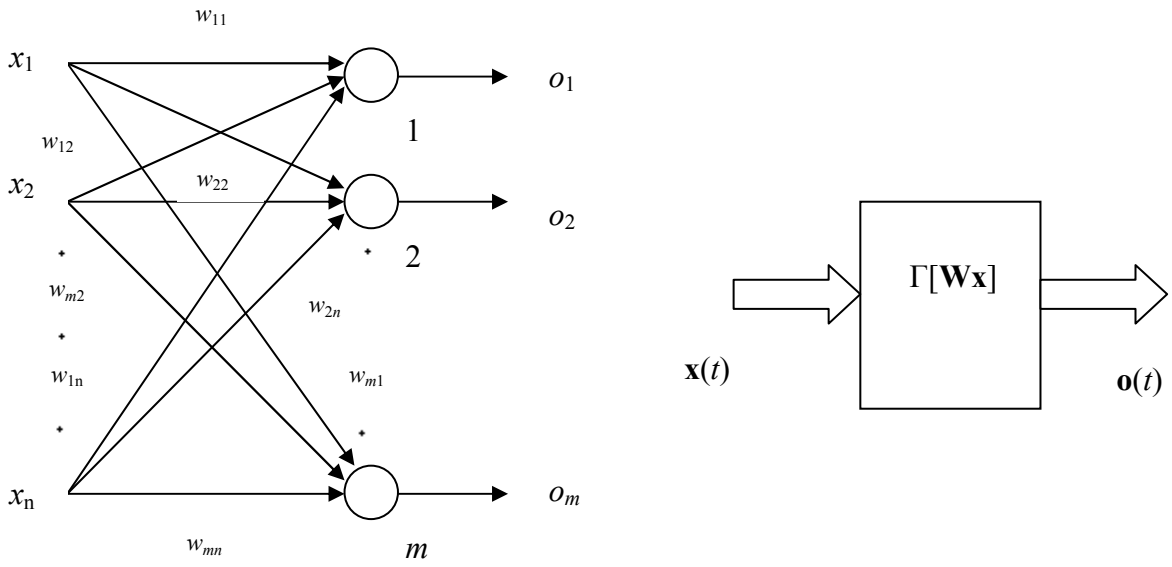


Figure 2.5: Single Layer Feedforward Network and Block Diagram

The transformation performed by each of the m neurons in the network, is a strongly nonlinear mapping expressed as

$$o_i = f(\mathbf{w}_i^t \mathbf{x}), \text{ for } i = 1, 2, \ldots, m$$

where weight vector $\mathbf{w}_i$ contains weights leading towards the i'th output node and is defined as follows

$$\mathbf{w}_i = [w_{i1} \quad w_{i2} \quad \ldots \quad w_{in}]^t$$

Introducing the nonlinear matrix operator Γ, the mapping of input space x to output space o implemented by the network expressed as follows

$$\mathbf{o} = \Gamma[\mathbf{W}\mathbf{x}]$$

where **W** is the *weight matrix*, also called the *connection matrix.* The input and output vectors **x** and **o** are often called *input* and *output patterns,* respectively. The mapping of an input pattern into an output pattern as shown in figure 2.5 is of the feed forward and instantaneous type, since it involves no time delay between the input **x**, and the output **o**. Thus it can also be represented as

$$\mathbf{o}(t) = \Gamma[\mathbf{W}\mathbf{x}(t)]$$

The generic feedforward network is characterized by the lack of feedback. This type of network can be connected in cascade to create a multilayer network. In such a network, the output of a layer is the input to the following layer. Even though the feedforward network has no explicit feedback connection when **x**(t) is mapped into **o**(t), the output values are often compared with the "teacher's" information, which provides the desired output value, and also an error signal can be employed for adapting the network's weights.

2.3.2 Feedback Network

The feedback network can be obtained from the feedforward network shown in figure 2.5 by connecting the neurons' outputs to their inputs. It is shown in figure 2.6.

The essence of closing the feedback loop is to enable control of output o_i through outputs o_j, for j = 1,2,…,m. Such control is especially meaningful if the present output, say **o**(t), controls the output at the following instant, **o**(t+Δ). The time Δ elapsed between t and t+Δ is introduced by the delay elements in the feedback loop as shown in figure 2.6. Using the notation introduced for feedforward networks, the mapping of **o**(t) into **o**(t+Δ) can be written as

$$\mathrm{o}(t+\Delta) = \Gamma[\mathbf{W}\mathbf{o}(t)]$$

This formula is represented by the block diagram shown in figure 2.6. The input **x**(t) is only needed to initialize this network so that **o**(0) = **x**(0). The input is then removed and the system remains autonomous for t > 0. Hence no input will be provided to the network for t > 0.

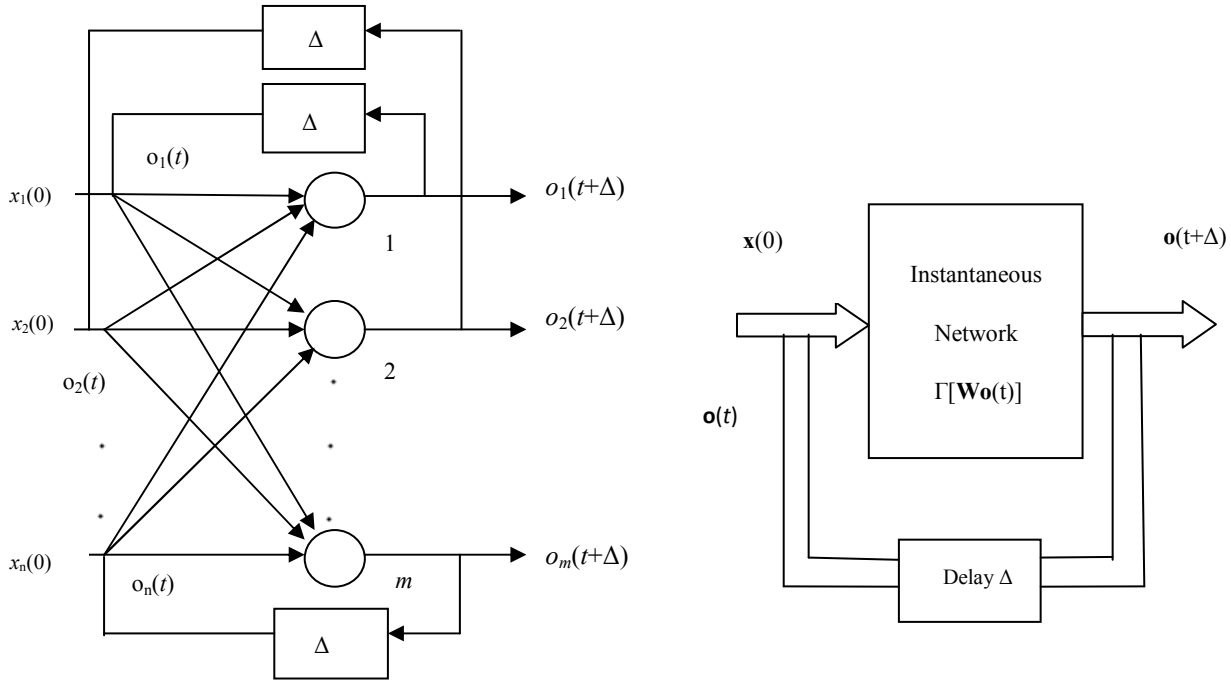


Figure 2.6: Single Layer Discrete Time Feedback Network and Block Diagram

There are two main categories of single layer feedback networks. If the time is considered as a discrete variable and decided to observe the network performance at discrete time instants Δ, 2Δ, 3Δ,…, the system is called *discrete-time*. For notational convenience, the time step in discrete-time networks is equated to unity, and the time instances are indexed by positive integers. Symbol Δ thus has the meaning of unity delay. For discrete time artificial neural system

$$\mathbf{o}^{k+1} = \Gamma[\mathbf{W}\mathbf{o}^{k}], \text{ for } k = 1,2,\ldots$$

where k is the instant number. The network in figure 2.6 is called *recurrent* since its response at the $k + 1$'th instant depends on the entire history of the network starting at $k = 0$. Therefore,

$$\mathbf{o}^{1} = \Gamma[\mathbf{W}\mathbf{x}^{0}]$$

$$\mathbf{o}^{1} = \Gamma\,[\mathbf{W}\Gamma[\mathbf{W}\mathbf{x}^{0}]$$

…

$$\mathbf{o}^{k+1} = \Gamma\,[\mathbf{W}\Gamma[….\Gamma\,[\mathbf{W}\mathbf{x}^{0}]….]]$$

Recurrent networks typically operate with a discrete representation of data; they employ neurons with a hard limiting activating function. A system with discrete time inputs and a discrete data representation is called automaton. Thus recurrent neural networks of this class can be considered as automations.

2.4 Neural Network Learning Modes

Learning is used for the experiential training of neural networks. It corresponds to parameter changes. Basically there are two modes of learning in case of neural networks:

2.4.1 Supervised Learning

In supervised learning, at each instant of time when the input is applied, the desired response **d** of the system is provided by the teacher. The distance μ[**d**,**o**] between the actual and the desired response serves as an error measure and is used to correct network parameters externally. In learning classifications of input patterns or situations with known responses, the error can be used to modify weights so that the error decreases. Since the weights are adjustable, the teacher may implement a reward and punishment scheme to adapt the network's weight matrix W.A set of input and output patterns called a training set is required for this learning mode. The block diagram for supervised learning mode is shown in figure 2.7

2.4.2 Unsupervised Learning

In learning without supervision, the desired response is not known; thus explicit error information cannot be used to improve network behavior. Since no information is available as to correctness or incorrectness of responses, learning must somehow be accomplished based on observations of responses to inputs of which we have marginal knowledge or no knowledge. Figure 2.8 shows the block diagram of unsupervised mode of learning.

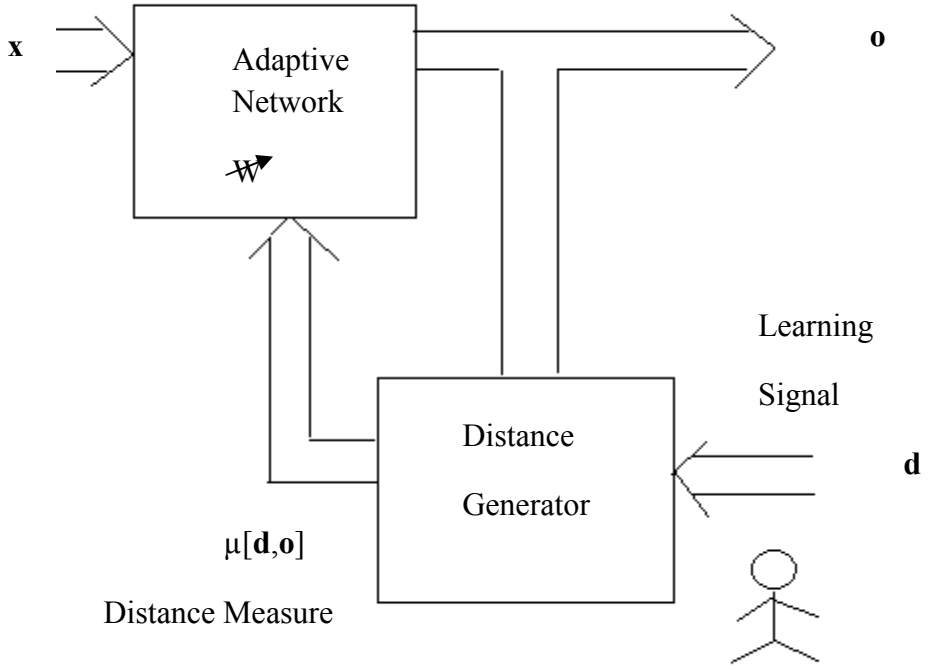


Figure 2.7: Supervised Learning Mode

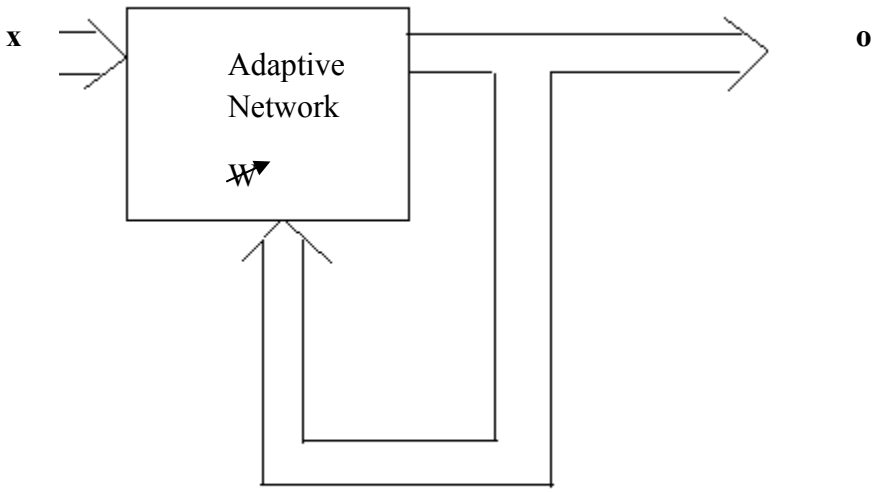


Figure 2.8: Unsupervised Learning Mode

2.5 Neural Network Learning Rules

A neuron is considered to be an adaptive element. Its weights are modifiable depending on the input signal it receives, its output value, and the associated teacher response. Under different learning rules, the form of the neuron's activation function may be different. The following *general learning rule* is adopted in neural network studies [26] :

The weight vector $\boldsymbol{w}_i = [w_{i1}\ w_{i2}\ \ldots\ w_{in}]^t$ *increases in proportion to the product of input* $\boldsymbol{x}$ *and learning signal r.*

The learning signal r is in general a function of $\mathbf{w}_i$, $\mathbf{x}$, and sometimes of the teacher's signal d_i. Therefore the learning signal for the network shown in figure 2.9 is

$$r = r(\mathbf{w}_i, \mathbf{x}, d_i)$$

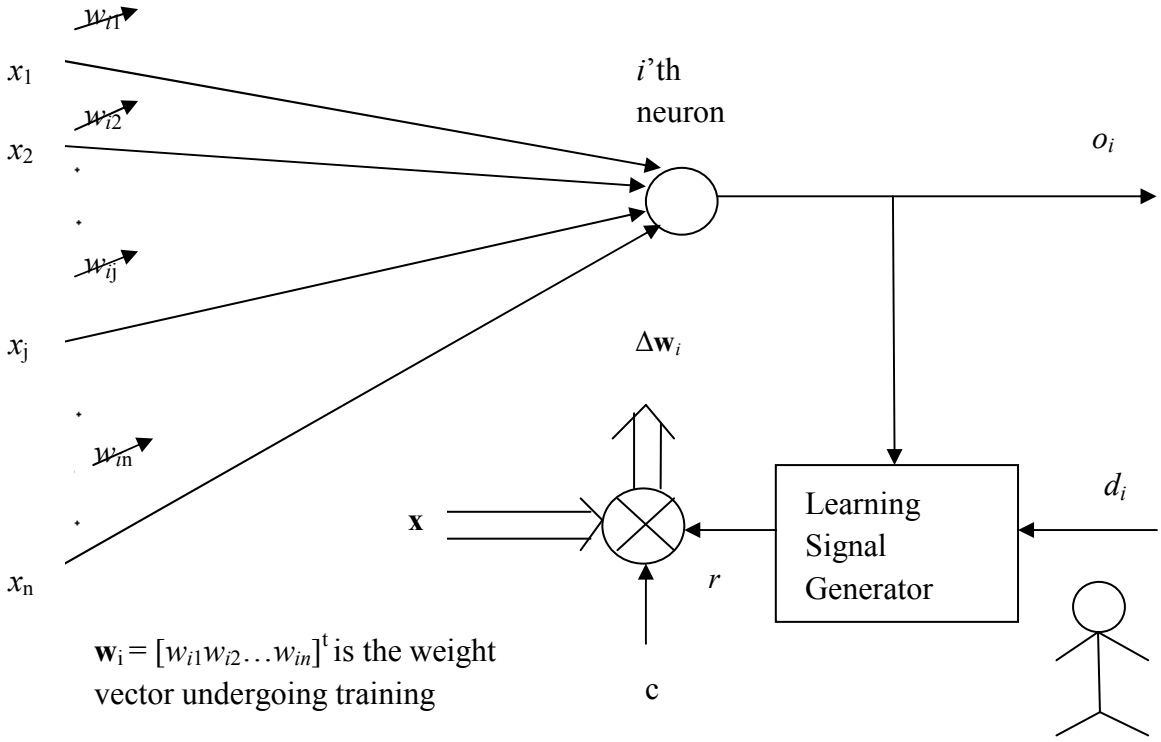


Figure 2.9 : Neural Network Learning

The increment of the weight vector $\mathbf{w}_i$ produced by the learning step at time t according to the general learning rule is

$$\Delta\mathbf{w}_i(t) = cr\,[\mathbf{w}_i(t), \mathbf{x}(t), \mathrm{d}_i(t)]\,\mathbf{x}(t)$$

where c is a positive number called the *learning constant* that determines the rate of learning. The weight vector adapted at time t becomes at the next instant or learning step,

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + cr\,[\mathbf{w}_i(t), \mathbf{x}(t), \mathrm{d}_i(t)]\,\mathbf{x}(t)$$

and for the k'th step,

$$\mathbf{w}_i^{k+1} = \mathbf{w}_i^{k} + cr\,(\mathbf{w}_i^{k}, \mathbf{x}^{k}, \mathrm{d}_i^{k})\,\mathbf{x}^{k}$$

The learning in the above equation assumes the form of a sequence of discrete-time weight modifications. Continuous time learning can be expressed as

$$\mathrm{dw}_i(t) / \mathrm{d}t = cr\mathbf{x}(t)$$

There are different kinds of learning rule. However in this thesis perceptron learning rule is applied to train the categorizer.

2.5.1 Perceptron Learning Rule

For the perceptron learning rule, the learning signal is the difference between the desired and actual neuron response [25]. Thus learning is supervised and the learning signal is equal to

$$r = d_i - o_i$$

where $o_i = \mathrm{sgn}(\mathbf{w}_i^t\mathbf{x})$, and d_i is the desired response as shown in figure 2.9. Weight adjustments in this method, $\Delta\mathbf{w}_i$ is

$$\Delta\mathbf{w}_i = c\,[d_i - \mathrm{sgn}(\mathbf{w}_i^t\mathbf{x})]\mathbf{x}$$

This rule is applicable only for binary neuron response and the above equation express the rule for bipolar binary case [27]. Under this rule, weights are adjusted if and only if o_i is incorrect. Error as a necessary condition of learning is inherently included in this

training rule. Since the desired response is either 1 or -1, the weight adjustment reduces to

$$\Delta \mathbf{w}_i = \pm 2c\mathbf{x}$$

Where a plus sign is applicable when $d_i = 1$, and $\text{sgn}(\mathbf{w}_i^t\mathbf{x}) = -1$, and a minus sign is applicable when $d_i = -1$, and $\text{sgn}(\mathbf{w}_i^t\mathbf{x}) = 1$. There will be no modifications in weight if $d_i = \text{sgn}(\mathbf{w}_i^t\mathbf{x})$. Perceptron learning rule is of central importance for supervised learning of neural networks. The weights can be initialized at any values in this method.

2.6 Single Discrete Perceptron Training Algorithm

Single Discrete Perceptron Training Algorithm can be implemented to perform the task of binary categorization. The neural network is needed to be trained first with the help of perceptron learning rule. Following are the given steps for the algorithm:

Given are P training pairs

$\{\mathbf{x}_1, d_1; \mathbf{x}_2, d_2; \mathbf{x}_3, d_3; \ldots\ldots; \mathbf{x}_p, d_p\}$, where $\mathbf{x}_i$ is $(n * 1)$, d_i is $(1 * 1)$, $i = 1, 2, 3, \ldots, P$

In this algorithm augmented input vectors will be used:

$$\mathbf{y}_i = [\mathbf{x}_i \quad 1]^t \text{, for } i = 1, 2, 3, \ldots\ldots, P$$

In the following, k denotes the training step and p denotes the step counter within the training cycle.

Step 1: Choose $c > 0$

Step 2: Initialize weight $\mathbf{w}$ at small random values, $\mathbf{w}$ is $(n+1) * 1$. Initialize counters and error.

$$k \leftarrow 1, p \leftarrow 1, E \leftarrow 0$$

Step 3: The training cycle begins here. Input is presented and output computed:

$$y = y_p, d \leftarrow d_p$$

$$o \leftarrow \text{sgn}(\mathbf{w}^t\mathbf{y})$$

Step 4: Weights are updated:

$$\mathbf{w} \leftarrow \mathbf{w} + 0.5c(d - o)\mathbf{y}$$

Step 5: Cycle error is computed:

$$E \leftarrow 0.5(d - o)^2 + E$$

Step 6: If $p < P$ then $p \leftarrow p + 1$, $k \leftarrow k + 1$, and go to step 3; otherwise go to step 7.

Step 7: The training cycle is completed. For E = 0, terminate the training session. Output weights and k.

If $E > 0$, then $E \leftarrow 0$, $p \leftarrow 1$, and enter the new training cycle by going to step 3.

2.6.1 Perceptron Convergence Theorem

Perceptron Convergence Theorem, states that a categorizer for two linearly separable classes of patterns is always trainable in a finite number of training steps. Therefore,

$$\mathrm{w}^* = \mathrm{w}^k_o = \mathrm{w}^{k+1}_o = \mathrm{w}^{k+2}_o = \ldots$$

where w^* is the solution vector. The integer k_o is the training step number starting at which no more misclassification occurs and thus no weight adjustments take place for $k_o \geq 0$.

Chapter-3

Problem Statement

3.1 Problem Definition

Web page categorization is one of the challenging tasks due to ever increasing traffic of web pages. A number of researches have been done in this field using different approaches and techniques as described in chapter 1 and 2. Each one of them has some limitations. Web pages are connected to each other by hyperlinks. Feature extraction is considered as the most important task of web page categorization and also the difficult one due to semi–structured source code and hyperlinked structure of web pages. Features can be divided into two: on page features and neighboring features. On page features are the features which can be directly extracted from the web page through textual content, visual content and various HTML tags present in web pages. Neighboring features are the features that can be extracted from the web pages which are connected to web page which is needed to be categorized. But it is very difficult to extract these features. Most of the algorithms rely only on the text content of the web pages and also difficult to implement. However besides text, each type of web has its own layout. The characteristics of web pages can also be used to categorize web pages. Thus the problem is to implement the technique which can categorize the web pages based on some characteristics of web pages which is easy to understand and use.

3.2 Proposed Objective

The main objectives that are addressed in the thesis to solve the above mentioned problem are as follows:

- To study and analyze different features of source web pages and select those features on the basis of which web pages can be categorized.

- To build a binary categorizer and train it with input values which consist of features extracted from web pages.
- To test the binary categorizer by comparing actual output and the desired output.
- To verify and analyze the result in support of this proposal.

3.3 Methodology Used

- Collection of data set.
- Study and analyzing of the dataset.
- Selection and extraction of features from the data set.
- Implementation and training of the algorithm.
- Verification and analyzing the categorizer using test data set.
- Performance Evaluation.

Chapter-4

Implementation

In this chapter various steps have been explained to implement the categorizer to categorize the newspaper web pages and education web pages.

The proposed approach is explained in the following steps:

4.1 Data Set Collection

First of all data is collected. Data set consists of education and newspaper web pages. These web pages are collected from different sites and also from Yahoo! [3] web directory.

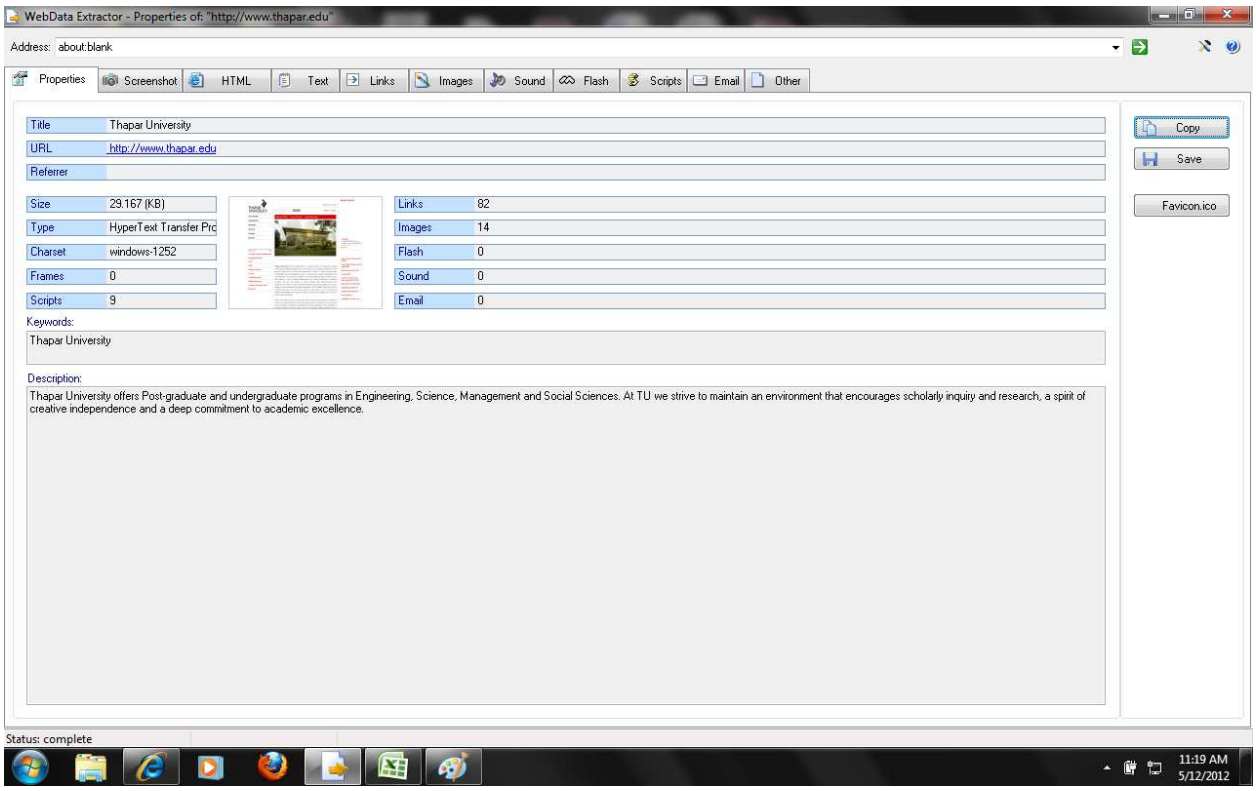
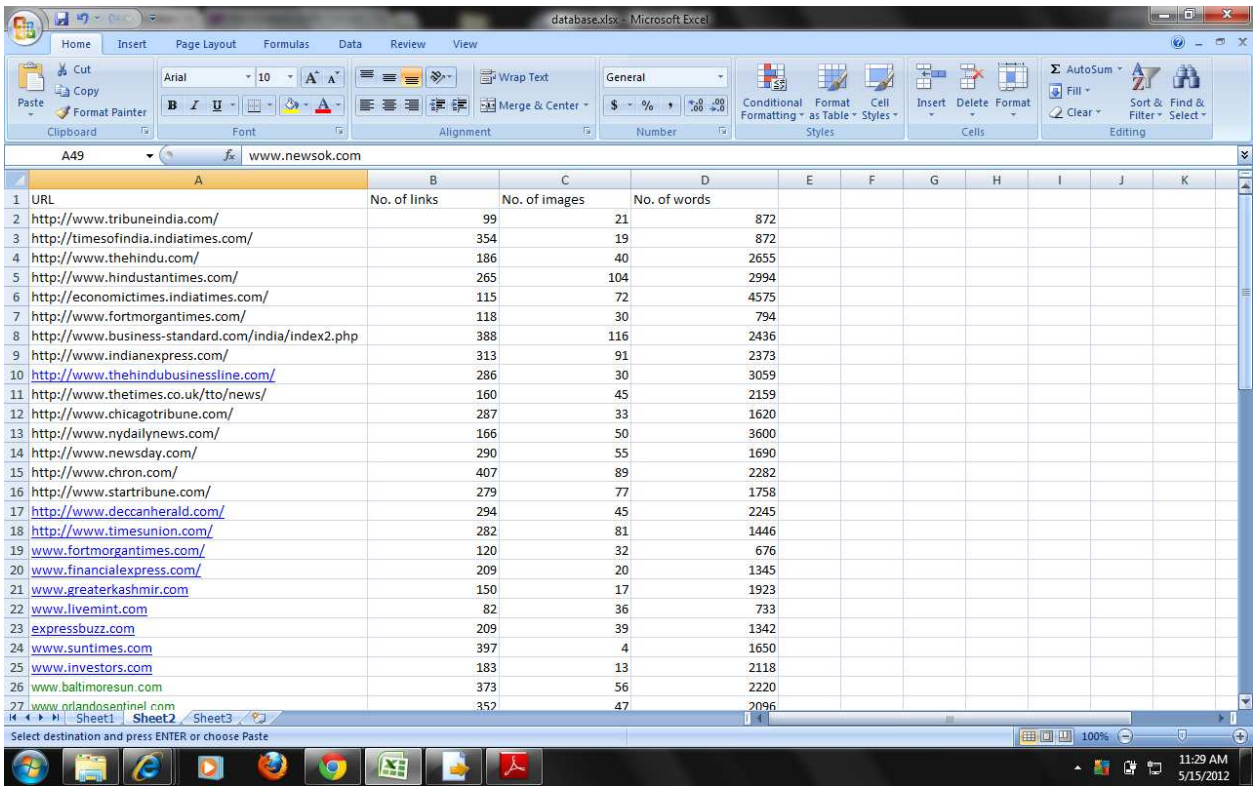


Figure 4.1: Data Set Collection

4.2 Feature Extraction

After collecting the data set, features of the web pages in data set are extracted automatically. The main features which are extracted are number of links, number of images and amount of text present on the web pages. After analyzing these features it has been found that the newspaper web pages contain more number of links, images and words as compared to education web pages. It helps in differentiating these two types of web pages. Figure 4.1 shows the values of extracted features along with the URL's of web pages.



	A	B	C	D
1	URL	No. of links	No. of images	No. of words
2	http://www.tribuneindia.com/	99	21	872
3	http://timesofindia.indiatimes.com/	354	19	872
4	http://www.thehindu.com/	186	40	2655
5	http://www.hindustantimes.com/	265	104	2994
6	http://economictimes.indiatimes.com/	115	72	4575
7	http://www.fortmorgantimes.com/	118	30	794
8	http://www.business-standard.com/india/index2.php	388	116	2436
9	http://www.indianexpress.com/	313	91	2373
10	http://www.thehindubusinessline.com/	286	30	3059
11	http://www.thetimes.co.uk/tto/news/	160	45	2159
12	http://www.chicagotribune.com/	287	33	1620
13	http://www.nydailynews.com/	166	50	3600
14	http://www.newsday.com/	290	55	1690
15	http://www.chron.com/	407	89	2282
16	http://www.startribune.com/	279	77	1758
17	http://www.deccanherald.com/	294	45	2245
18	http://www.timesunion.com/	282	81	1446
19	www.fortmorgantimes.com/	120	32	676
20	www.financialexpress.com/	209	20	1345
21	www.greaterkashmir.com	150	17	1923
22	www.livemint.com	82	36	733
23	expressbuzz.com	209	39	1342
24	www.suntimes.com	397	4	1650
25	www.investors.com	183	13	2118
26	www.baltimoresun.com	373	56	2220
27	www.orlandosentinel.com	352	47	2096

Figure 4.2: Database of Web Pages with Values of Extracted Features

After analyzing the values obtained for different extracted features, mean and standard deviation is calculated and each value is mapped to the value in the range [-2,2] as shown below:

No. of Images	Input Value
1-30	-2
31-60	-1
61-90	0
91-120	1
121-150	2

Table 4.1: Input Values for Number of Images

No. of Links	Input Value
1-100	-2
101-200	-1
201-300	0
301-400	1
401-500	2

Table 4.2: Input Values for Number of Links

No. of Words	Input Value
1-1000	-2
1000-2000	-1
2000-3000	0
3000-4000	1
4000-5000	2

Table 4.3: Input Values for Number of Words

4.3 Implementation and Training of Algorithm

The discrete perceptron training algorithm is used in the proposed approach. It is based on the concept of neural networks and has been described in chapter 2. The implementation of the algorithm is done in TurboC2 using object oriented programming language C++. The pseudo code is listed in appendix.

The platform used is Intel 64-bit with Core 2 Duo processor having a frequency of 2.0 GHz with Windows 7 64-bit Enterprise Edition running on it. The system had a RAM of 2.0 GB.

The first step after implementation to build a categorizer is to train it with the collected data set. The single perceptron training algorithm is based on perceptron learning rule. The weight vector used to train the categorizer is needed to be initialized to any values as shown below:

$$\mathbf{w} = [0.4 \quad -2.5 \quad -0.75 \quad 2.0]$$

After initializing the weight vector, training data is provided in the form of augmented input vectors as explained in algorithm described in chapter 2. Sample training input vectors are

$$\mathbf{x_1} = [1 \quad 1 \quad 0 \quad 1]$$

$$\mathbf{x_2} = [2 \quad 0 \quad 0 \quad 1]$$

$$\mathbf{x_3} = [-2 \quad -2 \quad -2 \quad 1]$$

$$\mathbf{x_4} = [-2 \quad -1 \quad -2 \quad 1]$$

Desired output values for $\mathbf{x_1}$, $\mathbf{x_2}$ is 1 and for $\mathbf{x_3}$, $\mathbf{x_4}$ is -1. Training will get complete in a finite number of steps according to Perceptron Convergence Theorem described in chapter 2. Weights will keep on modifying unless the desired output became equal to the actual output. After that there will be no modifications in the weight vector. Thus the final weights obtained after training are:

$$\mathbf{w_f} = [3.4 \quad -0.5 \quad 1.25 \quad 2.0]$$

Figure 4.2 shows the output for training phase of the algorithm and figure 4.3 shows the final weights obtained after training.

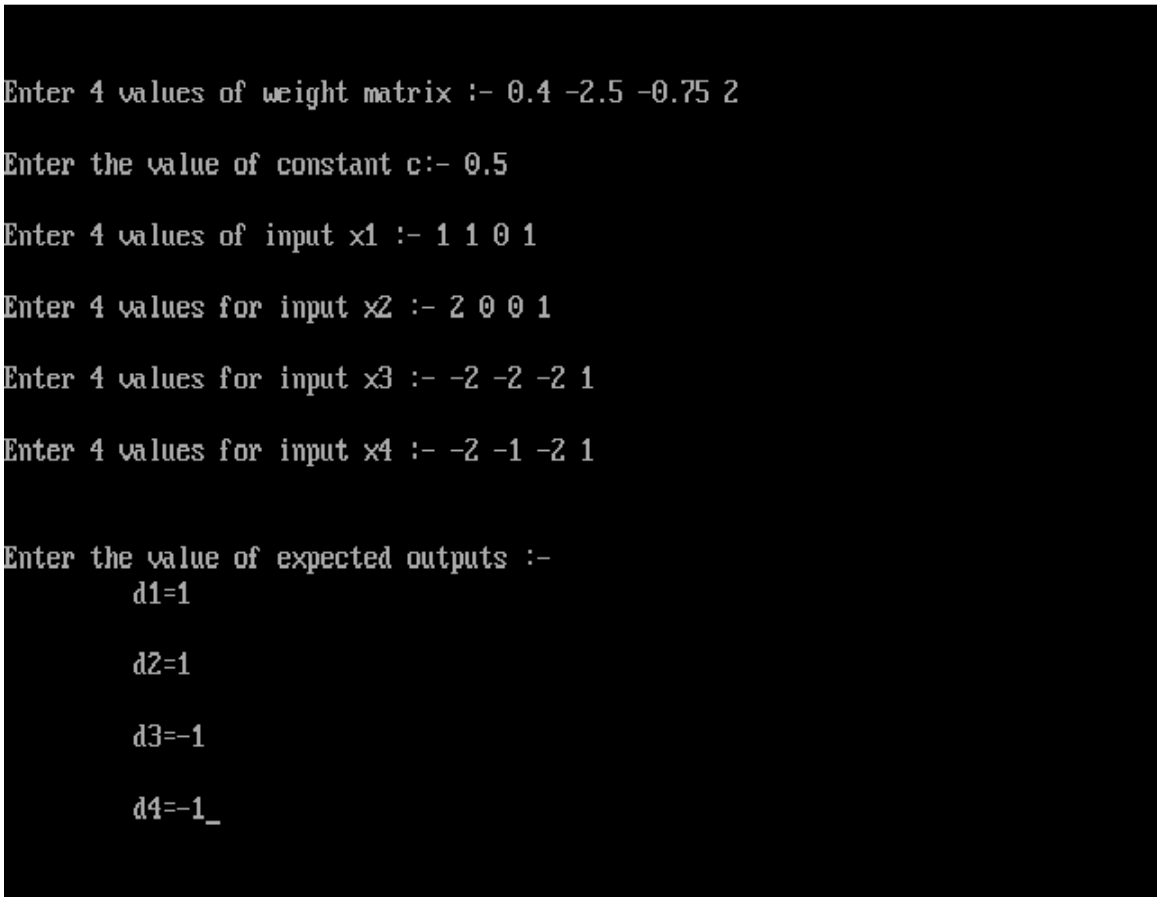


Figure 4.3 Training of the Algorithm

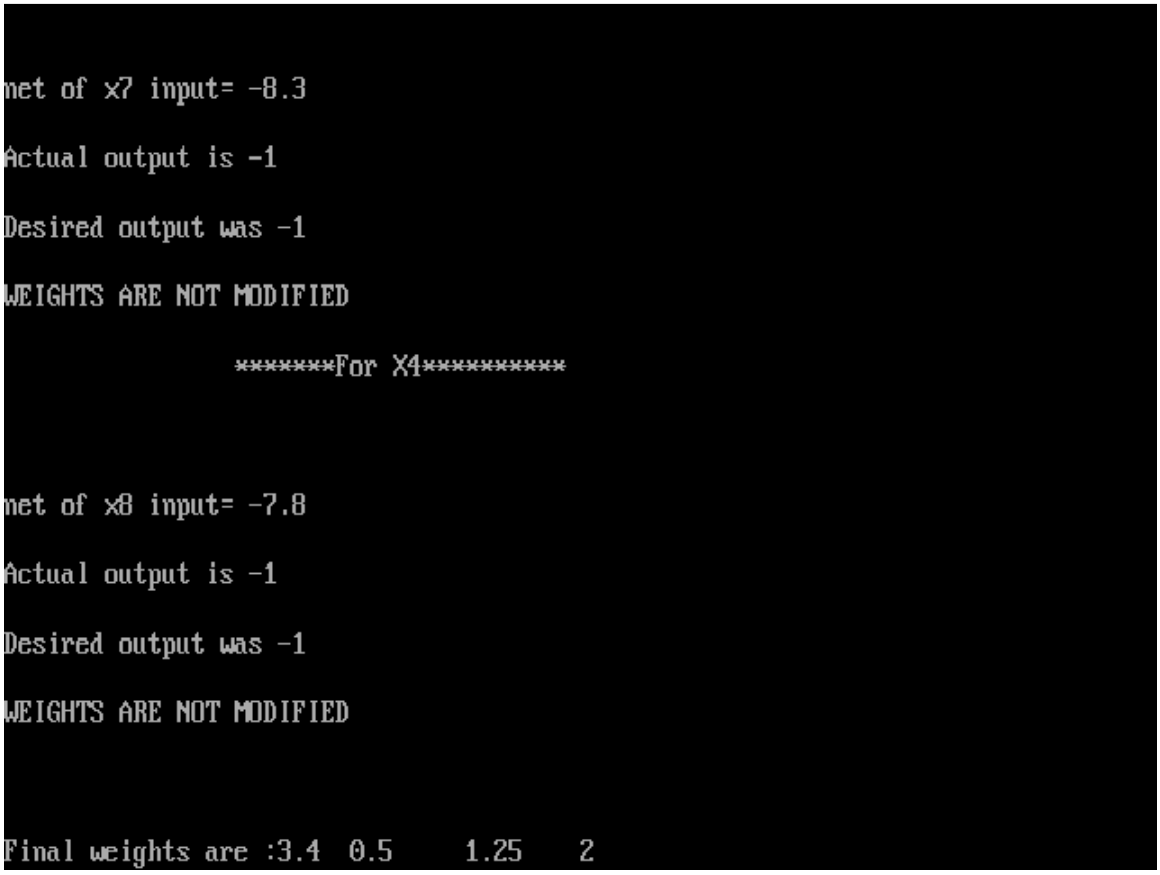


Figure 4.3: Final Weights after Training

4.4 Categorization of web pages

Once the weights are fixed, training will get completed. Testing data set can be applied to the program to categorize the web pages. 120 web pages are used to test the categorizer.

Chapter-5

Testing and Results

5.1 Testing

Testing of the system is done with the help of a good number of training examples. The training data set should reflect the real world situation. The true performance of any system can be evaluated on the basis of high quality training data set. Therefore training data set is collected from 120 home pages of different education websites and newspaper websites. The data obtained from different web pages is applied to the categorizer in the form of input vector whose value lies in the range of [-2, 2]. Testing results of the categorizer is shown in figure 5.1 and 5.2

5.2 Results

Out of 120 source web pages 110 web pages are categorized correctly. Accuracy of the results can be measured in terms of precision which can be defined as the number of correct categories assigned divided by the total number of categories assigned. The experimental or testing results are shown in table 5.1 and 5.2 along with accuracy.

Total Pages	60
Right Categorized Pages	58
Wrong Categorized Pages	2
Accuracy	96%

Table 5.1: Experimental Results for Education Web Pages

Total Pages	60
Right Categorized Pages	52
Wrong Categorized Pages	8
Accuracy	86.66%

Table 5.2: Experimental Results for Newspaper Web Pages

Hence the average accuracy obtained in the results is 91.33 percent which is very high. Figure 5.1 and 5.2 are depicting net values calculated corresponding to test input vectors along with their categories.

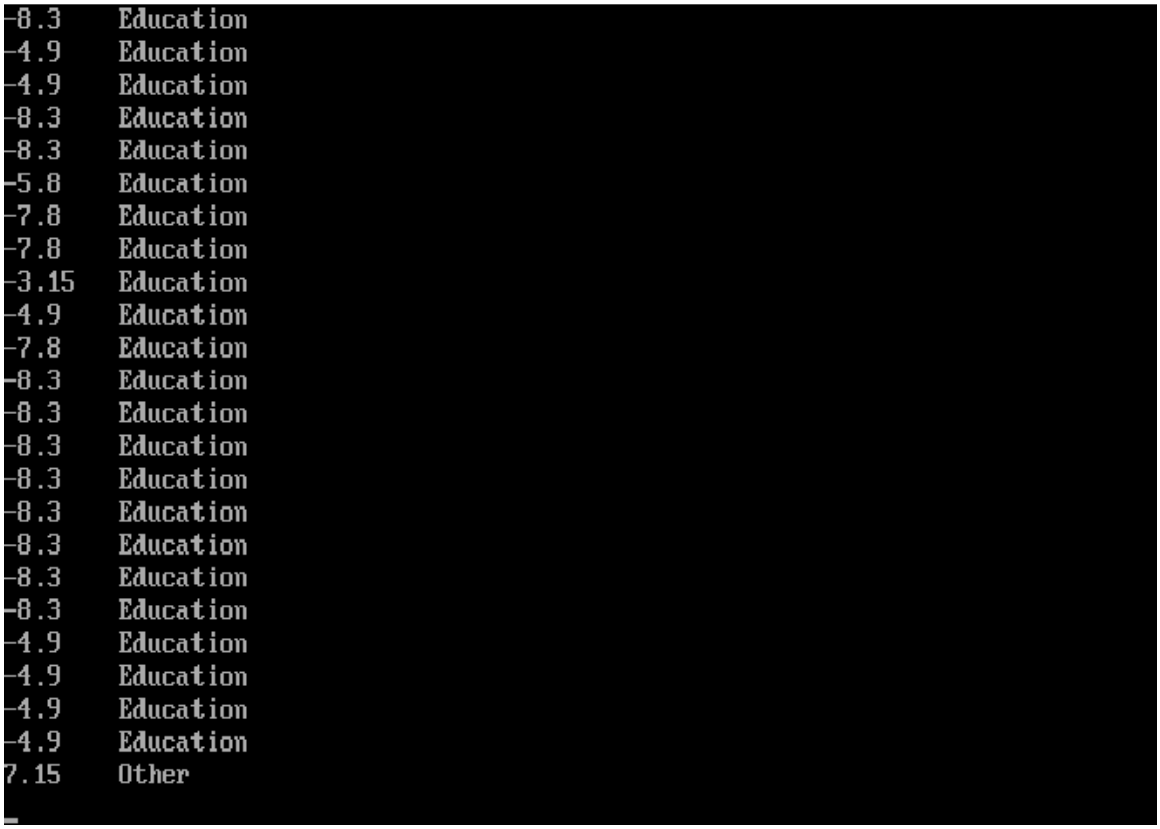


Figure 5.1: Categorization Output for Education

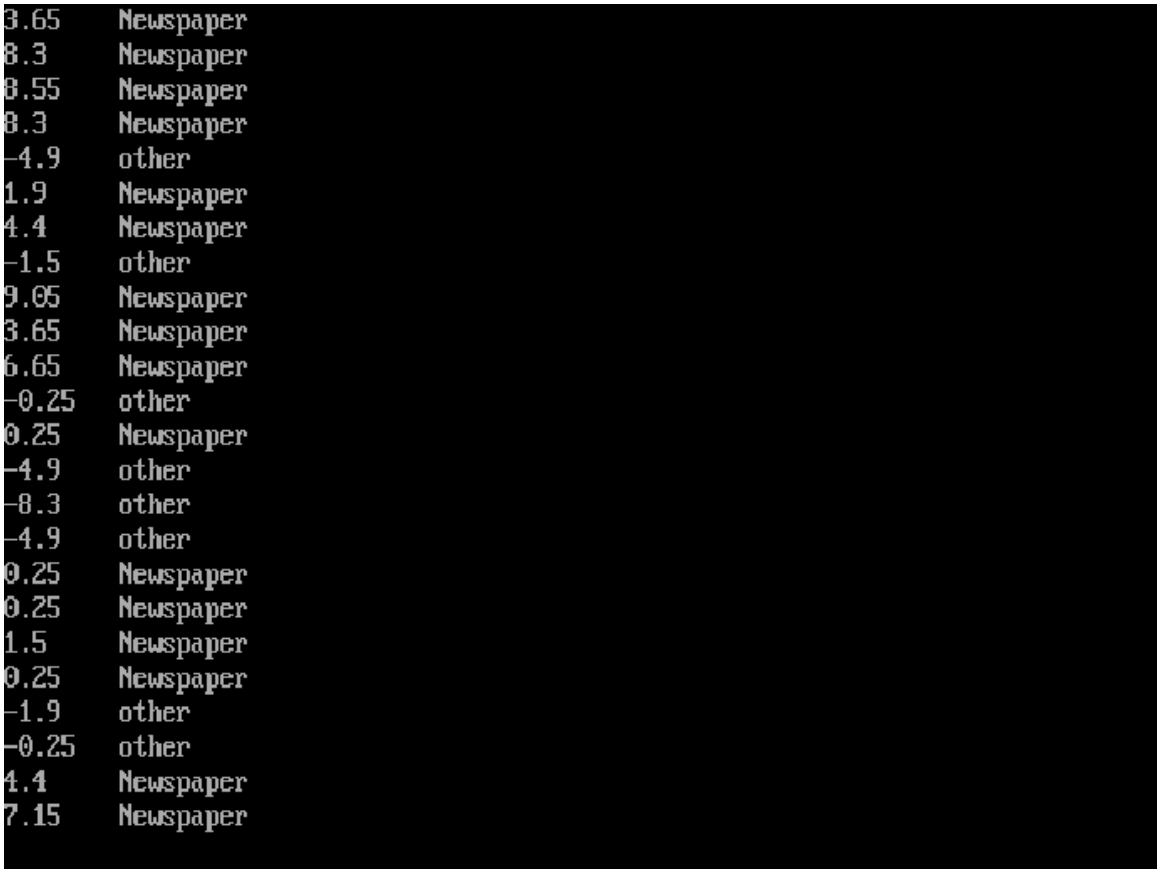


Figure 5.2: Categorization Output for Newspaper

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Web page categorization is considered as a fundamental problem these days due to uncontrolled nature of web. It is essential for web mining, developing directories and efficient data retrieval tasks. There are different kinds of categorization schemes. Now a day, numbers of applications are also growing. Each application demands different scheme of categorization. A number of researches and approaches proposed by different researchers have been outlined in the thesis. Most of these approaches are based on the concept of text content of web pages. Besides text there is a lot of multimedia content present in the web pages which can also be utilized to categorize them with high accuracy.

This thesis has presented an idea to categorize web pages with a new approach that is to categorize the web pages on the basis of their characteristics like number of links, number of images and amount of text present on them, which is very easy to understand, implement and use as compared to other approaches in the field of web page categorization. In support of this proposal the above mentioned characteristics are extracted from newspaper and education web pages. It has been found that newspaper web pages have more number of links, images and words than education web pages. This difference helped in differentiating between the two categories. The binary categorizer built for the categorization of web pages is based on the concept of neural networks. Neural networks can be used as categorizers. The algorithm used is single discrete perceptron training algorithm. It is implemented and trained with finite set of input data vectors. After training of the algorithm, final weights are obtained which can't be modified later by any number of input data vectors. Testing is performed with 120 home pages of different newspaper and education web sites and it is found that the results

obtained from this approach are 91.33 percent accurate. It can also be used to categorize web pages into broad categories. For example in order to classify blog and non blog sites, extract those features which can distinguish between the two categories. In blog sites one can find a lot of text in the form of articles, comments with lots of emoticons and also number of links. Such features can be extracted and used for categorization. Similarly one can distinguish between the research pages and content pages by analyzing the characteristics of web pages. Social networking sites and non social networking sites can also be categorized by analyzing the features which can distinguish between their characteristics.

6.2 Future Scope

The approach which is used to categorize web pages is limited only to extraction of a few features. Also, it is implemented for two specific categories only. In this approach visual information of the web pages as rendered by the web browser and also the neighbor pages have not been taken into consideration. Hence the visual information features like placement of links, area and size of images etc as well as the use of categories of neighboring web pages can play an important role in categorizing web pages into broad categories with high accuracy.

References

[1] Pierre J. M., "Practical Issues for Automated Categorization of Web Pages," September 2000.

[2] Xiaoguang Q. and Davison B. D., "Web page classification: Features and algorithms," ACM Computing Surveys, 41(2), 2009

[3] Yahoo!, http://www.yahoo.com, Accessed date 14th March, 2012.

[4] Open Directory Project, http://www.dmoz.org, Accessed date 15th March, 2012

[5] Xu Z. *et. al.*, "A Web Page Classification Algorithm Based On Link Information," in DCABES'11 Proceedings of the Tenth International Symposium on Distributed Computing and Applications to Business, Engineering and Science , pp. 82-86, 2011.

[6] Bartik V., "Text-Based Web Page Classification with Use of Visual Information," in ASONAM'10 Proceedings of the International Conference on Advances in Social Network Analysis and Mining, pp. 416-420, 2010.

[7] He Z. and Liu Z., "A Novel Approach to Naïve Bayes Web Page Automatic Classification," in FSKD'08 Proceedings of the Fifth International Conference on Fuzzy System and Knowledge Discovery, pp. 361-365, 2008.

[8] Radovanović M. and Ivanović M.,"Document Representation for Classification of Short Web Page Descriptions," in Yugoslav Journal of Operations Research, I8, Number 1, pp. 123-138, 2008.

[9] Dai W. *et. al.*, "A Novel Web Page Categorization Algorithm Based on Block Propagation Using Query-Log Information," in WAIM'06, LNCS 4016, pp. 435-446, 2006.

[10] Materna J., "Automatic Web Page Classification," in RASLAN'08 Proceedings of Recent Advances in Slavonic Natural Language Processing, pp. 84-93, 2008.

[11] Kwon O. and Lee J., "Web page classification based Nearest Neighbor approach," in IRAL'00 Proceedings of the fifth international workshop on Information retrieval with Asian languages, pp. 9-15, 2000.

[12] McCallum A. and Nigam K., "A Comparison of Event Models for Naive Bayes Text Classification," in AAAI-98 Workshop on Learning for Text Categorization, 1998.

[13] Koller D. and Sahami M., "Hierarchically classifying documents using very few words," in ICML'97 Proceedings of the Fourteenth International Conference on Machine Learning, pp.170-178, 1997.

[14] Lewis D. and Ringuette M., "A Classification of two learning algorithms for text categorization," in SDAIR'94 Third Annual Symposium on Document analysis and Information Retrieval, pp.81-93, 1994.

[15] Apte C. and Damerau F., "Automated Learning of Decision rules for Text categorization," ACM Transactions on Information Systems, Vol 12, No.3, pp.233-251, 1994.

[16] Weigend A. S., Weiner E. D. and Peterson J. O., "Exploiting Hierarchy on Text categorization," Information Retrieval, I(3), pp. 193-216, 1999.

[17] Dumais S. T., Platt J., Heckerman D. and Sahami M., "Inductive Learning Algorithms and representations for text categorization," in CIKM'98 Proceedings of the Seventh International conference on Information and Knowledge Management, pp. 148-155, 1998.

[18] Shen D. *et. al.*, "Web Page Classification through summarization," in SIGIR'04 Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 242-249, 2004.

[19] Kan M., "Web Page Categorization without the Web Page," WWW Alt.'04 Proceedings of the 13th international World Wide Web conference on Alternate track papers and posters, pp. 262-263, 2004.

[20] Dumais S., Chen H., ”Hierarchical Classification of Web Content,” in SIGIR’00 Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in informational retrieval, pp. 256-263, 2000.

[21] Song M. et. al., ”Automatic Classification of Web Pages based on the Concept of Domain Ontology,” in Software Engineering Conference, APSEC’05. 12th Asia Pacific, 2005.

[22] Shibu S., Vishwakarma A., Bhargava N., ”A combination approach for Web Page Classification using Page Rank and Feature Selection Technique,” in International Journal of Computer Theory and Engineering, Vol.2, No.6, December, 2010.

[23] Zurada J. M.,”Introduction to Artificial Neural Systems”. Chapter 2, pp. 30-66.

[24] McCulloch W. S. and Pitts W. H., “A Logical Calculus of the Ideas Imminent in Nervous Activity,” *Bull. Math. Biophy.* 5:115-133, 1943.

[25] Rosenblatt F., “The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain,” Psych. Rev. 65: 386-408, 1958

[26] Amari S. I., “Mathematical Foundations of Neurocomputing,” IEEE Proc. 78(9): 1443-1463, 1990.

[27] Zurada J. M., ”Introduction to Artificial Neural Systems”. Chapter 3, pp. 93-132.

Appendix

Pseudo code for Single Discrete Perceptron Training Algorithm.

1. Initialize the constant num to 4

2. Input initial weight vector wt[num]

3. Input value of learning constant c

4. Input values of input vectors X1, X2, X3, X4 and store these in two dimensional array arr[4][num]

5. Input desired output values d1, d2, d3 and d4 for input vectors X1, X2, X3 and X4 respectively

6. Initialize actual output variables sgn, sgn1, sgn2 and sgn3 to zero for input vectors X1, X2, X3 and X4 respectively

7. while the user has not as yet entered the sentinel

8. set sgn, sgn1, sgn2 and sgn3 to zero

9. set net values net[0], net[1], net[2], net[3] to zero for each input vector

10. set variable i to zero

11. for i less than num

12. net[0] += wt[i] * arr[0][i]

13. increment i

14. if net[0] is greater than zero

15. set sgn to 1

16. else

17. set sgn to -1

18. endif

19. If actual output is not equal to desired output

20. update weight vector as up_wt[i] = wt[i] + ((c * (d1 – sgn)) * arr[0][i])

21. print updated weight vector

22. else

23. print weight vector is not modified

24 endif

25. Repeat steps 8-24 for input vectors X2, X3 and X4

26. If sgn is equal to d1 and sgn1 is equal to d2 and sgn2 is equal to d3 and sgn3 is equal to d4

27. print final weight vector and exit from while loop

28. endif

29. endwhile

List of Publications

1. K. Taneja, and V. K. Bhalla, "Web Page Categorization: Tools, Techniques and Evaluation" International Journal of Advances in Computing and Information Technology, Vol. 1, Issue-2, pp. 212-219, 2012 (Published).

2. K. Taneja, and V. K. Bhalla, "Web Page Categorization based on Characteristics of Web Page", International Conference on Electrical Engineering and Computer Science (ICEECS-2012), (Accepted).