Desktop Control System with Voice Recognition by Using Hopfield Neural Network

Khaing War Swe, Thuzar Khin
Computer University (Hinthada)
khaingwarswe@gmail.com

Abstract

This paper describe the used of Hopfield neural networks for voice recognition. Speech (voice) recognition is multileveled pattern recognition tasks and neural networks are good at pattern recognition; many early researchers naturally tried applying neural networks to speech recognition. Hopfield neural network is a simple feedback network which is able to store patterns in a manner rather similar to the brain – the full pattern can be recovered if the network is presented with only partial information. There is a degree of stability in the system, the recalled pattern is not too badly corrupted and the network can respond with a best guess. Pattern storage is generally accomplished by a feedback network consisting of processing units. The stable states of the network represent the stored patterns. Voice recognition is used for controlling the icon from the desktop and office applications. The user speaks to the computer with the desire command that is the user's training commands. Then the system understands the commands and is in many application areas especially for the handicapped person.

Keywords: Neural Network, Hopfield-network, Voice Recognition

1. Introduction

In this paper, voice recognition is used for opening desktop icon with the specified user's voice. This system is based on the training that the specified user may speak to the computer. Recurrent Back-propagation algorithm is used for training Hopfield neural network. Training is usually carried out by iterative updating of weights based on the error signal. The error signal is the difference between the desired and actual output values. Then the error signal is back-propagated to the lower layers. The weights of the network are adjusted by the algorithm such that the error is decreased along a descent direction. Back-propagation algorithm is used for many applications.

Speaker recognition systems can be divided into two namely: user-dependent and user-independent systems. In user-dependent systems, the user is expected to use the same text (keyword or sentence) during training and recognition sessions. A user-independent system does not use the training text during recognition session. Both systems perform the following tasks: feature extraction, similarity analysis and selection. This paper described the user dependent voice recognition system.

This paper is recognized as follows. Section 2 discusses related area and problem issues. The implementation of the system is presented in Section 3. Section 4 shows the simulation results of the voice recognition and training issues. The last Section is the conclusion of the system.

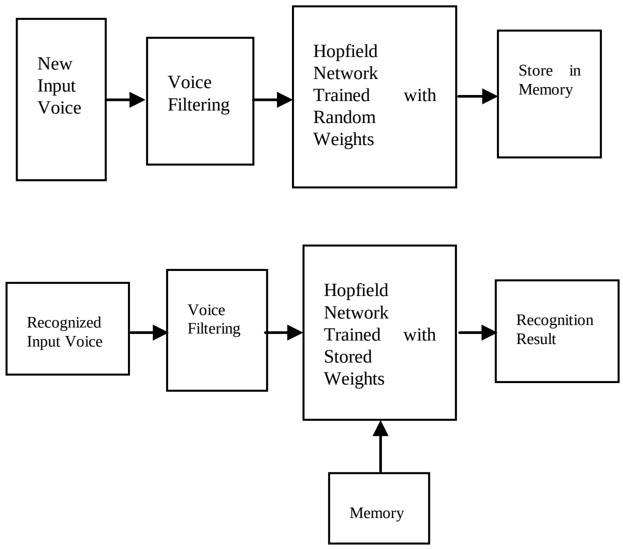

Figure 1: Block diagram of the system

2. Related Areas and Problem Issues

"Voice Recognition using Neural Network" is proposed by V.K.Ganesh, M.Viresh and S.Kumbes[2]. It represents the recognition system and used neural network for voice pattern training. M.Patricia, U.Jerica, S.Daniel, S.Miguel, L.Miguel, and C.Castillo, "Voice Recognition with Neural Networks, Type-2 Fuzzy Logic and Genetic Algorithms" proposed the case of speaker recognition by analyzing the sound signals with the help of intelligent techniques, such as the neural network and fuzzy systems [7].

T.P.Kostoulas, N. Fakotakis “Speaker Dependent Emotion Recognition Framework” described that speech is the basic way humans use in order to communicate [4]. The aim of this paper is to present an emotion recognition system, designed to predict basic emotion categories using features extracted purely from speech signal. A. Samouelian, “Knowledge Based Approach To Speech Recognition” described for speaker independent recognition, the performances of all the feature modules are similar [10].

2.1. Neural Network

Neural networks are used in a wide range of applications. One of the most successful and proven demonstrations of the ability of neural network is to take symbolic representation of words as letters and generate phoneme (sound segment) code that may be used to drive a voice synthesizer. The opposite problem is the speech recognition and another thing is that a net could be made to follow conversation about simple sentences and paraphrase simple stories. But the probably the major application of neural nets lies in the visual pattern. In this paper used the neural network for voice pattern training.

Neural network is classified into architecture structure. Neural networks are not only different in their learning processes but also different in their structures or topology. Bose (1996) has broadly classified neural networks into recurrent (involving feedback) and nonrecurrent (without feedback) ones.

In a little more details, Haykin (1999) has divided the network architectures into the following three classes.

(i) Single-layer perceptrons (feed forward networks)
(ii) Multi-layer perceptrons (feed forward networks)
(iii) Recurrent networks

2.2. Hopfield Neural Network

Hopfield network is that they are recurrent neural network, where the term “recurrent” means that the output values are feedback to the input of the network in an undirected way. Hopfield networks are mainly used in pattern recognition and optimization. Learning is performed by modifying the strength of the connections between the neurons and parameters of the activation function (thresholds). The network structure is built in a way that the outputs of each neuron are connected to the input of every other neuron.

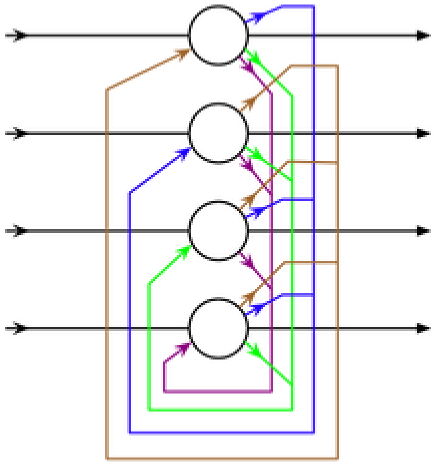

Figure: 2 A Hopfield network with four nodes

A Hopfield network is a form of recurrent neural network invented by John Hopfield. Hopfield nets serve as content addressable memory systems with binary threshold units. They are guaranteed to converge to a local minimum, but convergence to one of the stored patterns is not guaranteed.

In most simple case (when a threshold function is equal to one) the Hopfield model is described as a one-dimensional system of N neurons spins (s_i =1, i = 1, 2... N) that can be oriented along or against the local field. Hopfield net have a scalar value associated with each state of the network referred to as the energy, E, of the network. This value is called the energy because the definition ensures that if units are randomly chosen to update their activations the network will converge to states which are local minima in the energy function. Thus, if a state is a local minimum in the energy function it is a stable state for the network. The behavior of such spin system is described by Hamiltonian (also known as the energy of HNN).

$$E = -\frac{1}{2}\sum_{i \neq j}^{N} T_{i,j} s_i s_j$$

Where s_i is the state of i^{th} spin and

$$T_{i,j} = \sum_{m=1}^{M} s_{mi} s_{mj}$$

is an interconnection matrix organized according to the Hebb’s rule on M randomized patterns, i.e., on N-dimensional binary vectors S_m =(s_{m1}, s_{m2}, $s_{m,}$ N) (m=1,2,…,M). The diagonal elements of interconnection matrix are assumed to be zero ($T_{i,j}$ = 0). The traditional approach to such a system is that all spins are assumed to be free and their dynamics are defined only by the action of a local field, along which they are oriented. The algorithm of functioning of HNN is described as follows. The initial spin directions (neuron states) are oriented according the components of input vector. The local field, which acts on the i^{th} spin at time t (this field is produced by all the remaining spins of NN) is calculated as:

$$h_i = \sum_{j \neq i}^{N-1} T_{i,j} s_j$$

The spin energy in this field is $E_i = -s_i h_i$. If the spin direction coincides with the direction of the local field ($E_i < 0$), its position is energetically stable and the spin state remains unchanged at the next time step. Otherwise ($E_i > 0$), the spin position is unstable, and the local field overturns it, passing spin into the state $s_i(t+1)=-s_i(t)$ with the energy (E_i $(t+1)< 0$). The energy of the NN is reduced reducing each time any spin flips; i.e., the NN achieves a stable state in a finite number of steps. At some precise conditions each stable states corresponds to one of patterns added to interconnection matrix.

3. System Designs and Implementation

User can train the desired command with appropriate file path. Initially, user gives the input text and desired file path, and then gives the user's pronunciation of the text. Filter software processes the input voice signal to filter the noises, detect the end points and segment the input voice pattern. The resultant segmented voice pattern is limited in time frame. Then the neurons are built and trained the segmented voice pattern to dynamic HNN. After doing this, store the neuron pattern of voice into database with command and text. Store really trained interconnection weight matrix to the memory.

Training a Hopfield network involves lowering the energy of states that the network should 'remember'. This allows the network to serve as a content addressable memory system, that is to say, the network will converge to a 'remember' state if it is given only part of the state. The network can be used to recover from a distorted input and the trained state that is most similar to that input. This is called associative memory because it recovers memories on the basis of similarity.

The user gives the input voice to the computer with the specific microphone. That voice is filtered by filtering plug-in to filter the noise. The filtering output is the spectrogram. The pixel value of the spectrogram is used for constructing the Hopfield network.

At the running step, pick a node at random. The node's behavior is then deterministic: it moves to a state to minimize the energy of itself and its neighbors.

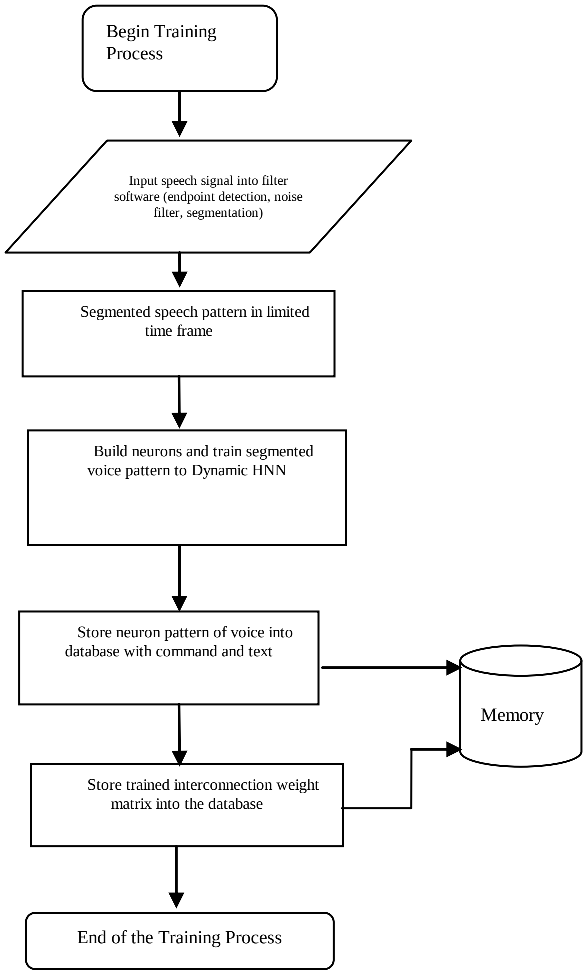

Figure: 3 (a) Voice training process flow diagram

In the recognition process, input voice signal is captured by microphone. Input speech signal from microphone is filtered by filter software. Filter software operates as in training process. Segmented speech pattern is limited in time frame. Then build the neurons pattern for dynamic Hopfield neural network until network stable. Database retrieved the trained neuron patterns for matching with input voice HNN pattern and recorded matched percent for specific pattern. The matching is operated repeatedly to all trained pattern.

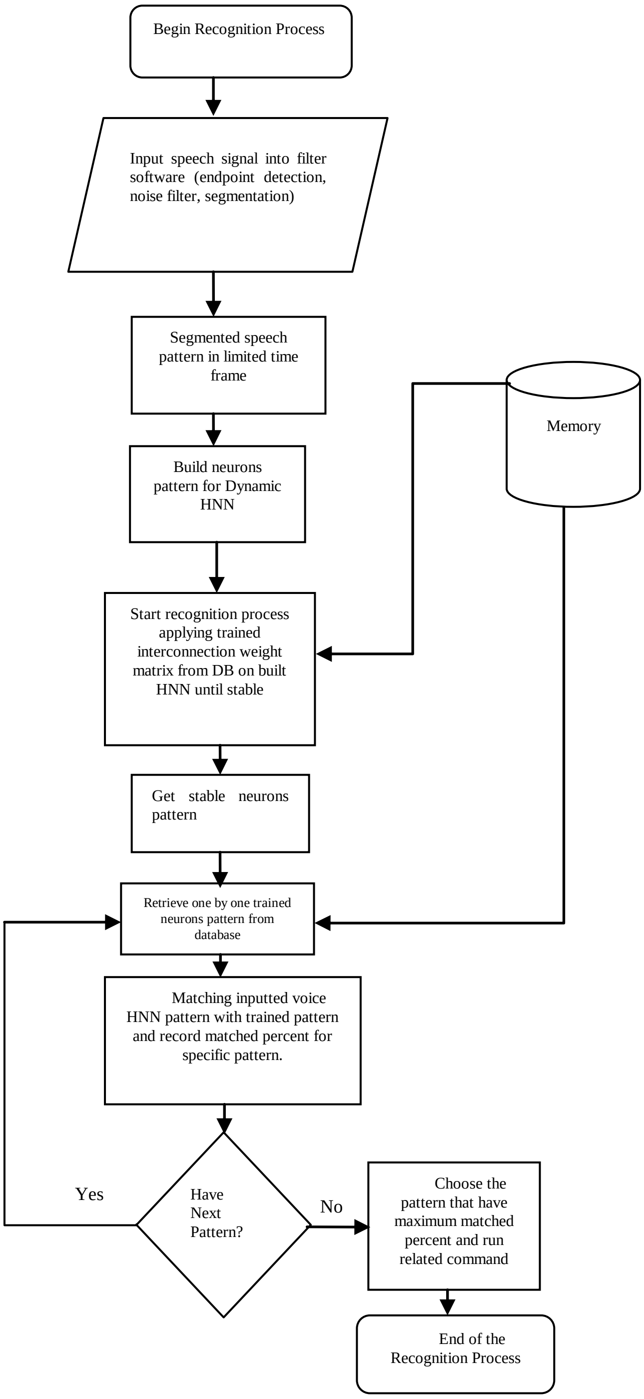

Figure: 3 (b) Voice recognition process flow diagram

4. Simulation Results

4.1. Training Section

First network must be trained. In this system, training dialog can be display by clicking training menu. User can choice desired executable file, text and record voice command then have to click training button to train voice pattern. Inputted voice command will be segmented and filtered by filtering plug in and output as spectrogram.

Figure: 4 (a) Dialog box for voice training

4.2. Recognition Section

Figure: 4 (b) Recognition section

The output from filter software is being recognized using weights from interconnection matrix. Then get one output pattern of neuron as result. The result neurons pattern is compare with train neuron patterns in memory system. Percentage of the most similar pattern will be shown as accuracy number. The accuracy percent should be 50 at least. And related command of this pattern will be performed.

4.3 Creating Training Commands

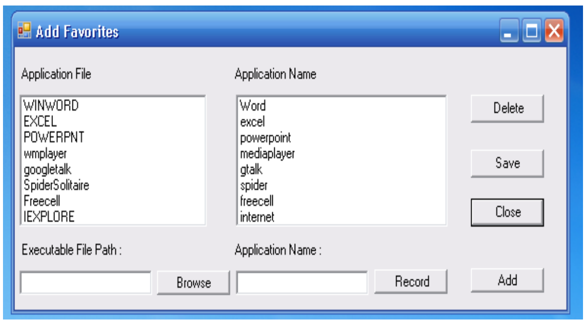

Figure: 4(c) Command list for adding command command

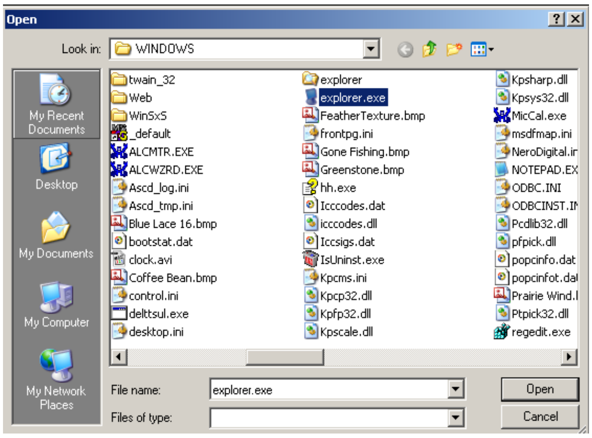

Figure: 4(d) Choosing the application file path

Adding new command in the command lists, it will need to point the destination of executable file of the related command and related application name.

5. Conclusion

In this paper described for voice recognition for the case of speaker dependent system. This system is an intelligent approach for voice recognition for the case of speaker recognition by using neural network. The advantage is that people may speak commands to operate the application open and close. Neural network are known to be excellent classifiers, but their performance can be hampered by the size and quality of the training set. In this paper showed the voice recognition for the case of speaker dependent system. Hopfield network neurons are constructed in dynamic. Neurons number based on the pixel value of spectrogram. This system is applied for Window XP only.

6. References

[1] S.R.K. Anjani, a Thesis in Electrical Engineering, Submitted to the Graduate Faculty of Texas Tech University, "Speech System for a Voice-impaired Person".

[2] V.K.Ganesh, M.Viresh and S.Kumbes, "Voice Recognition Using Neural Network". Institute for Information Sciences and Technology (IIST) Massey University, New Zealand.

[3] N.N.Karnik, and J.M.Mendel, "An Introduction to Type-2 Fuzzy Logic System",Technical Report, University of Southern California,1988.

[4] T.P.Kostoulas, N. Fakotakis "Speaker Dependent Emotion Recognition Framework".

[5] A. Mayer, G.Wiesbauer, M. Spitzlinger Leader: Dipl.Ing. Mag. Roland Schwaiger, "Applications of Hopfield Networks", Version from November 4, 1999.

[6] X.Y.Nancy, "Using the Modified Back-propagation Algorithm to Perform Automated Downlink Analysis".

[7] M. Patricia, U. Jerica, S. Daniel, S. Miguel, L. Miguel, and C. Castillo, "Voice Recognition with Neural Networks, Type-2 Fuzzy Logic and Genetic Algorithms".

[8] N. Selvanathan, and W. Lee Chee, Faculty of Computer Science & Information Technology, "Hopfield Model for Shortest Path Computation and Routing in the ATM Network**".**

[9] T. P. Singh, Dr. M P Singh, "Performance Analysis of Hopfield Model of Neural Network with Evolutionary Approach for Pattern Recalling".

[10] A. Samouelian, "Knowledge Based Approach to Speech Recognition".