Expert Systems with Applications 37 (2010) 6157–6164

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Diagnosis of liver disease by using CMAC neural network approach

İhsan Ömür Bucak *, Semra Baki

Computer Engineering Department, Engineering Faculty, Fatih University, 34500 Istanbul, Turkey

ARTICLE INFO

Keywords:
Liver disease
Liver cirrhosis
Child's classification
Liver enzymes
CMAC neural networks
Medical diagnosis
Clinical laboratory values

ABSTRACT

Liver performs several numbers of metabolic functions that are essential to human life. These functions make the liver one of the most important organs in the human body. There are diseases that occur in the liver in short time (acute) and long time (chronic). These diseases could occur because of medications, alcohol, viruses, or excessive fat accumulation or deposit in the liver. Some of these diseases are the inflammation of the liver, insufficient liver performance, Hepatitis A, B, C, D, and liver cirrhosis. If the liver malfunctions in anyway, people know that they are putting their life at risk. For this reason, diagnosing any disease in the liver is important and sometimes difficult. It is also important to notice the diagnosis of the patient at an early stage as the symptoms arise so that the patient might be able to carry on a normal life. The objective of this article is to diagnose the liver disease using an application of the CMAC (Cerebellar Model Articulation Controller) neural network so that it can shorten the medical diagnostic process and help the physician in the complex cases which are otherwise difficult to perceive.

1. Introduction

Today, early diagnosis and correct assessment of many diseases tend to have great importance in disease treatment. Therefore, diagnostic and classification process of a disease to be made by using today's technology and medical data would present many uses. In this paper, we have considered the diagnosis of liver diseases which are Hepatitis B, Hepatitis C, Cirrhosis and the cirrhotic phases.

Liver is a vital part of our body (Hopkins, 2008). If the liver does not perform any of its vital missions such as production of bile, regulation of blood levels of amino acids, and production of certain proteins for blood plasma, human being would face a serious health consequences (Darwin, 2008). Therefore, early diagnosis of the liver disease is very important.

Diagnosing the liver disease can be a difficult task, because symptoms do not often appear until the later stages of most liver diseases and conditions. By then the liver may have suffered serious or permanent damage (DuPage, 2007). When diagnosing the liver disease, the physician looks at the patient's symptoms first such as irregular sleep, jaundice, and portal hypertension, and conducts a physical examination. In addition, the physician may request a liver biopsy, liver function tests (AST, ALT, biluribin), an ultrasound, or a computerized tomography (CT) scan (Darwin, 2008). Liver function tests can diagnose viral hepatitis and autoimmune liver diseases. An ultrasound scan will show blockage of the bile duct, fatty liver, cirrhosis, and liver tumors (Babe, 2007).

The physician may spend very long time for the assessment of the enzyme values during normal diagnostic period while making a decision based on those enzymes. This paper provides a contribution to the medical diagnosis process by shortening the time through the use of an intelligent model and helps the physician to diagnose complex cases which are otherwise difficult to perceive. Physicians make a decision according to enzyme values in normal diagnosis stage of this method.

Artificial neural networks (ANNs) can be used as one of the most popular methods for medical diagnostic processes. ANNs have already proven its effectiveness and popularity for the medical diagnostic processes with different existing applications worldwide (Chen, Lin, Wu, & McCallum, 1995; Jabbar & Mehrotra, 2008; Mobley, Schechter, Moore, McKee, & Eichner, 2000; Piecha, 2001; Prahadan, Sadasivan, & Arunodaya, 1996; Scott, 1993; Smith, Graham, & Taylor, 1996). In this paper, CMAC neural network approach using human liver test data consisted mainly of liver enzymes has been used to diagnose the liver disease in four classes. Enzymes used to identify the classes were AST, ALT, AST/ALT, Albumin, Protein, Platelet (a count test to measure how many blood cells), and Prothrombin time (PT) (a measure of blood clotting).

The remaining of the paper is arranged as follows. Section 2 briefly explains the basics of the CMAC neural network and its significant properties. Section 3 describes the CMAC-based liver diagnosis system including description of liver disease, subjects, pattern collection, training mode, diagnosis/test mode, and presents the steps of the diagnosis algorithm. Section 4 discusses the simulation results. Finally, conclusions are drawn in Section 5.

* Corresponding author. Tel.: +90 212 8663300x5530; fax: +90 212 8663412.
E-mail addresses: ibucak@fatih.edu.tr (İ.Ö. Bucak), usemra@gmail.com (S. Baki).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2010.02.112

2. Cerebellar Model Articulation Controller (CMAC)

The CMAC was firstly proposed during 1970s by James Albus whose idea was based on a model of cerebellum which is a part of the brain responsible for learning process (Burgin, 1992). The CMAC can generally be described as a transformation device that transforms given input vectors into associated output vectors (Burgin, 1992). The CMAC is an algorithm that quantizes and generalizes input, produces active memory addresses, and produces an output by summing all the weights in the active memory addresses (Kim, 1993). This process of finding the output has several steps. Fig. 1 shows a CMAC functional diagram that has two inputs and one output.

In a two-input typical CMAC, each variable in the input state vector is fed to a series of input sensors with overlapping receptive fields. The total collection of receptive fields is divided into C subsets, referred to as *layers*, which represent parallel N-dimensional hyperspaces for a network with N inputs. The receptive fields in each of the layers have rectangular boundaries and are organized so as to span the input space without overlap. Fig. 2 shows an organization of CMAC neural network receptive fields for one dimensional case (Baotic, Petrovic, & Peric, 2001).

Any input vector excites one receptive field from each layer, for a total of C excited receptive fields for any input. In Fig. 2, the total number of excited receptive fields is 4 (i.e. $C = 4$) where the hatched regions show the active or excited fields. Each of the layers of receptive fields is identical in organization, but each layer is offset relative to the others in the input hyperspace (Miller & Glanz, 1996). The width of the receptive field of each sensor produces input generalization, while the offset of the adjacent fields produces input quantization. Each input variable excites exactly C input sensors, where C is the ratio of generalization width to quantization width. Each input sensor produces a binary output which is ON if the input falls within its receptive field and is OFF otherwise. The binary outputs of the input sensors are combined in a series of threshold logic units (called state-space detectors) with thresholds adjusted to produce logical AND functions (the output is ON only if all inputs are ON). Each of these units receives one input from the group of sensors for each input variable, and thus its input receptive field is the interior of a hypercube in the input hyperspace (Miller, Glanz, & Craft, 1990).

The leftmost step in Fig. 1 presents the physical input state space. It may contain one or more input vectors (Fig. 1 shows two). These vectors are composed of discrete points. These discrete points are connected to the second step of the CMAC known as state-space detectors. The state-space detectors are often called the CMAC's virtual memory (Burgin, 1992). This transformation contains quantization process and input generalization with generalization factor (width) (Burgin, 1992). Input sensors overlap and cover *width* number of inputs. Therefore, *width* is used to indicate the number of inputs covered by overlapped input sensors (Kim, 1993). Input values are quantized into one of *quant* values and hence *width* can vary between 1 and *quant*. Low numbers usually work best (Kim, 1993).

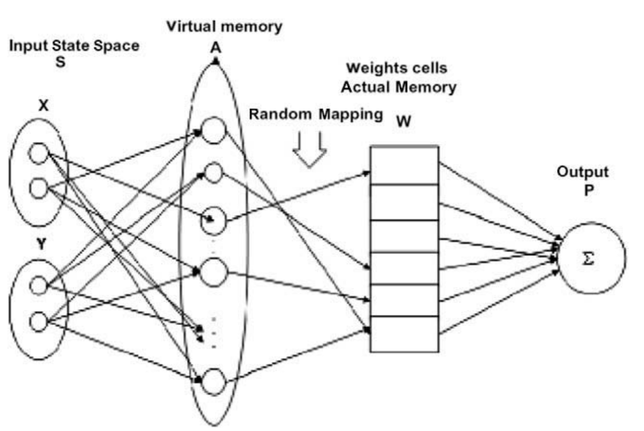

Fig. 1. A block diagram of CMAC.

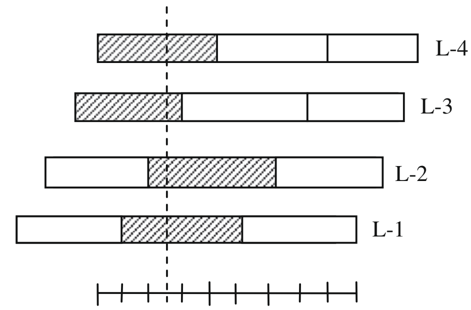

Fig. 2. Receptive field organization.

A vector of quantized input values specifies a discrete state and is used to generate addresses for retrieving information from memory for this state. Each state variable is quantized into discrete regions, called *blocks*. It is noted that the width of blocks affects the generalization capability of the CMAC. The number of blocks in CMAC is usually greater than two (Lin & Chiang, 1997). The output generalization capability of CMAC is controlled mainly by the width of the blocks. If two inputs are far apart in the input space, there will be no overlap and as the result, no generalization (Lin & Chen, 2008).

Quantization has been used due to the fact that the minimum variations in the input values do not affect the output values. Quantization levels affect the values of the input vector. The stability of inputs depends on the level of quantization. If the quantization level increases, the stability of inputs increases.

The resolution of the quantization depends on the expected maximum and minimum input values (see Fig. 3 for input quantization) (Handelman, Lane, & Gelfland, 1990; Kim, 1993). The quantization and mapping between input space and virtual memory give the CMAC the ability to the input generalization (Handelman et al., 1990; Kim, 1993) which means that the CMAC has the property that any two input vectors that are similar or close in the input space will select a highly overlapping subset of locations in the state space during mapping between input state and state-space detectors. Thus, the output response of the CMAC to similar input vectors will tend to be similar because of many memory locations that are in common. Hence, the CMAC tends to local generalization. The amount of generalization depends on the number of overlapping memory locations in the state-space detectors (Albus, 1975b).

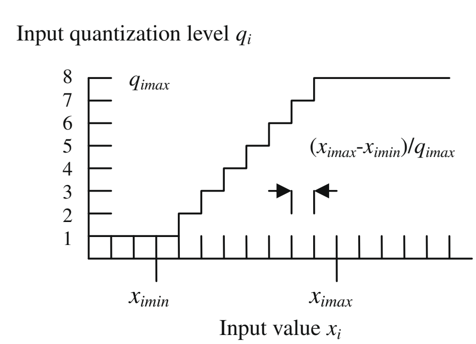

Fig. 3. CMAC input quantization.