TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
THAPATHALI CAMPUS

Speaker Recognition Using MFCC And Back Propagation Neural Networks

by

Kushal Gajurel	(2070/BEX/328)
Santosh Nepal	(2070/BEX/337)
Shiva Bhandari	(2070/BEX/340)
Surendra Shrestha	(2070/BEX/345)

A PROPOSAL
SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN ELECTRONICS AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
THAPATHALI CAMPUS
KATHMANDU, NEPAL

February, 2017

ABSTRACT

'**Speaker Recognition Using MFCC And Back Propagation Neural Networks'** aims in speaker recognition on the basis of speech sample fed to the processor, Raspberry PI that uses Python as programming language, using feature vector extraction by MFCC (Mel-Frequency Cepstral Coefficient) algorithms. Feature vector calculated is then reduced into a linear transformation vector using PCA(Principal Component Analysis) which targets for reduction in noise in data and low time complexity in training neural network. Thus reduced lower dimensional vector is now used to train the neural network, which learns by a method called gradient descent, achieved through an algorithm known as back propagation.

This document presents the proposal for developing a speech processing algorithms that can be used for speaker recognition and further applications in surveillance, navigation, forensic study, and data training.

Keywords: Speaker Recognition, Signal Processing, MFCC, Neural Network, PCA, Feature Vector , Reducing Dimension

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADC	Analog to Digital Converter
ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
BPNN	Back Propagation Neural Network
CSI	Camera Serial Interface
DSI	Display Serial Interface
DSP	Digital Signal Processing
FFT	Fast Fourier Transform
GPU	Graphical Processing Unit
GUI	Graphical User Interface
HDMI	High Definition Multimedia Interface
IDLE	Integrated Development Environment
IoT	Internet Of Things
ISP	Internet Service Provider
ISR	Interrupt Service Routine
IT	Information Technology
LDA	Linear Discriminant Analysis
LP	Linear Programming
LT	Linear Transformation
MFCC	Mel-Frequency Cepstral coefficient
OOP	Object Oriented Programming
PCA	Principal Component Analysis
PID	Packet ID
USB	Universal Serial Bus

1. INTRODUCTION

1.1 Background Introduction

Speech, a natural form of communication among humans, is characteristic to individuals pertaining to various parameters which characterize and identify attitude, identity, gender and other emotional state of the speaker [1]. Thus, by using the tools of DSP(Digital Signal Processing), speech can be processed and speaker recognized for various applications such as authenticity, voice based navigation, surveillance and many others.

The developments of speech recognition system have been made for many years. In all audio processing, it is necessary to convert the speech input into the feature matrix representation in order to recognize the unknown speech signal. The process converting the speech signal to the feature matrix representation is called the feature extraction of speech signal [2].

Recognizing a speaker mainly involves identification and verification of the speaker. Identification incorporates highlighting the features of the speaker in the sample speech and classifying those features. On the other hand, verification involves identifying a speaker previously involved in training the neural network by supervised learning, with the help of some sample speeches. Hence a speaker can be recognized.

1.2 Scope and Application

A venture like speaker recognition has a growing scope when viewed from variety of aspects. This project, on completion, can be used in numerous applications some of which are mentioned below :

- Transaction Authentication in various commercial ventures
- User identification in account management
- Home and workplace security
- Human machine communication
- Voice based navigation
- Caller identification in call center for call blocking

2. LITERATURE REVIEW

2.1 Speech Recognition

Speech and speaker recognition is a thoroughly studied and researched topic and many advancements have been achieved in this field. Speech processing has had much of research since 1975, which works in audio communication and speech recognition and verification [1]. AT&T Bell Labs started its automatic speaker verification system in 1970. Researchers there chose measurements that are largely insensitive to the phase and spectral amplitude distortions likely over telephone lines.[1]

The mainly used speech recognition engine is that of Google. Google Speech Recognition Engine is a platform inspired from the human brain's neural network. What it basically does is chops the spectrogram of the audio and sends them to eight different computer on the servers where they are processed using neural network models. Due to Google's ample cloud processing power which is used to analyze the patterns in the spectrogram and they are further used to predict new patterns like a human neural network does to accomplish a new task [3].

2.2 Feature extraction by MFCC

Paul Mermelstein in 1976 developed a technique crediting Bridle and Brown, who used a set of 19 weighted spectrum-shape coefficients given by the cosine transform of the outputs of a set of non-uniformly spaced band-pass filters. The filter spacing is chosen to be logarithmic above 1 kHz and the filter bandwidths are increased there as well.

The article referenced [2] has demonstrated a similar way for speaker recognition where the authors used MFCC for feature extraction to resolve the signal into the feature vector. The computation of effective and efficient speech features is necessary for accuracy or recognition rate of speaker identification. The features should be robust against noise, should have low variability for sessions of same speaker and large between-speaker variability [4]. For speaker verification, regularized linear prediction based MFCC extraction is considered in [5]. In the computation of MFCCs, the classical hamming

window is modified to obtain derivatives of discrete time Fourier transform coefficients in [6].

2.3 Dimension reduction by PCA

Principal Component Analysis is a popular data processing and dimension reduction method. Among the various notable works, in [6] the researchers have applied a very similar method (LDA, PCA and LP transformation – concatenating the LDA and PCA coefficients into one vector) to transform the base feature vectors. After investigation of this methodology (but only the PCA transformation will be used) with completion the experimental setup with concatenating the base vectors, which were used as input to the PCA-based processing. This similar approach resulted in comparable, in some cases better results in comparison with the above mentioned publications. In [7] the PCA was applied to the multi-frame context windows.

3. PROBLEM DEFINITION

In recent years, with introduction of modern forms of automation and digitization, life has undoubtedly become easier. But the challenges are also increasing in the same manner. Issues like home security, voice based navigation have encountered identity resolving problem. When there is issue of authenticity in voice command, we move from what is said to who said it. It could be possible to recognize speech directly form the digital wave-form, but owing to large variability of speech signal, for reducing the variability we use feature extraction especially on the source of information. One of the major advantage of using the MFCC technique over others is that the frequency bands are positioned in logarithmic fashion in MFCC, so, it approximates the human system response more closely than any other system [8].

.

4. OBJECTIVE

1. To exploit machine learning with artificial intelligence in training a Neural Network with speech samples
2. To implement a text independent speaker recognition and verification system using Mel-frequency Cepstral Coefficients, Principal Component Analysis and Back Propagation Neural Network

5. PROPOSED SYSTEM

In this project we have proposed to use a USB microphone for the input unit through which audio signals are recorded and fed to the processing unit, a Raspberry PI 3, where all the algorithms as in the block diagram shown below

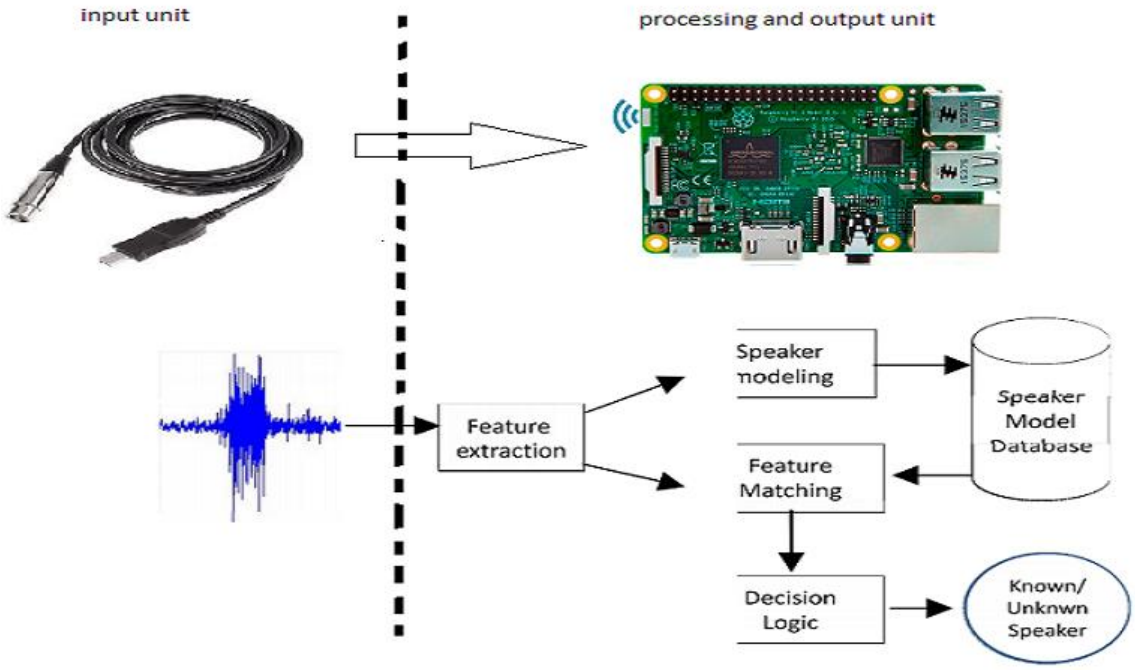

Figure 1 : Block diagram of speech recognition

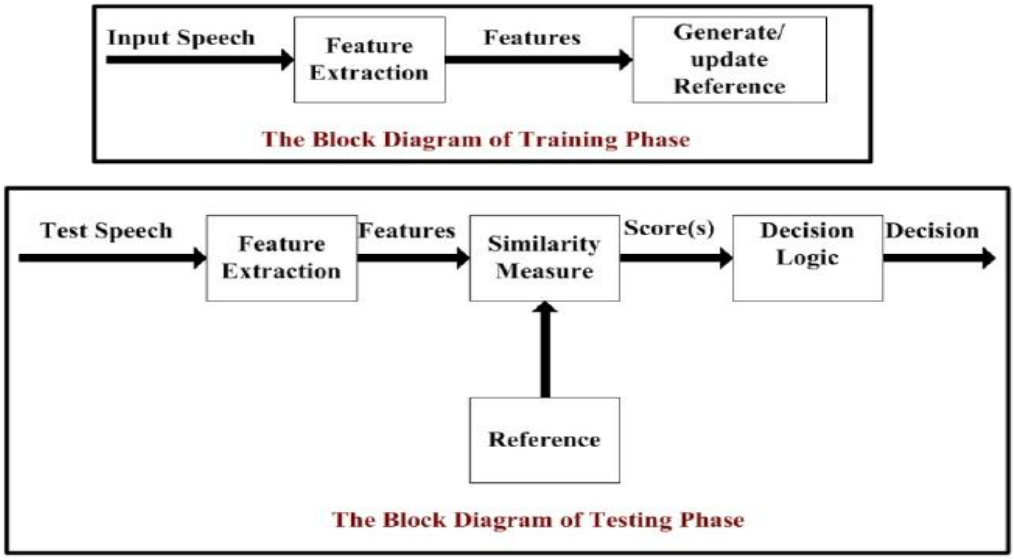

Figure 2: Detailed two phase proposed system

6. METHODOLOGY

The proposed system mainly implements hardware implementation and software development. These methods can be used to describe the total functionality of the proposed system.

6.1 Hardware Implementation

6.1.1 Hardware requirements

The various hardware components to be used in the project are introduced briefly below:

6.1.1.1 USB Microphone

Microphone is transducer to convert sound into an electrical signal. An USB microphone uses a USB communication protocol and employ different methods to convert the air pressure variations of a sound wave to an electrical signal.[9]

Figure 3 : USB microphone [9]

6.1.1.2 Raspberry Pi 3

The Raspberry Pi 3 Model B features a quad-core 64-bit ARM Cortex A53 clocked at 1.2 GHz. This puts the Pi 3 roughly 50% faster than the Pi 2. Different operating systems like Ubuntu Linux, Debian, Ubuntu mate etc can be installed in it. It uses Broadcom GPU, with a RAM of 1GB, with networking capabilities including 10/100 Ethernet, 2.4GHz 802.11n wireless. The storage memory is an external micro SD. There are 40 general input output pins with port features of HDMI, Ethernet, Camera Serial Interface, DSI (Display Serial Interface). [10]

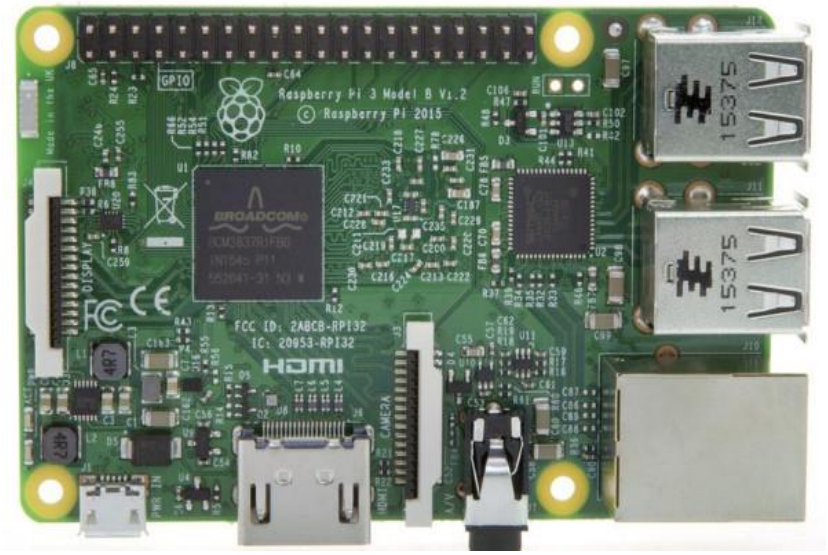

Figure 4: Raspberry Pi 3 [10]

6.1.2 Hardware communication

The communication between the hardware components is by the use of USB communication protocol. The microphone receives the speech sample. As there is no predefined input device in Pi, the microphone is added as the input device recognized by Raspberry Pi as the sound card which will handle the input data. For further processing , ALSA utility is used , which will convert the sample input fed into '.wav' file. This file is now processed by the processor based on the various algorithms defined in the software to produce the logical output.

Data transfer using USB communication Protocol

The actual data is sent across the bus in packets. Each packet is a bundle of data along with information concerning the source, destination and length of the data, and also error detection information. Each packet is made up of a set of components called fields including the following.

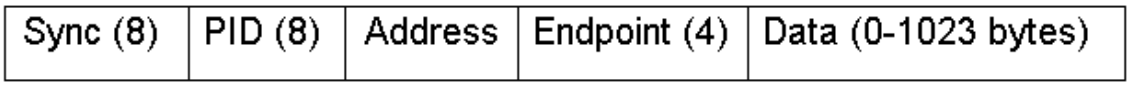

Figure 5: data fields in USB communication [11]

An eight bit "SYNC" synchronization field used by inputs to correct their timing for accepting data. Part of this field is a special symbol used to mark the start of a packet. The 8 bit Packet Identifier (PID) which uses 4 bits to determine the type, and hence format, of the packet data. The remaining 4 bits are a 1's complement of this, acting as check bits. Part of this field determines which of the four groups (token, data, handshake, and special) that the packet belongs to, and also specifies an input, output or setup instruction. An address field which gives the address of the function on the end of the pipe to be used. The 4 bit endpoint field, giving the appropriate endpoint which sends or receives the packet. A data field consisting of 0-1023 bytes.[11]

6.2 Firmware Development

6.2.1 Software requirements

The software to be used in the development of firmware is as follows:

6.2.1.1 Python

Python is an open source high-level programming language for general programming to advanced implementation. An interpreted language, Python has a OOP design philosophy which emphasizes code readability and a syntax which allows programmers to express concepts in fewer lines of code than most of the programming languages. The language provides constructs intended to enable writing clear programs on both a small and large scale.

Python GUI features a dynamic type system and automatic memory management and supports multiple programming paradigms, including object-oriented, imperative, functional programming, and procedural styles. It has a large and comprehensive standard library. Python is widely used and interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems.

6.2.1.2 Anaconda

Anaconda is an open source distribution of the Python programming languages for large-package management and deployment. In this project, we're going to use it as an interpreter for our code written. It is the packet management system of python distribution.

6.2.2 Algorithms and Techniques

For the process of signal processing, training neural networks and recognition of the speaker, we use various existing algorithms. Some of which are as follows.

6.2.2.1 Mel-Frequency Cepstral Coefficient (MFCC)

The acoustic features of a person are embedded within the sample of speech uttered by the speaker. For the process of training and recognition of speaker those features need to

be extracted and refined. This extraction is obtained through the algorithm known as Mel-Frequency Cepstral Coefficient (MFCC). The extracted feature acts as the fingerprint of the sound files which can then be used to train a neural network or to compare the unknown speaker with the existing speaker in database. The accuracy or recognition rate of speaker identification depends on extracting effective and efficient speech features.

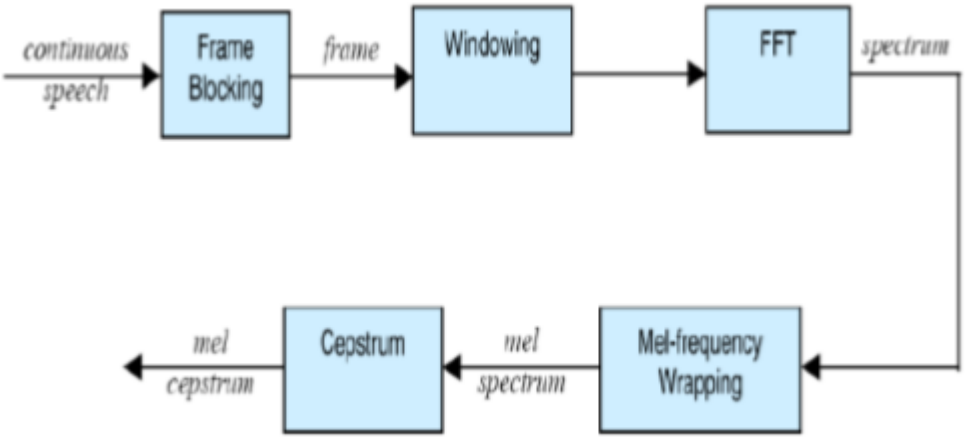

Figure 6 : Block Diagram of MFCC

The various steps involved in the feature extraction process are

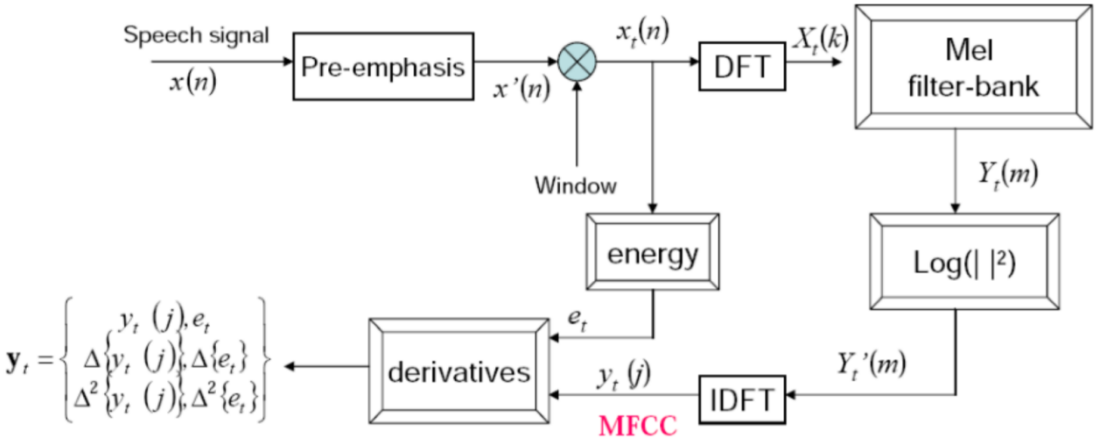

Figure 7: Feature extraction process [11]

A. **Pre-emphasis**: This step boosts the higher frequencies of the signal. The first order high pass filter given by following equation is applied to speech signal.[11]

$$Y(n) = x(n) - \alpha x(n-1),$$

Where x (n) is the input speech signal and 0.9 <= α <= 1, we take α = 0.97

B. **Framing**: The audio / speech signal is quasi periodic in nature. It is assumed that the speech signal is stationary over short period of few milliseconds , so to remove these dead signals , frames are created with certain overlapping such that no info is lost.

C. **Windowing**: Windowing makes signal smooth by tapering signals, by Hamming window given by : $w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{M-1})$

D. **FFT**: For finding the short-term frequency content of the signal, i.e. conversion from time domain to frequency domain. Fourier transform is used in windowed frame and the spectrum computed. Then power spectrum is estimated by taking the absolute value of signal. FFT is demonstrated as

$$Y_2[n] = \sum_{k=0}^{N-1} Y_1[k] e^{-\frac{2\pi jkn}{N}}$$

E. **Mel-filter bank** : Human ear perceives sound signal non linearly, linear (upto 1 kHz) and logarithmic (above 1kHz). Mel filter, used to imitate human ear , perceives sound by calculating Mel frequency based on frequency of sample frequency given as:

$$F_{mel} = 2595 * log_{10}(1 + \frac{f}{700})$$

The windowed signal spectrum on multiplication with these coefficients and log operation performed to separate vocal tract response from excitation [11].

F. **DCT (Discrete Cosine Transform)**: Used to de-correlate log energies and compress the signal [11].

G. Cepstrum: Cepstrum name was derived from the spectrum by reversing the first four letters of Spectrum. The speech signal is represented as a convolution between slowly varying vocal tract impulse response and quickly varying glottal pulse .The goal is to separate these two parts. In the time domain, convolution becomes multiplication. Hence, by taking the inverse FFT or DCT of the logarithm of the magnitude spectrum glottal pulse and Impulse response are separated [12].

H. Mel-Frequency Cepstral Coefficients

In this final step, the Mel-spectrum scale is converted back to standard frequency scale and then to time. This spectrum provides a good representation of the spectral properties of the signal which is key for representing and recognizing characteristics of the speaker.[12]

$$C_n = \sum_{k=1}^{K} (\log S_k)[n(k - \frac{1}{2})\frac{\pi}{2}]$$

Where n=1,2,…,K , normally K=20

6.2.2.2 Principal Component Analysis (PCA)

The analysis of finding the principal component in the data set is called PCA.it is one of the most common dimensionality reduction techniques. Varying from pattern recognition, computer vision to data analysis it has many uses. The acoustic vector formed by MFCC feature extraction, the feature matrix(of dimensionality N) is subjected to dimensional reduction process , such that new feature matrix(of dimensionality n, such that n<N) is formed. Thus formed new matrix is more compact, but with no loss of information. The method is based on the assumption that most of the information of the original feature data is in the directions along which the variations are largest and thus PCA convert using principal components to increase the performance of the system [2].

For M feature vectors corresponding to speech signal in training set, we create a centered data matrix as follows:

$\phi_i = x_i - x' = x_i - \frac{1}{M}\sum_{i=1}^{M} x_i$ where ϕ_i is i-th centered and x' is mean.

Now, A= $[\phi_i \phi_i \ldots \ldots \phi_M]$ is the centered matrix.

Principal components can be given by K leading eigen vectors of global covariance matrix C resulting from eigen decomposition as $Cu_i = \lambda_i u_i$ where u_i are eigen vectors and λ_i are eigenvalue of covariance matrix.

$$C = \frac{1}{M-1} \sum_{n=1}^{M} \phi_n \phi_n^T$$

The dimensionality reduction step is performed by keeping only the eigenvectors corresponding to the K largest eigenvalues (K < N) and put them into matrix U_k=[u_1, $u_{1..}$ U_n], where $\boldsymbol{\lambda}_1 > \boldsymbol{\lambda}_2 > \ldots > \boldsymbol{\lambda}_K$. The linear transformation is now :

$$y_i = U_K^T x_i$$

Where y_i represents the transformed vector and U_k is reduced rank PCA transformation matrix. The determination of K can be done via comparative criterion with threshold T varies from (0.9, 0.95).

$$\frac{\sum_{i=1}^{K} \lambda_i}{\sum_{i=1}^{N} \lambda_i} > T$$

6.2.2.3 Back Propagation Neural Network

Artificial Neural Networks (ANNs)

Neural network is the simulation of many interconnected network inside a computer that can help to learn things, recognize patterns, and make decisions in humanlike way. Fundamentally, they are good function estimator.

Neural network are structured as a series of layers, each composed of one or more neurons. The network is trained using input vector, then a series of operation is done, and finally an output is produced. To train the network to estimate an unknown functions, we give it a collection of data points, which we denote the "training set" that the network will learn from and generalize on to make future inferences.

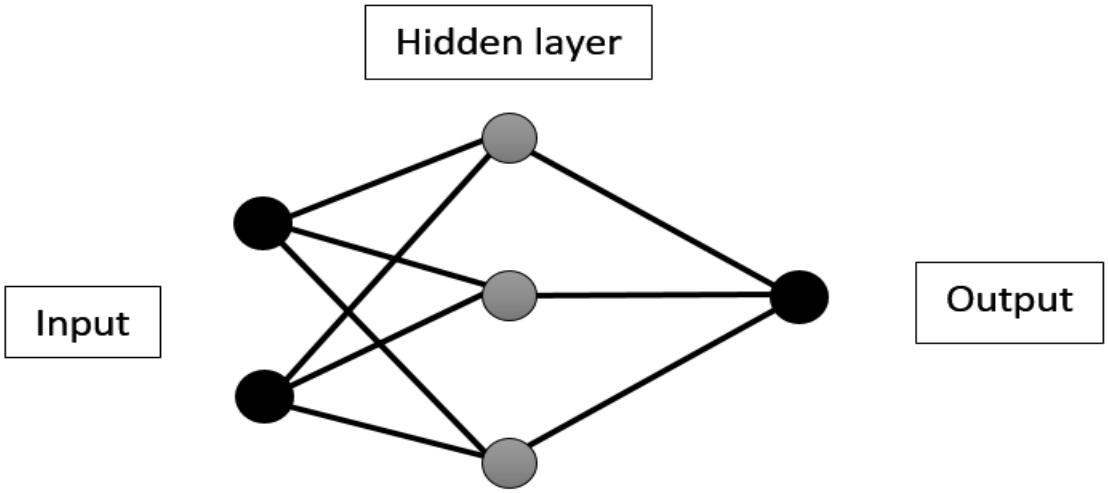

Figure 8 : Artificial Neural Network

As shown in above figure each circle is a neuron, and the arrows are connection between neurons in consecutive layer. The connections between one unit and the other are represented by a number called a weight. When it's learning or being trained or simply testing, the input speech signal are fed into the network via the input units. This process is called feed forward network. All nodes are not activated/fired at all the time. The

activation of each node is based on the outputs of the previous layer and a set of weights which is given by

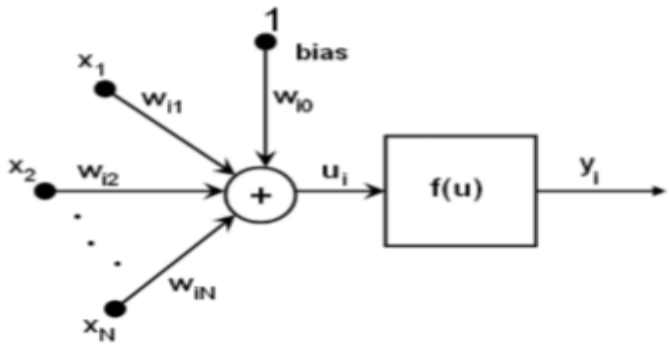

Figure 9 : Multiple input neuron

$$A_n = act\,(W_1x_1 + \ldots + W_nx_n)$$

And the threshold value is calculated using **sigmoid function** given by

$$S(\alpha) = \frac{1}{1 + e^{-\alpha}}$$

where z is the sum of product of weight and input values.

For a neural network to learn, we use typical feedback process called back propagation. This process compare the output it was meant to produce and the output fed, and use the difference between them to modify the weight of the connections between the unit in the network, working from the output units through the hidden units to the input units. With time, back propagation causes the network to learn, reducing the difference between the actual output and intended output. While training a network, we have a metric called cost function that looks at the function the network has inferred and uses it to estimate values for the data points in our training set. The value of this cost function should be as low as possible and should be more or less convex. One of the problems that occur during this process is over fitting which is solved using regularization term to the cost function. We will use gradient descent algorithm to reduce thus obtained data for faster calculation. Our cost function isn't a simple parabola but a complex multidimensional function with countless minima for which we will use back propagation.

7. TIME SCHEDULE

		Start: Jan 9,2017 NO OF WEEKS												
S.NO	WORK SCHEDULE	1 ,2	3 ,4	5 ,6	7 ,8	9 ,10	11 ,12	13 ,14	15 ,16	17 ,18	19, 20	21 ,22	23 ,24	25 ,26
1	project identification													
2	proposal writing													
3	research and reviews													
4	Programming And hardware implementation													
5	testing and debugging													
6	report writing													

Table 1 Gantt Chart

The time window presented here doesn't necessarily guarantee accurate time that will be spent for each phase. This estimate is only a rough estimate based on the number of people participating and helping in the project. The overall duration may be shortened or may lengthen based on the situation and problems faced during project implementation.

8. COST ESTIMATION

S.N.	Item	Unit Cost (Rs)	Qty	Total (Rs)
1	Raspberry PI 3	6,000	1	6,000
2	USB microphone	1,200	1	1,200
3	32 GB micro SD	1,500	1	1,500
4	Miscellaneous		-	1200
	Total			**9,900**

Table 2 Cost Estimation

REFERENCES

[1] J.Larry.(2012,December5).IntroductiontoDigitalSpeechProcessing[Online].Availabl e: www.coursehero.com/file/6664278/Lecture-1winter2012robotvideo/. Accessed: 29–1–2017

[2] H. Trang, T. H. Loc, and H. B. H. Nam, "Proposed combination of PCA and MFCC feature extraction in speech recognition system," 2014 International Conference on Advanced Technologies for Communications (ATC 2014), Oct. 2014.

[3] cezarepc, "The secret of Google‟s amazing voice recognition revealed: It works like a brain," Phone Arena, 2013. [Online]. Available: http://www.phonearena.com/news/The-secret-of-Googles-amazing-voicerecognition-revealed-it-works-like-a-brain_id39938. Accessed: Nov. 17, 2016.

[4] Tomi Kinnunen, Haizhou Li, "An overview of text-independent speaker recognition: From features to supervectors", Journal on Speech Communication, Elsevier, vol.52, issue 1, pp. 12–40, 2010.

[5] Cemal Hanilçi, Tomi Kinnunen, Figen Ertaú, Rahim Saeidi, Jouni Pohjalainen, and Paavo Alku, "Regularized All-Pole Models for Speaker 163 Verification Under Noisy Environments", IEEE Signal Processing Letters, vol.19, no.3, pp. 163-166, March 2012.

[6] X. Wang and D. O'Shaughnessy, „Improving the efficiency of automatic speech recognition by feature transformation and dimensionality reduction", In: Eurospeech, pp. 1025-1028, Geneva, September 1-4, 2003.

[7] P. Somervuo, "Experiments With Linear And Nonlinear Feature Transformations In HMM Based Phone Recognition", In Proceedings of ICASSP, pp. 52-55, 2003.

[8] PoojaV.Janseetal.,“AComparativeStudybetweenMFCCandDWTFeatureExtractionTechnique,”InternationalJournalofEngineeringResearch&Technology(IJERT)., Gandinagar,ISSN:2278-0181,Vol.3,Issue1,January2014.

[9] "Microphone," in Wikipedia, Wikimedia Foundation, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Microphone. Accessed: Feb. 19, 2017.

[10] B. Benchoff, "Introducing the raspberry pi 3," Hackaday, 2016. [Online]. Available: http://hackaday.com/2016/02/28/introducing-the-raspberry-pi-3/. Accessed: Feb. 19, 2017.

[11] "USB (universal serial bus): How it works and what it does,". [Online]. Available: http://www.geoffknagge.com/uni/elec101/essay.shtml. Accessed: Feb. 19, 2017.

[12] Pawan K. Ajmera, Dattatray V. Jadhav, Ragunath S. Holambe, “Textindependent speaker identification using Radon and discrete cosine transforms based features from speech spectrogram”, Journal on Pattern Recognition, Elsevier, vol.44, issue 10-11, pp. 2749-2759, 2011

[13] S. Tripathi and S. Bhatnagar, "Speaker recognition," 2012 Third International Conference on Computer and Communication Technology, Nov. 2012.