ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

Machine Learning Approach for Voicing Detection

BY

Mohammed Abebe Yimer

A THESIS SUBMITED TO

THE SCHOOL OF GRADUATE STUDIES OF ADDIS ABABA UNIVERSITY IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTER OF SCINECE IN COMPUTER SCIENCE

March, 2012

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF COMPUTER AND MATHEMATICAL SCEINCES

DEPARTMENT OF COMPUTER SCIENCE

Machine Learning Approach for Voicing Detection

BY

Mohammed Abebe Yimer

Signature of the Board of Examiners for Approval

	Name	Signature
1.	Dr.Sebsbie Hailemariam,Advisor_	_________________
2.	_____________________________	_________________
3.	_____________________________	_________________

Acknowledgments

First and for most I would like to thank and praise the almighty God (Allah) for his incredible guidance, support and blessing my life in all my activities.

I wish to pass my appreciation to the following people or entities for their diverse contribution directly and/or indirectly throughout my study and this research work. Nevertheless, it is hard to know where to start thinking and acknowledging people for help and assistance that they have shown me. The problem is that I am likely to omit some very important names. The following are those to whom I am particularly indebted.

I would like to express my deepest gratitude to my research advisor, Dr. Sebsbie H/Mariam, for the guidance, encouragement, and friendship that he managed to extend to me. Without his supervision, invaluable suggestions, dedicated advice and remarks, the project wouldn‟t have been a reality. I really appreciate his encouragement, friendly approach and fatherly hood consultancy.

I am greatly obligated to forward my gratitude to my parents and my families for their intensification of the whole spectrum of my life. Particularly I have a special thank you to my dad, Abebe Y., and my mom Zemzem K. for having raised me in a stable and loving environment, which has enabled me to come so far and whose great ambitions to see my accomplishment, was a lot to me in my study and throughout my life.

I would like to forward my heartfelt love and thanks to my sister Jemila A. for her encouragement and moral support over all those times in my study.

Many colleagues and friends have influenced this thesis; I wish to express my love and gratitude to all my friends and colleagues working at Arba Minch University and Admas University College and to all my friends and class mates for their invaluable support in every aspect for this thesis. Friends whose role in my study are invaluable and need to be at least mentioned are Teklay G., Tulu T., Nirayo H., Abreham W., and Eshete D.

Finally, I would like to thank everyone who has contributed positive impacts to the successful realization of this thesis work, as well as expressing my apology that I could not mention individually one by one.

Table of Contents

Contents ***Pages***

List of Figures

List of Tables

Acronyms and Abbreviations

ANN	Artificial Neural Network
AR	Auto Regressive
ASR	Automatic Speech Recognition
CART	Classification And Regression Tree
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DSP	Digital Signal Processing
EBP	Error Back-Propagation
EM	Expectation Maximization
GMM	Gaussian Mixture Models
HMM	Hidden Markov Model
LPC	Linear Predictive Coefficients
MFCC	Mel-Frequency Cepstral Coefficients
MFN	Multilayer Feedforward Network
ML	Maximum Likelihood
MLP	Multilayer Perceptron
RMS	Root Mean Square
SNR	Signal to Noise Ratio
STFT	Short Time Fourier Transform
VDA	Voicing Detection Algorithm
ZCR	Zero-Crossing Rate

Abstract

Voiced/unvoiced/silence speech segment detection is a method of assigning and labelling a specific speech category (voiced/unvoiced/silence) to a speech segment.

Assigning speech categories to speech segment in a speech sound is an important component of many speech processing systems. An accurate classification of a speech segment as voiced/unvoiced/silence with voicing detection system is often used as a prerequisite for developing other higher level and efficient applications of speech processing systems such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, speaker identification, and the recognition of speech pathologies.

The interest in speech segment category discrimination has intensified lately due to the increasing demand for potential use in a number of commercial or non-commercial speech-based systems. Current personal communication systems such as a cellular phone are examples of commercial systems that integrate speech coding and speech recognition capabilities in their operation.

In this study a supervised method of voiced/unvoiced/silence speech segment detection is proposed. Text corpus with size of 900 sentences is collected from political news, economy news, sport news, health news, fictions, Bible, penal code and Federal Negarit Gazzeta. These texts are recorded by one male speaker to prepare the speech corpus. Both text and speech corpuses are split in to training (66.67%) and test (33.33%) data sets.

ANN based voiced/unvoiced/silence classifier in particular an MLP with single hidden layer and 25 neurons on the hidden layer shows a high classification performance than other models tested. The network has 15 neurons on the input layer and 3 neurons on the output layer that match the number of features in the feature vector and the number of classes to be classified respectively (MLP 15-25-3). Energy, ZCR and 13 MFCCs of the speech signal are used as feature vectors to train and test the classifier model selected. The feature vectors are extracted for 20, 25, 30 and 35 millisecond of speech segment to see the effect of frame length on the classification performance of the classifiers.

The evaluation of the experiments shows that best performance with the selected classifier model and feature vector is achieved on 35 millisecond frame size. The MLP classifier shows an accuracy of 89.69%. Hence, it is found that an MLP with single hidden layer and 25 neurons on the hidden layer outperforms other classifiers tested.

Keywords: Voicing Detection, Voiced/Unvoiced/Silence, Machine Learning

CHAPTER ONE

INTRODUCTION

1.1 Background

Digital signal processing (DSP) is concerned with the representation of signals as a sequence of numbers and the algorithmic operations carried out on the signals to extract specific information contained in them. DSP is mainly concerned with the processing of a discrete-time signal, called the *input signal*, to develop another discrete-time signal, called the *output signal*, with more desirable properties.

Digital signal processing systems are available that will do almost everything that analogue signals can do, which are more versatile, that can be easily changed (programmable), they can be made to process signals identically (repeatable) and are not affected by temperature or ageing (physically stable) [21]. DSPs can be used for speech and audio processing (speech coding/compression, speech analysis/synthesis, speech recognition, audio coding), image and video processing (still image coding, video coding), adaptive filtering (echo cancellation in telephone lines, active noise control, medical signal processing).

In certain applications, such as speech analysis-synthesis, noise suppression and enhancement, it may be necessary to extract some key properties of the original signal like voiced, unvoiced or noise speech sounds, using specific digital signal processing algorithms. It is also possible to investigate the properties of a discrete-time system by observing the output signals for specific input signals.

The knowledge of acoustic speech feature in particular voiced or unvoiced segment plays an important role in many speech analysis-synthesis systems. Thus the issue of voicing detection (voiced/unvoiced) algorithms (VDAs) has been one of the topics most analysed in the field of speech processing research during the last three decades [5].

Voiced/Unvoiced detection involves identifying the regions of speech when there is significant glottal activity (i.e., the vibration of vocal folds). Such regions of speech are generally referred to as voiced speech. The unvoiced regions of speech include both silence (background noise) as well as unvoiced speech (such as voiceless fricatives and stops) [9]. The vocal tract is the path through the human vocal organs that produce speech. The particular acoustic sound that is created is dependent on the action and position of the vocal organs. The vocal organs shape the frequency characteristics of the vibrating air travelling through the vocal tract.

Voiced speeches are produced with vibration of the vocal cords. Air flow from the lungs and up the trachea is modulated by vibrations of the vocal folds, located in the larynx. Voiced speeches include all English vowels and some consonants, such as /m/, /n/, /l/, /w/, /b/, /d/, /g/, /v/, and /z/. Unvoiced speeches are produced by a turbulent air flow crossing some constriction in the vocal tract, without vibration of the vocal cords. Unvoiced sounds include consonants like /p/, /t/, /k/, /f/. Silence is produced with the absence of speech sound. The signal of a voiced sound is more or less periodic, while an unvoiced signal is noise-like.

The correct voiced/unvoiced/silence classification of a sound signal is essential in several speech processing systems. Voicing detection algorithms have more application in the field of speech processing such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, voice activity detection, speaker identification, and the recognition of speech pathologies.

There are various aspects to be analysed and taken into consideration in developing a voiced/unvoiced/noise detection system such as, the complexity of the algorithm, the delay introduced (and thus the duration of the analysis window in which the decision is made), robustness to noise (which is mainly channel and/or background noise), the overall performance of the system, the phonetic classes to be considered (silence/background noise, mixed sounds, etc.), and the training and testing database used to design and test the algorithm (in particular the duration, the number of different speakers, the number of languages, the types of digitally added noise, the sampling frequency, etc.).

The voiced/unvoiced/silence detection and classification can be made using temporal and spectral analysis of the speech signal. Spectral Analysis of speech segment and pattern extraction of the segment is proposed to have important information on distinction of noise vs. speech waveform, consonant vs. vowels waveform, voiced vs. unvoiced speech waveform and on place and manner of articulation. A sample annotated speech waveform and its spectrogram[1] representing the Amharic version of the sentence “Abebe has come (አበበ መጣ)” spoken by an adult male is shown in Figure 1.1.

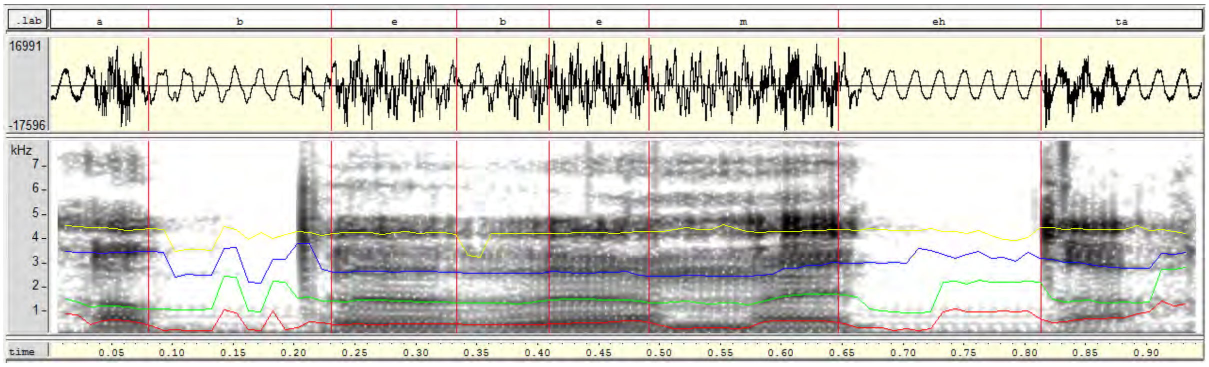


Figure 1.1: Speech Waveform (upper) and spectrogram (bottom) of the sentence “አበበ መጣ”

[1] A graphical display of sound in which time is on the horizontal axis and frequency is on the vertical axis. Intensity is shown by a grey-scale representation (the darker the display, the higher the intensity) or by a color display (the brighter the color, the hi gher the intensity). [15]

It is obvious that users are in need of using information technology in such a way that it can simplify their life. This striking interest of users brings to the idea of speech processing. Speech processing, specifically speech signal extraction and classification into voiced, unvoiced and noise, provides a preliminary acoustic segmentation for audio signal processing applications, such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, voice activity detection, speaker identification, and recognition of speech pathologies.

1.2 Statement of the Problem

Correct voiced, unvoiced, noise classification of a sound provides a preliminary acoustic segmentation of speech, which is important for several speech processing systems. The performance of such systems, such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, speaker identification, and the recognition of speech pathologies depends on the ability of the system to detect correct voiced/unvoiced/silence speech segments.

In spite of this, the absence of good pattern extraction of voiced/unvoiced/silence speech segment detection system limits the performance of speech processing systems that require the use of it. Most approaches developed so far for voiced/unvoiced/silence speech segment detection mainly used waveform temporal features in order to classify a segment of speech as voiced, unvoiced or silence. This is mainly because, all voiced speech segments are produced by the quasi-periodic vibration of the vocal cords and unvoiced & silences do not which can be easily measured in time representation of the signal. However, this deteriorates the performance of the system as the waveform features overlap between categories and are susceptible for incorrect classification with decreasing signal to noise ratio (SNR). This in turn drops the performance of the speech processing system. In addition, the performance of voiced/unvoiced/silence classification systems developed so far is not up to the standard i.e. their performance is less that doesn‟t meet the requirements of systems that require the use of it. For example artificial neural network (ANN) based automatic speech recognition (ASR) is believed to outperform HMM based approach, if such segmentation is in place for ANN systems as ANN is superior in phone recognition over HMM. Moreover, to the best of the researcher‟s knowledge, there is no voiced/unvoiced/silence speech segment detection system developed so far that considers Amharic language speech.

1.3 Objective of the Study

1.3.1 General Objective

The general objective of this research work is to conduct a research on voiced/unvoiced/silence region of speech signal segment using machine learning techniques from the acoustic feature.

1.3.2 Specific Objectives

So as to achieve the above general objective, the research will accomplish the following specific objectives:

- To build speech corpus for Amharic sentences.
- To extract acoustic features from the speech signal.
- To determine appropriate features for voicing detection.
- To determine appropriate frame size for voicing detection.
- To build an appropriate voicing extraction and detection model using the training data.
- To implement a prototype of the voicing detection.
- To test and analyse the system performance on the testing data.
- To derive conclusion and recommendation for future work.

1.4 Methodology

In this study, we first thoroughly studied the speech sound categories (voiced, unvoiced and silence) and approaches to voiced/unvoiced/silence speech segment detection in parallel with a survey of relevant literatures. The researchers have also assessed and conducted literature review on voiced/unvoiced/silence extraction and detection developed so far for different languages.

1.4.1 Data Collection

A speech corpus is one of the fundamental requirements for any voiced/unvoiced/silence speech segment detection related researches. Speech corpus is a collection of speech recordings which is accessible in computer readable form, and which has an annotation and documentation sufficient to allow re-use of data. Standard speech corpora consist of a training set and evaluation test sets. The training set is intended to collect speech data for training the model and the evaluation test set is for the purpose of final evaluation of the voiced/unvoiced/silence model.

For this study, speech corpus is collected from different sources to study the spectral and temporal characteristics of the speech signal towards voiced/unvoiced/silence speech segment detection. The selection of sentences aims at both a phonetically rich and balanced collection of sentences. The sentences are selected from different Amharic sources such as political news, economy news, sport news, and health news which contribute to the inclusion of all Amharic phones. A total of 900 sentences are collected. The collected sentences are recorded by one male speaker under normal environment.

1.4.2 Data Preparation

The prepared sentences are digitized using audio recording software with the sampling frequency of 48 KHz, sample size of 16 bits and Mono channel. The recording process is carried out in a quite resident environment using microphone. The training and test data are recorded in the same environment to normalize the effect of noise.

The digitized signal is windowed with various size rectangular window and is processed frame by frame. In the frame-by-frame processing stage, the speech signal is segmented into a non-overlapping frame of samples each of which having the same window size. The signal is processed into frame by frame until the entire speech signal is covered. A feature vector was obtained from each frame of speech and 20, 25, 30 and 35 ms of frame size is used. The feature vector was a combination of 13 cepstral coefficients and two waveform temporal features, the zero-crossing rate and energy. The cepstral coefficients were derived from 12 pole LPC analysis and 13 MFCCs. The autocorrelation method of autoregressive (AR) modelling is used to find the LP coefficients.

The whole feature vector obtained from the speech signal is split randomly into two: 66.67% for training and the rest 33.33% for testing the system.

1.4.3 Modeling

The study used artificial neural network (ANN) approach which consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. The neural network approach is chosen as it offers numerous advantages to develop an efficient voiced/unvoiced/silence detection algorithm [24].

Based on the assessment and study of the speech sounds made, voiced/unvoiced/silence extraction and detection algorithm that is suitable and efficient is developed.

1.4.4 Tools and Implementation

In conducting the research, extraction and detection of voiced/unvoiced/silence speech signal using the information in the frequency domain, Matlab 7.9 specifically matlab signal processing toolbox and speech signal processing methods (functions) are used for implementing the model. Wavesurfer and Audicity are used as a major development tool to record and prepare the frequency domain representation of the speech signals for acoustic analysis. Weka 3.6 machine learning tool is used for machine learning analysis and experiment.

The rationale behind the choice of these tools is that they are suitable for performing different speech signal processing and analysis tasks and machine learning processes.

1.4.5 Experimental Analysis

As stated in Section 1.4.2, the collected feature vector is randomly divided into two: the training data set which is used to train the model prototype and test data set which is used to test the model.

After the voiced/unvoiced/silence model is developed, testing is conducted on the model. The model is trained using the speech data selected for training. The performance of the model on each category of speech is also evaluated. Accuracy is taken as the performance measure of the model. Accuracy is the closeness of the agreement between the test result and the accepted reference value (the manually tagged speech). Then, performance evaluation is made through Weka machine learning tool by counting the number of the correct voiced/unvoiced/silence detection that the system has made by comparing against hand categorized counterpart which in turn can be expressed in percentage. This activity is conducted by repetitive experiments on the sample training and test sets until satisfactory results are obtained.

1.5 Application of Results and Beneficiaries

As specified in Section 1.1, there are many advantages of developing voicing detection algorithm. In the first place, it is the basis for developing other higher level and efficient applications of speech processing systems such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, voice activity detection, speaker identification, and the recognition of speech pathologies. In view of that, the beneficiaries of this study will be:

- Researchers who want to conduct research on the above application areas.
- Linguistic professionals who want to conduct research on sound speech categories and their extraction and detection.
- Linguistic professionals who want to learn and use glottal activity detection (voicing activity detection) of speech sound.

1.6 Scope of the Study

The study will do voiced/unvoiced/silence speech segment detection on a speech data collected in a normal environment from one speaker on Amharic language speeches. The system may not show the same performance on speech data collected in a different environment for different speaker and for other languages other than Amharic.

1.7 Organization of the Thesis

The whole thesis is organized into six chapters. The first chapter describes the introductory part. The second chapter is all about literature review and related works. It describes the approaches used so far for voicing detection and works that are done using rule-based, statistical and neural network approach. It also focuses on study and assessment of the nature and structure of speech production and articulation. After the thorough study of the characteristics of speech sounds, chapter three deals with windowing and feature extraction. Afterwards, the fourth chapter deals with architecture and design of the voiced/unvoiced/silence classifier, corpus preparation and implementation of the pre-processing tools and algorithms for the architecture. Chapter five mainly focuses on the experimental analysis of the voiced/unvoiced/silence classifier conducted for this study. Finally, the last chapter is all about drawing conclusion that comprises both summary of the work done and recommendation on future work.

CHAPTER TWO

LERATURE REVIEW AND RELATED WORKS

2.1 Human Speech Production

In the production of different speech sounds, different organs (parts of the human body that are engaged in various ways with the production of speech) are involved. The vocal tract is the path through the human vocal organs that produce speech. The principal organs of speech include: lips, teeth, alveolar ridge, palate, velum, uvula, tongue, mouth cavity, nasal cavity, pharynx, epiglottis, oesophagus, glottis, and larynx. The particular acoustic sound that is created is dependent on the action and position of the vocal organs. The vocal organs shape the frequency characteristics of the vibrating air travelling through the vocal tract.

These organs are classified into two categories: *active* and *passive* articulators. The active articulators usually move or actively involve making the constriction when producing a sound. These include velum, jaw, tongue blade, tongue tip, tongue body, tongue root and lips. On the other hand, the passive ones usually sit there and get approached by the active organs. These are alveolar ridge, hard palate, teeth and uvula.

Figure 2.1 displays a simplified schematic of the primary vocal operators of the vocal tract. The diaphragm expands and contracts assisting the lungs in forcing air through the trachea, across the vocal cords and finally into the nasal and oral cavities. The air flows across the tongue, lips, and teeth and out the nostrils and the mouth. The glottis (opening formed by vocal cords or vocal folds) can allow the air from the lungs to pass relatively unimpeded or can break the flow into periodic pulses. The velum can be raised or lowered to block passage, or allow acoustic coupling, of the nasal cavity. The tongue and lips, in conjunction with the lower jaw, act to provide varying degrees of constriction at different locations. The tongue, lips, and jaw are grouped under the title *articulators*, and a particular configuration is called an *articulatory position* or *articulatory gesture*.

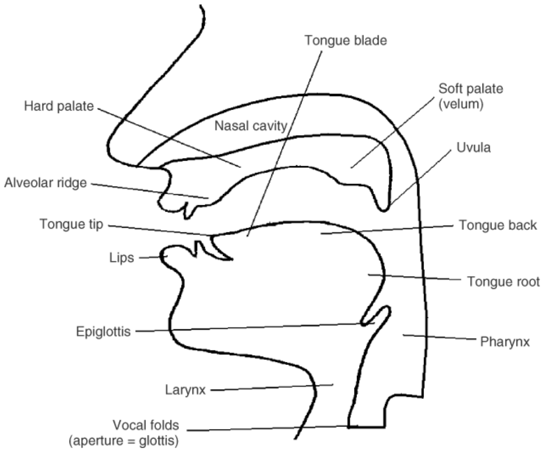


Figure 2.1: Primary articulators of the vocal tract

2.1.1 Excitation

Speech sounds are produced as air is pushed from the lungs and converted into fluctuating energy. Air from the lungs goes up the windpipe (the trachea) and into the larynx, at which point it must pass between two small muscular folds called the vocal folds. This air source and the nature of its flow are referred to as the *excitation signal.* The excitation can be considered to be voiced or unvoiced. If the vocal folds are apart, as they normally are when breathing out, the air from the lungs will have a relatively free passage into the pharynx and the mouth. But if the vocal folds are adjusted so that there is only a narrow passage between them, the air stream will cause them to vibrate. Sounds produced when the vocal folds are vibrating are said to be voiced, as opposed to those in which the vocal folds are apart, which are said to be unvoiced/voiceless [19]. All vowel phonemes are classified as voiced where as consonant phonemes can be classified as either voiced or unvoiced. Some voiced consonant phonemes include /b/, /d/, /g/, /v/, /z/. Each of these obstruents has an unvoiced counterpart, /p/, /t/, /k/, /f/, /s/ [20].

The two types of excitation (voiced or unvoiced) are produced by different mechanisms at different places in the vocal tract. In view of this fact, it is also possible to have both present at once in a mixed excitation. The simultaneously periodic and noisy aspect of the sound /z/ is one example. How to classify such a sound depends on the viewpoint: from a phonetic view, the sound /z/ has a periodic excitation, so it is considered to be voiced. But, from the viewpoint of wanting to represent that sound in a speech coder, both the periodic and noisy attributes are present and perceptually significant, hence the *mixed* labelling.

2.1.2 Vocal Tract

Given the excitation as either voiced or unvoiced, the shape of the vocal tract, and how it changes shape over time, is another primary determinant of a particular speech sound. In order to form consonants, the air stream through the vocal tract must be obstructed in some way. Consonants can be classified according to the place and manner of articulation.

The vocal tract has specific natural frequencies of vibration like all fluid filled tubes. These resonant frequencies, or resonances, change when the shape and position of the vocal articulators change. The resonances of the vocal tract shape the energy distribution across the frequency range of the speech sound. These resonances produce peaks in the spectrum that are located at specific frequencies for a particular physical vocal tract shape. The resonances are referred to as *formants* and their frequency locations as the *formant frequencies*.

2.1.2.1 Place of Articulation

Place of articulation, also called point or area of articulation, refers to the different places of articulation where possible human speech sounds are articulated. The primary articulators that can cause an obstruction in most languages are the lips, the tongue tip and blade, and the back of the tongue. Speech gestures using the lips are called ***labial*** articulations; those using the tip and blade of the tongues are called ***coronal*** articulations; and those using the back of the tongue are called ***dorsal*** articulation [19]. For example, the Amharic word “ጠባቂ” (guard) begins with a coronal consonant; in the middle there is a labial consonant; and at the end a dorsal consonant.

These terms, however, do not specify articulatory gestures in sufficient details for many phonetic purposes. It is possible to identify the place of articulation by specifying its distance along the length of the vocal tract from the glottis using acoustically oriented investigations.

Places of articulation are identified by the main zones of the vocal tract that are involved in the articulation of sounds. And most sounds are named after the place where they are formed. The following discussion is made on the different kinds of sounds according to their names.

A. Bilabials

Bilabials, bi means two and labial means lips, are sounds that are produced when the upper and lower lips come close. Some consonantal sounds like /p/, /b/ and /m/ are produced when the lower lip articulates against the upper lip. The lips are not important only in the production of consonant sounds, but also when describing vowels. In the production of speech sounds, the lips help in forming different vowels, partly by modifying the form of the mouth opening. When producing bilabial sounds, we could say that the lower lip is the active articulator which moves up where as the upper lip though it moves, at least a little, is considered as passive.

B. Labiodentals

Labiodental sounds are produced when the lower lip comes against the upper teeth. These sounds include the voiceless /f/ and its voiced counterpart /v/. In such kinds of sounds, the lower lip is the active articulator while the upper teeth are passive.

C. Dentals

Dentals are sounds formed when the tip of the tongue articulates behind the upper front teeth. The first sounds of the Amharic words “**ተው**” (**stop**) and “ደወለ” (he called) are dentals. These sounds also referred to as inter-dentals because they are produced when the tip of the tongue comes between the upper and the lower teeth (they are also called lamino-dental sounds). These sounds involve the upper teeth as the passive articulator. The active articulator may be either the tip of the tongue or the blade of the tongue.

D. Alveolars

These sounds are produced when the tip and/or blade of the tongue comes in contact with the alveolar ridge, which is the rough, bony ridge immediately behind and above the upper teeth. The initial sounds in the Amharic words “ለመነ” (he begged), “ዘፈነ” (he sang) and “ሰመረ” (accomplished) are all alveolars. English alveolar sounds include /t/, /d/, /s/, /z/ and /n/. Among these sounds, /t/ and /s/ are voiceless sounds while the rest are voiced. There are also other English alveolar sounds like /l/ and /r/, both voiced. The initial sounds of the following words lip, lion, right and room are alveolars too. In the production of these sounds, the alveolar ridge involves as a passive articulator while either the blade of the tongue or (usually) the tip of the tongue participates as an active articulator.

E. Post-alveolars

These are sounds that are formed when the tongue front articulates immediately behind the teeth ridge. They are also referred to as palato-alveolars. These sounds include /ʃ/ (sh/s/ti as in **sh**are, **s**ure, emo**ti**on respectively) and /ʒ/ (s/ge/z as in mea**s**ure, bei**g**e, sei**z**ure, respectively). These sounds involve the area behind the alveolar ridge as the passive articulator whereas the active articulator is usually the blade or the tip of the tongue.

F. Retroflex

These sounds are produced when the tip of the tongue is curled back past the alveolar ridge. Many speakers of English do not use retroflex sounds at all [19]. Some examples of this group of sounds occur initially in words such as "rye" and "row". Speakers who pronounce /r/ at the ends of words may also have retroflex sounds with the tip of the tongue raised in "ire", "hour" and "air". In retroflex, the tip of the tongue is the active articulator whereas the alveolar ridge is passive. Other common retroflex consonants are the retroflexed correlates of /t/ and /d/ – /ʈ/ and /ɖ/ – heard in many Indian languages, such as Hindi, and also in the English spoken by native-speakers of such languages [7].

G. Palatals

Palatals are sounds made when the front of the tongue articulates against the hard palate. The more we feel back behind the alveolar ridge, we find a hard part in the roof of our mouth. This part is called the hard palate or just the palate. In other words, sounds which are produced with the tongue and the palate are called palatals or alveo-palatals. The first sounds of the words **ch**urch and **j**udge /c/ and /ɟ/ respectively are palatals. In the articulation of palatal sounds, the active articulator is the body of the tongue and the passive one is the hard palate.

H. Velars

Further back in the roof of the mouth, beyond the hard palate, we find a soft area called soft palate or the velum. Sounds that are articulated when the back of the tongue comes close against the soft palate are called velars. The first sounds of the words car and gate /k/ and /g/ are velar sounds respectively. The velum can also be lowered to allow the air flow through the nasal cavity and thereby produce a velar sound which is represented by the IPA symbol /ŋ/, typically referred to as „angma‟. The active articulator in the generation of velar sounds is the body of the tongue and the passive one is the soft palate or the velum.

I. Uvulars

Uvular sounds are made by the back of the tongue against the uvula – the fleshy appendage which hangs at the back of the soft palate. In other words, uvulars are sounds made when the back of the tongue which is an active articulator and the uvula, a passive articulator, form a constriction or a closure.

As stated in [7], the /r/ of "standard" French is uvular, and this quality may be heard in some regional dialects of English, especially in the north-east of England. Uvular plosive consonants are found in Arabic, for example, and are transcribed /q/ and /G/ for the voiceless and voiced types respectively.

J. Pharyngeal

Pharyngeals are sounds that are articulated when constriction is formed in the pharynx, the tubular cavity which constitutes the throat above the larynx. Pharyngeal consonants occur in languages such as Arabic and Hebrew. They do not occur as speech sounds in English, but similar effects can be heard in stage whispers.

K. Glottals

Glottal sounds are produced in the larynx due to the closure or narrowing of the glottis, the aperture between the vocal folds. They when the glottis is open, as in the production of other voiceless sounds, and there is no manipulation of the air passing out of the mouth.

2.1.2.2 Manner of Articulation

In the description of speech sounds, besides the places of articulation, manner of articulation is also considered. It is all about the way how the speech sounds are uttered. Manner of articulation also refers to degree of constriction, the degree how close the organs are. This parameter helps us to further make a distinction among sounds formed in the same place of articulation. For instance, /t/ and /s/ are both alveolars, but they differ in their manner of articulation, that is, in the way they are pronounced.

To produce vastly different speech sounds, the excitation is altered by different general categories of the vocal tract configurations. For example, vowel sounds are produced by periodic excitation, and the airflow passes through the vocal tract mostly unrestricted. This open, but not uniform, configuration produces the resonances associated with the formant frequencies. In a loose analogy, this is similar to the resonances produced by blowing across an open tube. The following is a detailed analysis of classes of speech sounds by manner of articulation.

A. Plosives/Stops

Plosives, also called stops, result from the sudden release of an increased air pressure due to a complete restriction of airflow. This is the case where the active articulator touches the passive articulator and completely cuts off the air flow through the mouth. Ladefoged [19] identified two types of stops: **oral** and **nasal**. Oral stops are stops formed when the soft palate is raised so that the nasal tract is blocked off, then the air stream will be completely obstructed. Pressure in the mouth will build up and an oral stop will be formed. When the articulators come apart, the air stream will be released in a small burst of sound. This kind of sound occurs in the consonants in the words "play, buy" (bilabial closure), "time, dance" (alveolar closure), and "kick, garage" (velar closure). The second domain treats nasal stops. If the air is stopped in the oral cavity, but the soft palate is lowered so that it can go out through the nose. We can find sounds of this kind at the beginning of the words “my” (bilabial closure), “nose” (alveolar closure) and at the end of the word “hang” (velar closure).

Stops can be voiced such as sound /b/ or unvoiced like the /p/ sound. The difference between the unvoiced plosive phonemes and the voiced plosive phonemes is not just a matter of whether (articulatory) voicing is present or not. Rather, it includes when voicing starts (if at all), the presence of aspiration (airflow burst following the release of the closure), and the duration of the closure and of the aspiration.

B. Fricatives

Sometimes called spirants, are sounds made when two organs come so close together that the air moving between them produces audible friction. Fricatives have no periodic component and are the result of a steady airflow meeting some constriction. Examples of fricatives are /s/ and /f/. According to Ladefoged [19], the mechanism in the production of such sounds seems wind whistling around a corner. The first consonants in fine, viva (labiodentals), this, that (dentals/inter-dentals), size, zoo (alveolars), and shy (postalveolar) are examples of fricative sounds.

C. Affricates

Affricates can be seen as a sequence of a stop and a fricative which have the same places of articulation. In other words, it is a case where a single articulator effects first plosive, then a fricative, articulation at the same, or a close, place; the soft palate is raised. The first sounds in the words church, judge transcribed as /tʃ/ and /dʒ/ are affricates.

D. Laterals(Approximants)

Such kinds of sounds are formed when an active articulator effect a partial closure in the mouth by allowing its sides to be free of any contact; the velum is raised. They also referred to as lateral approximants. For example, say the word “lie” and note how the tongue touches near the centre of the alveolar ridge. Prolong the initial consonant and note how, despite the closure formed by the tongue, air flows out freely, over the side of the tongue. Because there is no stoppage of the air, and not even any fricative noises, these sounds are classified as approximants.

E. Nasal

Nasal consonants are produced by lowering the velum so that air can flow through the nasal cavity. At the same time, a complete constriction in the mouth prevents airflow through the lips. The most common nasal examples are /m/, /n/ and /ng/.

F. Sonorants

Other English sounds, the vowels, nasals /m/, /n/, /ng/, and approximants /l/, /r/, /w/, /j/ (the last spelled as the English letter /y/) are called *sonorants*.

2.1.3 Acoustic Cues

There are a number of acoustic cues of a speech sound used by a listener to correctly perceive the underlying phoneme.

Vowels: Formant frequencies have been determined to be a primary factor in identifying a vowel. The listening experiments of Peterson and Barney approximately map the first two formants (F1 and F2) to vowel identification. Higher formants also have a role in vowel identity. Another factor in vowel perception is nasalization, which is cued primarily by the bandwidth increase of the first formant (F1) and the introduction of zeros.

Consonants: Consonant identification depends on a number of factors including the formants of the consonant, formant transitions into the formants of the following vowel, the voicing (or unvoicing) of the vocal folds during or near the consonant production, and the relative timing of the consonant and the onset of the following vowel.

Every articulatory movement and posture has its own acoustic effects. Voiced sounds are essentially produced due to the vibrations of the vocal cords, and tend to be louder and are oscillatory. Unvoiced sounds tend to be more abrupt like and are more noise-like.

2.2 Approaches to Voicing Detection

So far many voicing detection researches have been done and different approaches have been used for voiced/unvoiced/silence classification, where the well-known ones are rule-based, statistical and neural network approach. All of the proposed methods have their merits, and preference for one over another.

In this chapter, three different models were examined namely the rule-based, statistical and neural network. The rule based approach as its name indicates relies on rules which are either handcrafted or machine learned rules. The rules are the important elements for annotating speech segments in the rule-based approach. The statistical based detector relies on the statistical property of speech signals. Such statistical property can be distributional probability of speech signals with tags which can be obtained during the training phase of the system. Finally, the neural network (NN) detectors use acoustic features of speech signals that can be extracted from the speech waveform or spectrogram to classify speech categories.

This section mainly deals with the most common approaches to voicing detection and related woks to this thesis work. Accordingly the approaches used so far are described in detail in the following sections.

2.2.1 Rule-Based Approach

The rule based approach uses rules to identify the regions of speech when there is significant glottal activity (i.e., the vibration of vocal folds). Such regions of speech are generally referred to as voiced speech. The unvoiced regions of speech may include both silence as well as unvoiced speech. The rules are based on knowledge of specific information (property) extracted from the speech segment and setting threshold values to identify/detect regions of speech as voiced/unvoiced/silence. The rule-based approach as its name indicates relies on rules which are either handcrafted or machine learned rules. The specific information (properties) used to setup the rules can be obtained using two basic methods that extract useful information from the analysis of variations across the sequences of the speech segments used: *time-domain methods* and *frequency-domain methods*.

The time-domain and frequency-domain methods for rule-based approach measure one or more acoustic features which reflect the production characteristics of the sounds such as energy, periodicity, short-term correlation, cepstrum and an LPC distance. Voiced/Unvoiced/Silence decisions are taken by setting thresholds on individual parameter values (chosen empirically), and the decisions are combined in a hierarchical manner.

In [3], a rule-based method was developed for classifying the speech into voiced/unvoiced using zero-crossing rate and energy of a speech signal. In this work, two acoustic features were combined: zero crossings rate and energy of the signal. The zero crossing count is an indicator of the frequency at which the energy is concentrated in the signal spectrum. Voiced speech is produced because of excitation of vocal tract by the periodic flow of air at the glottis and usually shows a low zero-crossing count, whereas the unvoiced speech is produced by the constriction of the vocal tract narrow enough to cause turbulent airflow which results in noise and shows high zero-crossing count. Energy of a speech is another parameter used for classifying the voiced/unvoiced parts. The voiced part of the speech has high energy because of its periodicity and the unvoiced part of speech has low energy.

A speech threshold is determined which takes into account the silence energy and the peak energy. Initially, the endpoints are assumed to occur where the signal energy crosses this threshold. Corrections to these initial estimates are made by computing the zero-crossing rate in the vicinity of the endpoints and by comparing it with that of the silence. If detectable changes in zero-crossing rate occur outside the initial thresholds, the endpoints are re-designed to the points at which the changes take place. If the zero-crossing rate is high and the energy is low, the speech signal is unvoiced, while if the zero-crossing rate is low and the energy is high, the speech signal is voiced.

2.2.2 Statistical Approach

Statistical models such as Gaussian mixture models (GMM) or Hidden Markov Models (HMM) are also used for combining evidence from multiple features to detect voiced/unvoiced/silence speech segment. These methods do not depend critically on threshold setting, but require training data for the different types of speech segment. Statistical approaches assume different models of random process for speech and background noise, and estimate the parameters of the underlying distributions. Performance of these approaches depends on the choice of the probability distributions, and the ability to estimate the parameters of the noise distribution [9].

The algorithms which do not use thresholds could be an improvement over algorithms which use thresholds for the detection of voiced/unvoiced/silence speech segments. Such demands satisfy the usage of statistical classification algorithms.

The most commonly used statistical method for voicing detection is the Hidden Markov Model (HMM). It is the probabilistic function of Markov Process, a process which moves from state to state, from left to right on the states, to find optimal state sequence. An HMM is characterized by the following criteria [12]:

- A finite set of states each of which is associated with a probability distribution
- Transitions among the states are governed by a set of probabilities called transition probabilities.
- In a particular state an outcome or observation can be generated according to the associated probability distribution. The observation is visible and the states are hidden to the observer and hence the name Hidden Markov Model.

There are two approaches to voiced/unvoiced/silence speech segment detection using HMM [31]. The first one is based on separated training of HMM and following testing. After training the HMMs will not alter its parameters and the model will be insensitive to changes of environment - detection without adaption. The second approach combines steps of training and testing - detection with adaption.

When the HMM model is taken to the application of voicing detection, the hidden states are the voiced/unvoiced/silence categories and the sequences of speech frames are the sequence of observations. The transition probability in voicing detection is the probability of moving from one category to the next and the emission probability is the probability of getting a speech category Ci being in frame Fi. The algorithm can be trained using the Expectation Maximization (EM) algorithm or the Baum-Welch algorithm.

To evaluate how the voiced/unvoiced/silence speech segment detection model designed by this method performs, we need to present several types of input signals and monitor the matching between the detected and actual speech segment category. In an HMM method, we don‟t need any assumptions about the speech category. The characteristics of the voiced, unvoiced and silence speech segments will already be implied within the model itself during the training procedure.

2.2.3 Neural Network Approach

Today, neural networks are used to solve a wide variety of problems. These applications, to which a neural network approach can be applied to, fall into one of the following three categories:

- Forecasting: predicting one or more values of dependent variable(s) from independent variable(s) input values,
- Classification: classifying input data into one of two or more categories, or
- Clustering: uncovering patterns, typically spatial or temporal, among a set of variables.

Neural Networks (NN) are self-organizing pattern matchers, providing a classification output for messy or fuzzy input. Logically, they consist of a collection of nodes, connected by links with associated weights. At each node, signals from all incoming links are summed according to the weights of these links, and if the sum satisfies a certain transfer function, an impulse is sent to other nodes through output links. Neural network learning is a type of supervised learning, meaning that we provide the network with example inputs and the correct answer for that input.

In [24], a procedure is developed for making the voiced/unvoiced/silence classification using a multilayer feed-forward network (MFN). An MFN is an interconnection of layers in which data and calculations flow in a single direction, from the input layer to the output layer. A perceptron is a neural network that has no hidden units. The number of layers in a neural network is the number of layers of perceptrons. The simplest neural network is one with a single input layer and an output layer of perceptrons. The network in Figure 2.2 illustrates this type of network. Technically, this is referred to as a one layer feed-forward network because the output layer is the only layer with an activation calculation.

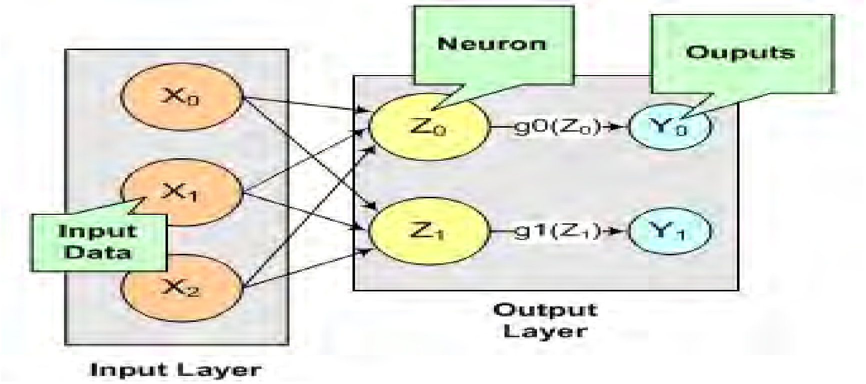


Figure 2.2: A Single-Layer Feedforward Neural Network

The next most complicated neural network is one with two layers. This extra layer is referred to as a hidden layer. In general there is no restriction on the number of hidden layers.

The feature vector for the classification of voiced/unvoiced/silence speech segment using neural network can be combination of cepstral coefficients and waveform features [24]. The voiced/unvoiced/silence classification can be for each input feature vector after training was completed.

The unique advantage of an MFN is that the decision rule can be much more easily synthesized than both parametric and non-parametric methods. The network implementation of the classifier also promotes the perspective of building voiced/unvoiced/silence classification hardware with adaptive training mechanism [24].

A connectionist model for voiced/unvoiced/silence speech segment detection might take as input a set of spectral partials, or the time-domain waveform, or the phase space representation of the signal. The choice of the dimensionality and domain of the input set is crucial to the success of any connectionist model. A problem with connectionist models is that even if a good model is found, it does not provide any understanding of how the problem is solved.

2.3 Related Works

In recent years considerable efforts has been spent by researchers in solving the problem of classifying speech into voiced/unvoiced/silence parts [3, 8, 9, 24, 26, 31]. A pattern recognition approach and statistical and non statistical techniques has been applied for deciding whether the given segment of a speech signal should be classified as voiced, unvoiced or silence.

In this section, different works on voiced, unvoiced and silence speech segment detection are presented starting from the works of Atal and Rabiner followed by the work of Qi and Hunt and others. The order of presentation of the works in this section is selected based on the similarities of the works to this thesis work.

Atal and Rabiner [2], describe a pattern recognition approach for deciding whether a given segment of speech signal should be classified as voiced speech, unvoiced speech, or silence, based on measurements made on the signal. The researchers have made five different measurements on the speech segment to be classified. The measured parameters are the zero-crossing rate, the speech energy, the correlation between adjacent speech samples, the first predictor coefficient from a 12-pole linear predictive coding (LPC) analysis, and the energy in the prediction error.

The researchers [2] stated that the pattern recognition approach provides an effective method of combining the contributions of a number of speech measurements. They stated it as follows:

> *The pattern recognition approach provides an effective method of combining the contributions of a number of speech measurements-which individually may not be sufficient to discriminate between the classes-into a single measure capable of providing reliable separation between the three classes.*

In this work [2], for each of the three classes, a non-Euclidean distance measure is computed from a set of measurements made on the speech segment to be classified and the segment is assigned to the class with the minimum distance. As stated in the paper, the distance function is chosen so as to provide minimum classification error for normally distributed measurements. As stated in this work, results based on the computed one-dimensional distributions of the chosen measurements suggest that the assumption of normal distribution is a reasonable one.

As depicted in Figure 2.3, the speech signal is low-pass filtered to 4 kHz, sampled at 10 kHz, and each sample is quantized with an accuracy of 12 bits. Prior to analysis, the speech signal is high-pass filtered at approximately 200 Hz to remove any dc, low-frequency hum, or noise components which might be present in the speech signal. Following high-pass filtering, the speech is formatted into blocks of 100 samples (an interval of 10 ms at 10 kHz sampling frequency); with each block spaced 100 samples apart. For each block, the researchers defined s(n), n = 1, 2, …, N, to be the nth sample in the block. The samples N, N-1, N-2, etc. of the previous block are numbered 0, -1,-2, etc. Thus s(0) is the last sample of the previous block.

The speech segment is assigned to a particular class based on a minimum distance rule obtained under the assumption that the measured parameters are distributed according to the multidimensional Gaussian probability density function. The means and covariance for the Gaussian distribution are determined from manually classified speech data included in a training set.

The researchers have tested the performance of the method on a wide variety of speech material for both speech synthesis and segmentation applications. The researchers [2] stated that the voiced/unvoiced/silence decision has performed satisfactorily. They stated it as follows:

> *The voiced/unvoiced/silence decision has performed satisfactorily as a part of a speech analysis-synthesis system based on linear prediction. The method has been found to provide reliable classification with speech segments as short as 10 ms and has been used for both speech analysis-synthesis and recognition application*

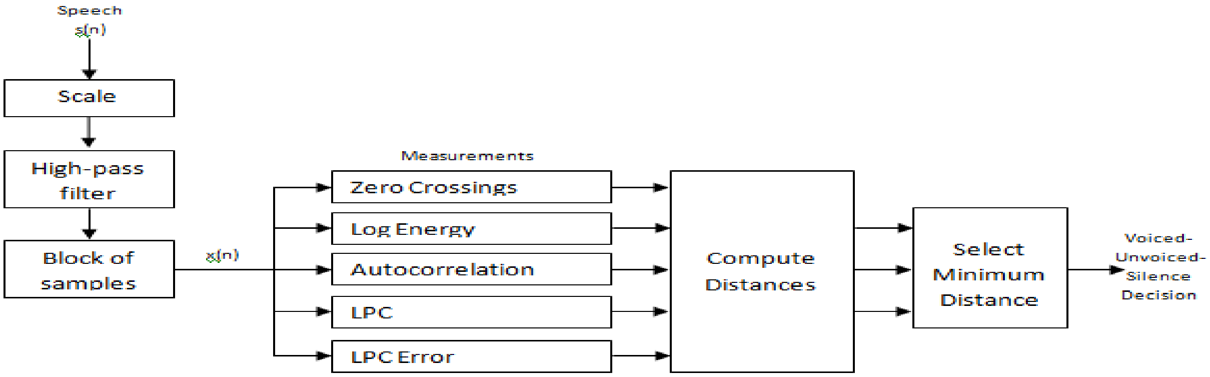


Figure 2.3: Block diagram of the analysis system

Another work for the discrimination of voiced/unvoiced/silence speech sounds is developed by Qi and Hunt using neural network approach. In this paper [24], the researchers describe a neural network approach for deciding whether a given segment of a speech signal should be classified as voiced speech, unvoiced speech, or silence, based on measurements made on the signal.

The basic idea of the researchers, in the proposed voicing detection, is to develop a procedure for making the voiced/unvoiced/silence classification using a multilayer feed-forward network (MFN). The feature vector used for the classification is a combination of cepstral coefficients and waveform features.

A block diagram of the network training and classification process is shown in Figure 2.4. as shown in the figure the speech signals were low-pass filtered at 4.5 kHz, sampled at 10 kHz, and quantized with 16-bit accuracy. The digitized signals were further high-pass filtered at 300 Hz by a fourth-order Butterworth digital filter to eliminate low-frequency hum or noise. A feature vector was obtained for each 20 ms segment of speech. The feature vector was a combination of 13 cepstral coefficients and two waveform parameters, the zero-crossing rate and a nonlinear function of root mean square (RMS) energy. The cepstral coefficients were derived from 12 LP coefficients and the energy of prediction error. The autocorrelation method, hamming window, and pre-emphasis (0.98) were used in calculating the LP coefficients. An inverse square root function was applied to the RMS energy to limit its numerical range.

The voiced/unvoiced/silence classification was made for each input feature vector after training was completed. The classification output was further decoded and passed through a three-point median filter to eliminate isolated "impulse" noise. The network was trained using the generalized delta rule for back propagation of error with a learning rate of α= 0.9. The network performance as a function of the size of training set and signal to noise ratio was also evaluated and compared to a Bayesian, maximum-likelihood (ML) classifier.

The performance of the network is evaluated using speech samples provided for six speakers (3 men and 3 women). The researchers [24] stated that the voiced/unvoiced/silence decision can be accomplished effectively using multilayer feed-forward network. They stated it as follows:

> *The results of the study demonstrates that the voiced, unvoiced, and silence classification can be effectively accomplished using a multilayer feed-forward network and hybrid features.*

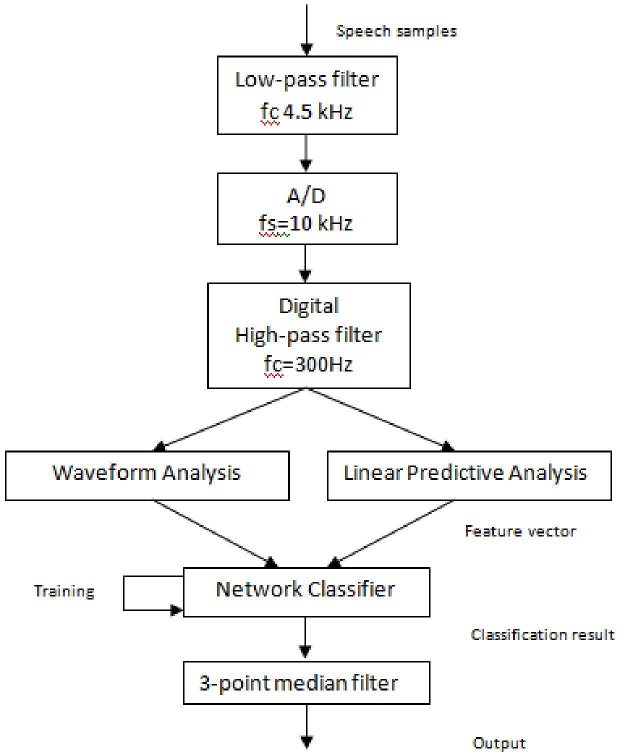


Figure 2.4: Flow chart of network training and classification processes

Another work [3], which has used zero-crossing rate and energy measurements made on the signal using rule based approach, was done for making voiced/unvoiced decision for speech signals. In here, the researchers evaluated the results by dividing the speech sample into some segments and used the zero crossing rate and energy calculations to separate the voiced and unvoiced parts of speech.

The overall voiced/unvoiced classification process is depicted on Figure 2.5. At the first stage, the speech signal is divided into intervals in frame by frame without overlapping. In the frame-by-frame processing stage, the speech signal is segmented into a non-overlapping frame of samples. It is processed into frame by frame until the entire speech signal is covered. And then, Short-time energy (E) and short-time average zero-crossing rate (ZCR) is calculated for each frame and a speech threshold is determined to categorize a frame as voiced or unvoiced. At the beginning, the frame size is set to 400 samples (50 ms) for a sampling rate of 8000Hz. At the end of the algorithm, if the ZCR is small and E is high, the speech is categorized as voiced otherwise it is said unvoiced but if decision is not clear (Not sure), energy and zero-crossing rate is recalculated by dividing the related frame size into two frames.

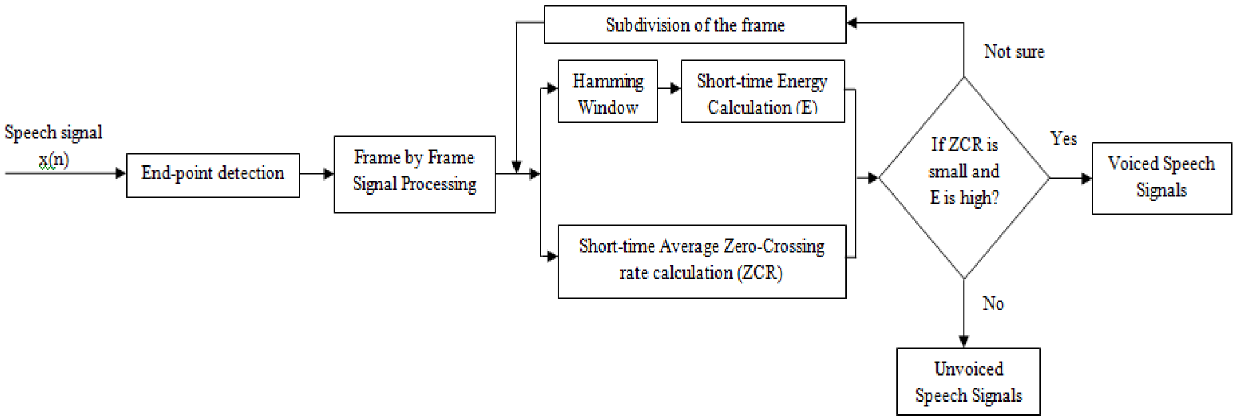


Figure 2.5: Block diagram of the Voiced/Unvoiced Classification

In this work, end-point detection is applied to the voiced/unvoiced algorithm at the beginning of the algorithm to separate silence and speech signal. A small sample of the background noise is taken during the silence interval just prior to the commencement of the speech signal. The short-time energy function of the entire utterance is then computed. A speech threshold is determined which takes into account the silence energy and the peak energy. Initially, the endpoints are assumed to occur where the signal energy crosses this threshold. Corrections to these initial estimates are made by computing the zero-crossing rate in the vicinity of the endpoints and by comparing it with that of the silence. If detectable changes in zero-crossing rate occur outside the initial thresholds, the endpoints are re-designed to the points at which the changes take place.

In this model, Hamming window is used as it gives much greater attenuation outside the band pass than the comparable rectangular window. MATLAB 7.0 is used for the implementation.

2.4 Summary

The time-domain and frequency-domain approaches measure one or more acoustic features which reflect the production characteristics of the voiced sounds such as energy, periodicity and short-term correlation. Some parameters used are zero-crossing rate, autocorrelation coefficient, normalized LP error, normalized low-frequency energy, cepstral peak strength, harmonic measure from the instantaneous frequency amplitude spectrum. In rule-based approach, voiced/unvoiced/silence decisions are taken by setting thresholds on individual parameter values (chosen empirically). The main problem with these methods is in setting thresholds which are critical in determining the performance of voiced/unvoiced detection. Also, most of these measures of voicing are susceptible to noise, and the performance deteriorates with decreasing signal-to-noise ratio (SNR).

Other models such as neural network models, Gaussian mixture models (GMM) or hidden Markov models (HMM) are also used for combining evidence from multiple features. These methods do not depend critically on threshold setting, but require training data for different types of background noises. They assume different models of random process for speech and background noise, and estimate the parameters of the underlying distributions. Performance of these approaches depends on the choice of the probability distributions, and the ability to estimate the parameters of the noise distribution. Generally these methods do not make use of the knowledge of speech production mechanism in any significant way. Also, most of these methods do not evaluate separately the performance of detecting voiced and unvoiced regions of speech.

CHAPTER THREE

FEATURE EXTRACTION AND WINDOWING

3.1 Windowing

To perform acoustic analysis on speech signal, the speech signal may be long enough to make the process impossible. Also the properties of speech are time-varying and we cannot analyse the whole signal to extract its features. To solve these problems we should divide signal into small frames and prepare them for short term analysis.

When we prepare digital speech signals for acoustic analysis, we must first select a number of speech samples upon which we will carry out the analysis. As acoustic analysis attempt to extract the sine waves that add up to produce a complex waveform, it is necessary to analyse more than one sample. One sample does not change and so there is nothing to analyse. To select a series of speech samples for spectral analysis we need to "window" the original waveform. Hence, the speech signal must be multiplied by an appropriate window function.

When a signal is multiplied by a window function, the product is zero-valued outside the interval (a more general definition of window functions does not require them to be identically zero outside an interval): all that is left is the part where they overlap; the "view through the window". The window function, w(n), serves not only to select the correct segment of speech for processing, but also to weight the speech samples of s(n). The selected segment of speech is referred to as the *speech frame*. The shape of the window affects the frequency representation, S(k), by the frequency response of the window itself. The multiplication of a time-domain speech sequence s(n) with a time domain window w(n) is the same as the convolution of S(k) and W(k) in the frequency domain. So, the impact of a window shape can be analysed by examining its discrete Fourier transform (DFT).

Below is a vivid analysis of the most commonly and widely used window types used for acoustic analysis of speech signal.

3.1.1 The Rectangular Window

The rectangular window is sometimes known as a Dirichlet window. It is the simplest window, taking a chunk of the signal without any other modification at all, which leads to discontinuities at the endpoints (unless the signal happens to be an exact fit for the window length).

- ❖ A rectangular window simply passes all values between times t_1 and t_2 without modification (effectively multiplying each sample by 1) as depicted on Figure 3.1 and 3.2.
- ❖ A rectangular window is not normally used as it has a complex spectrum of its own which contaminates (distorts) the spectrum of speech.
- ❖ A rectangular window is given by:

w(n) = 1 for (m-1)N ≤ n ≤ mN
0 otherwise

Where, N is the length (width) of the window, in samples and n is an integer, with values 0 ≤ n ≤ N-1.

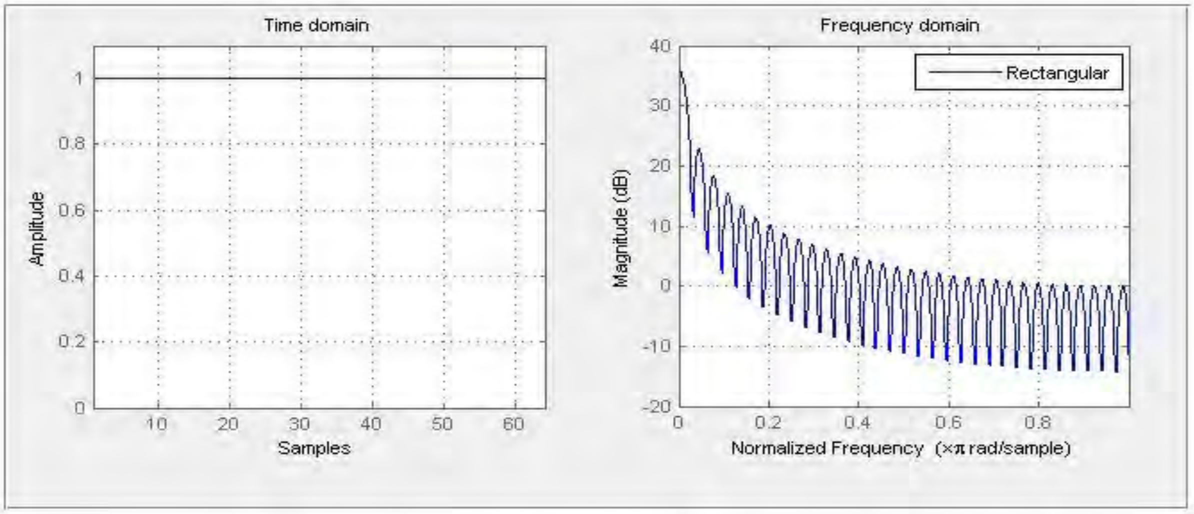


Figure 3.1: Window shapes of a rectangular window in time-domain and frequency domain

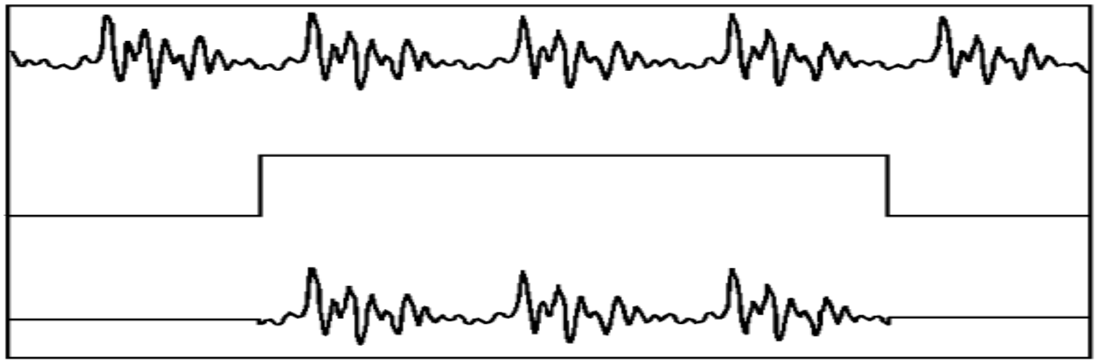


Figure 3.2: A Rectangular window multiplies the signal by "1" between two points and by "0" outside those points.

3.1.2 The Hamming, Hanning and Gaussian Window

The Hamming, Hanning and Gaussian are raised cosine functions with similar frequency characteristics, except the hamming window is raised at the edges as illustrated in Figure 3.3, 3.4, 3.5 and 3.6. Hamming, Hanning and Gaussian windows are members of a family of windows known as *raised cosine windows*.

- A single cycle of a cosine is inverted and shifted so that its values range between 0 and 1.

- This class of windows has no significant effect on the shape of the spectrum of the windowed speech and so these windows are often used during the frequency analysis of speech sounds.

The Hamming window is given by:

$w(n) = 0.54 - 0.46\cos\frac{2\pi n}{N}$, for $0 \le n \le N-1$

And the Hanning by:

$w(n) = 0.5 - 0.5\cos\frac{2\pi n}{N}$, for $0 \le n \le N-1$

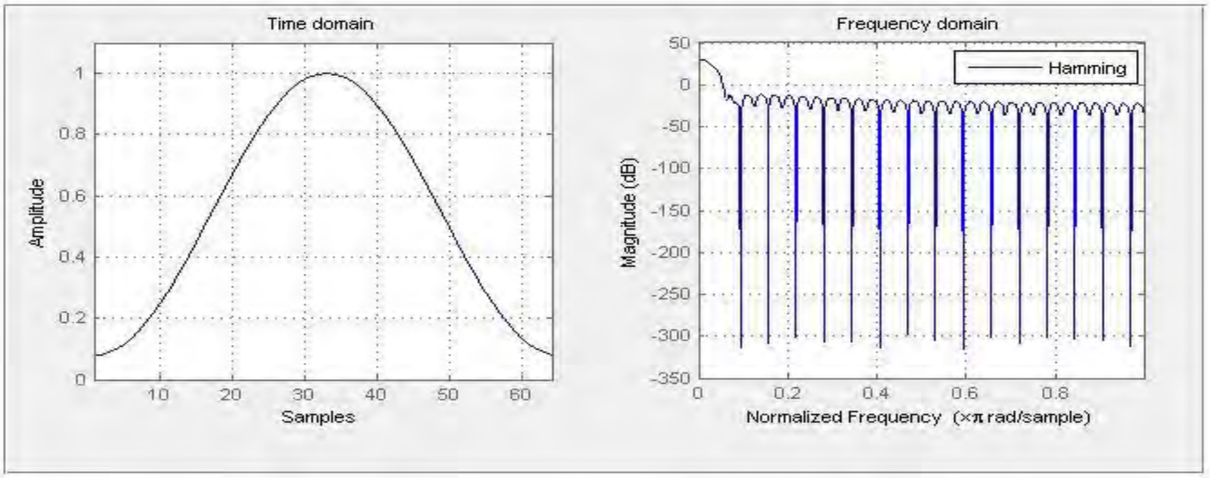


Figure 3.3: Window shapes of a hamming window in time-domain and frequency domain

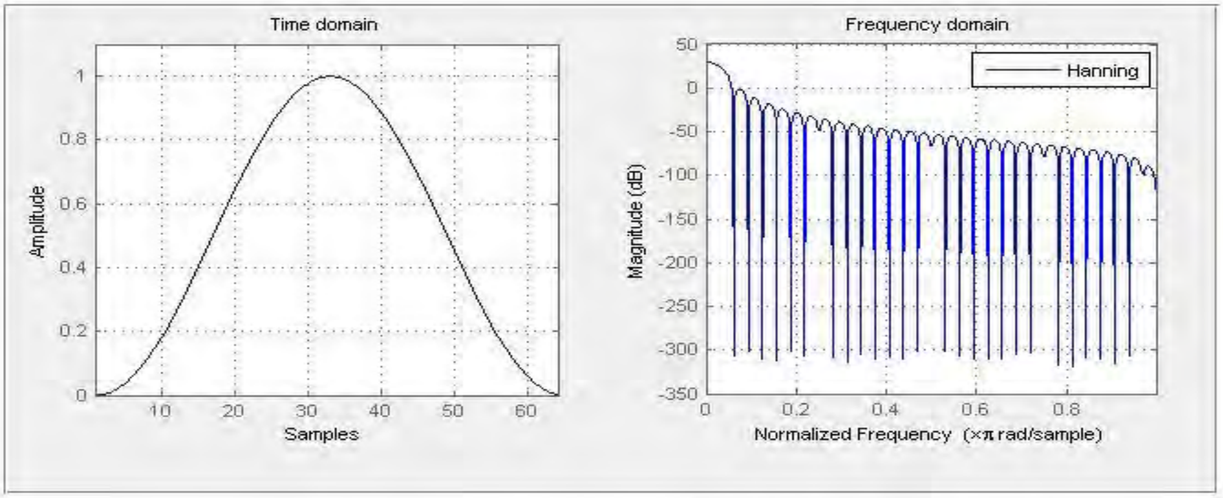


Figure 3.4: Window shapes of a Hanning window in time-domain and frequency domain

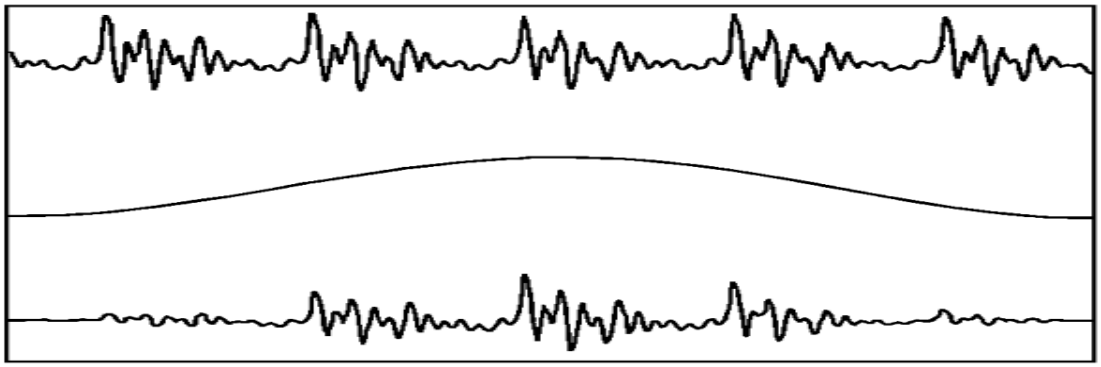


Figure 3.5: A Hanning window multiplies the sound by continuously changing values (based on a cosine) and by "0" outside this window

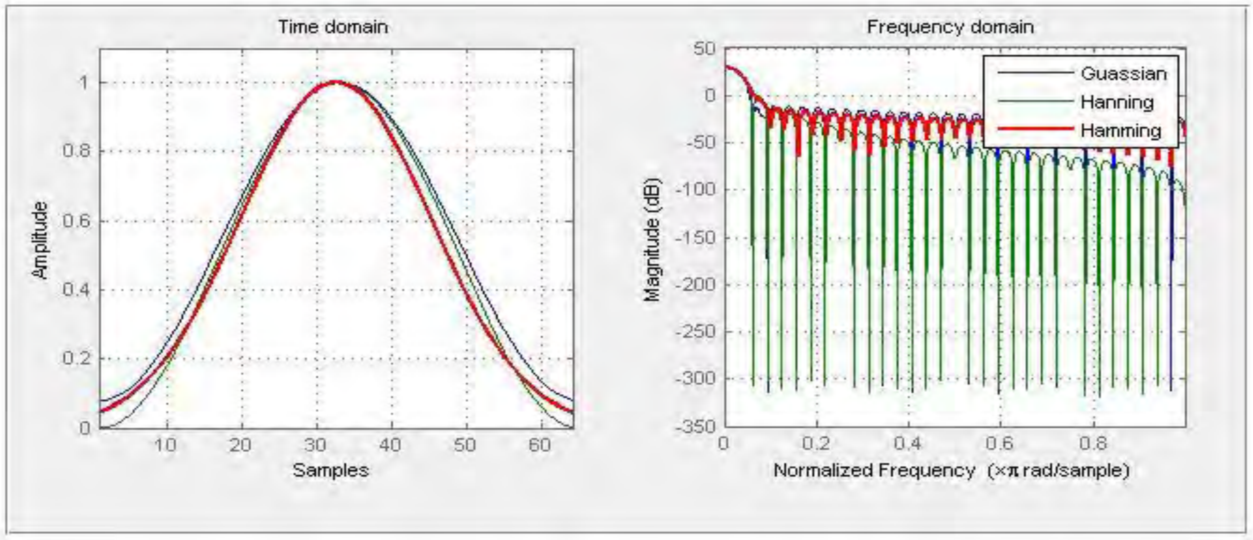


Figure 3.6: Comparison between Hanning, Hamming and Gaussian window

Spectral analysis involves a trade-off between resolving comparable strength signals with similar frequencies and resolving disparate strength signals with dissimilar frequencies. That trade-off occurs when the window function is chosen. One metric used to compare windows is the maximum scalloping loss of the window. The rectangular window is noticeably worse than the others in terms of this metric. Other metrics that can be seen are the width of the main lobe and the peak level of the side lobes, which respectively determine the ability to resolve comparable strength signals and disparate strength signals. The rectangular window (for instance) is the best choice for the former and the worst choice for the latter.

There are other windowing functions that can be used for different applications with different characteristics. Table 3.1 summarizes the most common windows and their features. This table can be used to choose the best windowing function for each application.

Table 3.1: Table of window types

Window	***Best for these Signal Types***	***Frequency Resolution***	***Spectral Leakage***	***Amplitude Accuracy***
Barlett	Random	Good	Fair	Fair
Blackman	Random or Mixed	Poor	Best	Good
Flat top	Sinusoids	Poor	Good	Best
Hanning	Random	Good	Good	Fair
Hamming	Random	Good	Fair	Fair
Kaiser-Bessel	Random	Fair	Good	Good
None (boxcar)	Transient and synchronous sampling	Best	Poor	Poor
Tukey	Random	Good	Poor	Poor
Welch	Random	Good	Good	Fair

3.1.3 Overlap Processing

One of the disadvantages of windowing functions is that the beginning and end of the signal is attenuated in the calculation of the spectrum. This means that more averages must be taken to get a good statistical representation of the spectrum, increasing the time to complete the measurement. Overlap processing is a feature that is available in most signal analysers that can recover the lost data and reduce the measurement time. This processing reduces the total measurement time by recovering a portion of each previous frame that otherwise is lost due to the effect of the windowing function. Overlap processing is particularly effective at reducing the measurement time for low frequency tests (generally under 50 Hz) for which the frame acquisition times are very long.

3.2 Feature Extraction

When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of *features* (also named *feature vector*). Feature is synonymous of input *variable* or *attribute*. Transforming the input data into the set of features is called *feature extraction*. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

Finding a good data representation is very domain specific and related to available measurements. For example, in medical diagnosis the features may be symptoms, that is, a set of variables categorizing the health status of a patient (e.g. fever, glucose level, etc.). In voicing detection the specific information (features) used in the feature vector can be obtained using two basic methods that extract useful information from the analysis of variations across the sequences of the speech segments used: *time-domain methods* and *frequency-domain methods*. These two methods of feature extraction in voicing detection are described in detail in Section 3.2.1 and 3.2.2 respectively.

3.2.1 Time Domain Methods

Time-domain methods use the most basic approach to the problem of voicing detection which is to look at the waveform that represents the change in air pressure over time, and attempt to detect the voiced/unvoiced/silence speech from that waveform.

There are important features such as short-time energy, average magnitude, short-time average zero crossing rate (ZCR) and auto correlation function, that can be extracted from a signal in the time domain. Time domain processing methods involve the waveform of the speech signal. A waveform is the representation of a speech signal as a function of time. All of the properties are obtained with a short-term approach and provide a rough but meaningful representation of the speech signals. Short-term analysis is an approach which takes into account segments short enough (e.g. 20-35 ms) to be considered as sustained sounds with stable properties. As pointed out in the work of [6], the reason behind using short-term approach is to detect rapid changes in the speech signal, and this is based on physiological measurements made using x-rays on a human vocal tract. The measurements have shown that during such a time humans cannot significantly change the shape of the vocal tract.

The first two properties are ***short-time energy*** and ***average magnitude***. They carry the same kind of information, but the second one is less sensitive to local fluctuations. They are especially important to detect silences or to distinguish between voiced and unvoiced segments in spoken data, but they can also be used to detect the transition from unvoiced to voiced speech and vice versa.

The short-time energy E_n of a signal x(n) with window size n can be extracted through the following equations [27]:

$$E_n = \sum_{m=-\infty}^{\infty} [\mathrm{x(m)w(n)}]^2 \qquad (1)$$

Where:- *w(n)* is a windowing function.

E_n is the nth frame short-time energy

One problem with the short time energy function is that it is very sensitive to large signal levels since the sample values are squared. Moreover, the lowest energy parts of the signal tend to be suppressed. For example the energy of the unvoiced phonemes at the end of the word “six” is so much lower than the other parts of the word that makes it difficult to distinguish them with respect to the silence. For this reason, E_n is often replaced with the short-term average magnitude M_n:

$$M_n = \sum_{m=-\infty}^{\infty} |\mathrm{x(m)w(n)}| \qquad (2)$$

The energy of voiced speech is much greater than the energy of unvoiced speech. The values of E[n] for unvoiced speech segment are significantly smaller than the voiced segments. From this we can deduce that if the calculated energy of the speech signal being examined is greater than some threshold value it can be said that it is voiced speech, otherwise it is unvoiced. Moreover the energy function can also be used to locate approximately the time at which voiced speech becomes unvoiced and vice versa and for a high quality speech the energy can also be used to distinguish speech from silence.

Another important aspect of a signal can be obtained through a simple time domain measure, called ***short time average zero-crossing rate*** (ZCR).The ZCR enables us to obtain a rough idea of the frequencies represented in the data. In the context of discrete-time signals, a zero crossing is said to occur if successive samples have different algebraic signs. The rate at which zero crossings occur is a simple measure of the frequency content of a signal. Zero-crossing rate is a measure of number of times in a given time interval/frame that the amplitude of the speech signal changes its sign (passes through a value of zero) as shown in Figure 3.7 [3].

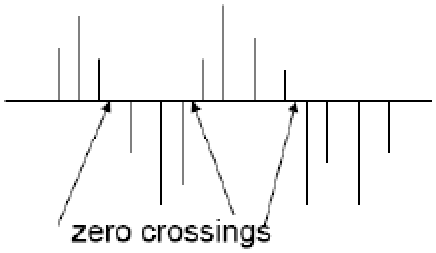


Figure 3.7: Sample Zero Crossing Rate

An appropriate definition is [27]:

$$Z_n = \sum_{m=-\infty}^{\infty} |\mathrm{sgn}[x(m)] - \mathrm{sgn}[x(m-1)]| w(n-m) \qquad (3)$$

Where:

$$\mathrm{sgn}[x(n)] = \begin{cases} 1 \text{ if } x(n) \geq 0 \\ -1 \text{ if } x(n) < 0 \end{cases}$$

The zero crossing count is an indicator of the frequency at which the energy is concentrated in the signal spectrum. The model for speech production suggests that the energy of voiced speech is concentrated below about 3 KHz because of the spectrum falloff introduced by the glottal wave, whereas for unvoiced speech, most of the energy is found at higher frequencies. Since high frequencies imply high zero-crossing rates, and low frequencies imply low zero-crossing rates, there is a strong correlation between zero-crossing rate and energy distribution with frequency.

Voiced speech is produced because of excitation of vocal tract by the periodic flow of air at the glottis and usually shows a low zero-crossing count, whereas the unvoiced speech is produced by the constriction of the vocal tract narrow enough to cause turbulent airflow which results in noise and shows high zero-crossing count. A reasonable generalization from this is that, if the zero-crossing rate is high, the speech signal is unvoiced, while if the zero-crossing rate is low, the speech signal is voiced.

The thought was that the ZCR should be directly related to the number of times the waveform repeated per unit time. It was soon made clear that there are problems with this measure. If the spectral power of the waveform is rich with harmonic spectra, then it will cross the zero line twice per cycle.

Even if this measurement has obvious limitations, it gives useful information that could be used in conjunction with other measurements.

Another useful short time measurement in the time domain is the ***autocorrelation function (ACF)***: The correlation between two waveforms is a measure of their similarity. The waveforms are compared at different time intervals, and their “sameness” is calculated at each interval. The result of a correlation is a measure of similarity as a function of time lag between the beginnings of the two waveforms. The autocorrelation function is the correlation of a waveform with itself.

The auto-correlation of a short-time signal x(n) is defined as:

$$\varphi_n(k) = \sum_{m=-\infty}^{\infty} \mathrm{x(m)x(m+k)\,w(n-m)} \qquad 0 \leq \mathrm{k} < T \qquad (4)$$

Where T is the number of autocorrelation points to be computed. The variable k is called lag, or delay.

Periodic waveforms exhibit an interesting autocorrelation characteristic: the autocorrelation function itself is periodic. As the time lag increases to half of the period of the waveform, the correlation decreases to a minimum. This is because the waveform is out of phase with its time-delayed copy. As the time lag increases again to the length of one period, the autocorrelation again increases back to a maximum, because the waveform and its time-delayed copy are in phase. The first peak in the autocorrelation indicates the period of the waveform.

The short-term properties considered so far (energy, average magnitude and ZCR) provide a single value for each analysis frame identified by a specific position of the window. This is not the case of the short-time autocorrelation function which provides, for each analysis frame, a function of the lag. Usually a constant threshold is adopted to which the correlation peak value is compared for voiced/unvoiced/silence decision. Problems with this method arise when the autocorrelation of a harmonically complex, pseudo-periodic waveform is taken.

In general, time domain representation methods are attractive because the required digital processing is very simple to implement, and in spite of this simplicity, the resulting representation provides a useful basis for estimating important features of the speech signal. Time domain methods are intrinsically quite weak in the case of inharmonic signals or in signals with most of the power in high frequencies. In addition, such a method can only achieve limited accuracy because the value of any single parameter usually overlaps between categories, particularly when the speech is not recorded in a high fidelity environment.

3.2.2 Frequency Domain Methods

There is much information in the frequency domain that can be related to the voiced/unvoiced/silence segment of a speech signal. This class of methods involve (either explicitly or implicitly) some form of spectrum representation. The most well-know frequency domain methods for voiced/unvoiced/silence speech segment detection include cepstrum analysis and linear predictive coding methods.

Cepstrum analysis is a form of spectral analysis where the output is the log of the Fourier transform of the magnitude spectrum of the input waveform [8]. In this case, we pick the first peak in the signal synthesized from the log-magnitude of the Fourier transform. This algorithm tends to perform quite well in noisy conditions. However, it handles uneasily inharmonic sounds since it"s based on the assumption of evenly spaced partials.

The name cepstrum comes from reversing the first four letters in the word “spectrum”, indicating a modified spectrum. The independent variable related to the cepstrum transform has been called “quefrency”, and since this variable is very closely related to time it is acceptable to refer to this variable as time.

The short-time cepstrum can be applied to detect local periodicity (voiced speech) or the lack thereof (unvoiced speech). Presence of a strong peak implies voiced speech, and the quefrency location of the peak gives the estimate of the pitch period.

The other frequency domain method towards voiced/unvoiced/silence speech segment detection is the ***Linear Predictive Coding (LPC)*** analysis. It is a predominant technique for estimating basic speech parameters, e.g., pitch, formants, spectra, vocal tract area function, and for representing speech for low bit-rate transmission or storage. The importance of this method lies in its ability to provide extremely accurate estimates of the speech parameters, and in its relative speed of computation [27].

The basic idea behind linear predictive analysis is that a speech sample can be approximated as a linear combination of past speech samples. By minimizing the sum of the squared differences (over a finite interval) between the actual speech samples and the linearly predicted ones, a unique set of predictor coefficients can be determined. (The predictor coefficients are the weighting coefficients used in the linear combination).

For voiced/unvoiced/silence classification, a spectral characterization of each of the three classes of signal is obtained during a training session, and an LP Coefficient distance measure can be computed to make the final discrimination as voiced/unvoiced/silence based on some threshold value and some rules defined.

The basic problem of linear predictive analysis is to determine a set of predictor coefficients directly from the speech signal in such a manner as to obtain a good estimate of the spectral properties of the speech signal.

Frequency-domain approaches have some downsides, including the need to enhance the resolution with zero padding. The algorithm accuracy depends also on the harmonicity of the signal spectrum.

CHAPTER FOUR

DESIGN AND IMPLEMENTATION OF A VOICING DETECTION

4.1 Design of the Voicing Detection

Voiced/unvoiced/silence speech segment detection is a method of assigning and labelling a specific speech category (voiced/unvoiced/silence) to a speech segment to discriminate a speech sound from a background noise based on acoustic characteristics of the speech segment.

In this chapter, a detail description of design issues and techniques of voicing detection are dealt.

4.1.1 Approaches and Techniques

So far many voicing detection researches have been done and different approaches have been used for voiced/unvoiced/silence classification, where the well-known ones are rule-based approaches, statistical approaches and neural network approach. All of the proposed methods have their merits, and preference for one over another.

In this thesis work, three different models were examined namely the rule-based, statistical and neural network. The rule based approach as its name indicates relies on rules which are either handcrafted or machine learned rules. The rules are the important elements for annotating speech segments in the rule-based approach. The statistical based detector relies on the statistical property of speech signals. Such statistical property can be distributional probability of speech signals with tags which can be obtained during the training phase of the system. Finally, the neural network (NN) detectors use acoustic features of speech signals that can be extracted from the speech waveform or spectrogram to classify speech categories.

4.1.2 Design Goals

The general goal of this research work is to explore the possibility for systematic pattern extraction and detection of voiced/unvoiced/silence region of speech signal thereby attaining better accuracy by using neural network approach.

4.1.3 Architecture of the Voiced/Unvoiced/Silence Classifier Model

The overall architecture of the voicing detection process for this research work is described in Figure 4.1 and 4.2.

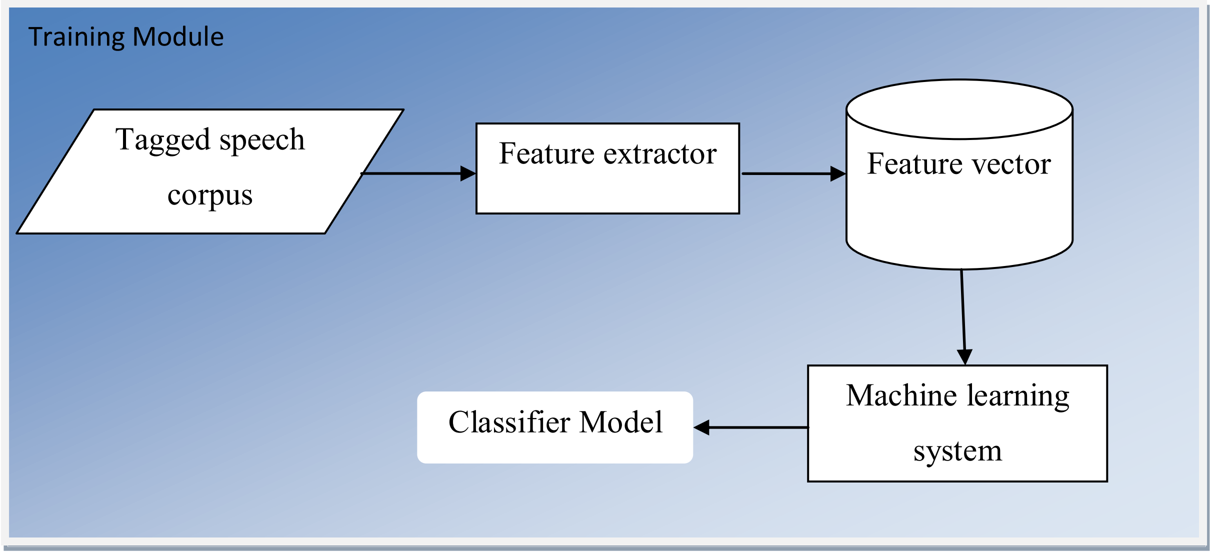


Figure 4.1: The general architecture training process

Figure 4.1 shows the voicing detection classifier training process. A supervised learning method is used for training the classifier model i.e. the training speech corpus contains tagged (labelled) amharic speech sentences collected from different sources. The tagged speech corpus is given to the feature extractor module. The result of the feature extraction process is the production of a feature vector, which is used as an input to the machine learning system. The feature vector is given to the machine learning system, for learning a pattern and finding an optimal solution for each speech category (voiced, unvoiced and silence). After the machine learning system is trained with the training set a classifier model is created. The result of the training process is the production of a classifier model which is used for annotating untagged speech sentences which in turn be evaluated against manually tagged data (true class) of the input speech sentences. The testing and evaluation process is shown in Figure 4.2. The detailed implementation issues of the speech corpus, feature extractor and machine learning system is decribed in Section 4.2.1, 4.2.2 and 4.2.3, respectively.

As shown in Figure 4.2, the untagged speech corpus is given to the feature extractor so as to prepare a feature vector and make ready for classsification by the classifier model developed during the training process. Afterwards, the classifier model selects an optimal speech category for a given speech segment and gives the classified speech category along with the corresponding feature vector as an output. The output of the classifier (classified speech) is compared against a manually tagged (so called true class) speech courpus and accuracy of the classifier is calculated by counting the number of correctly classified speech segments.

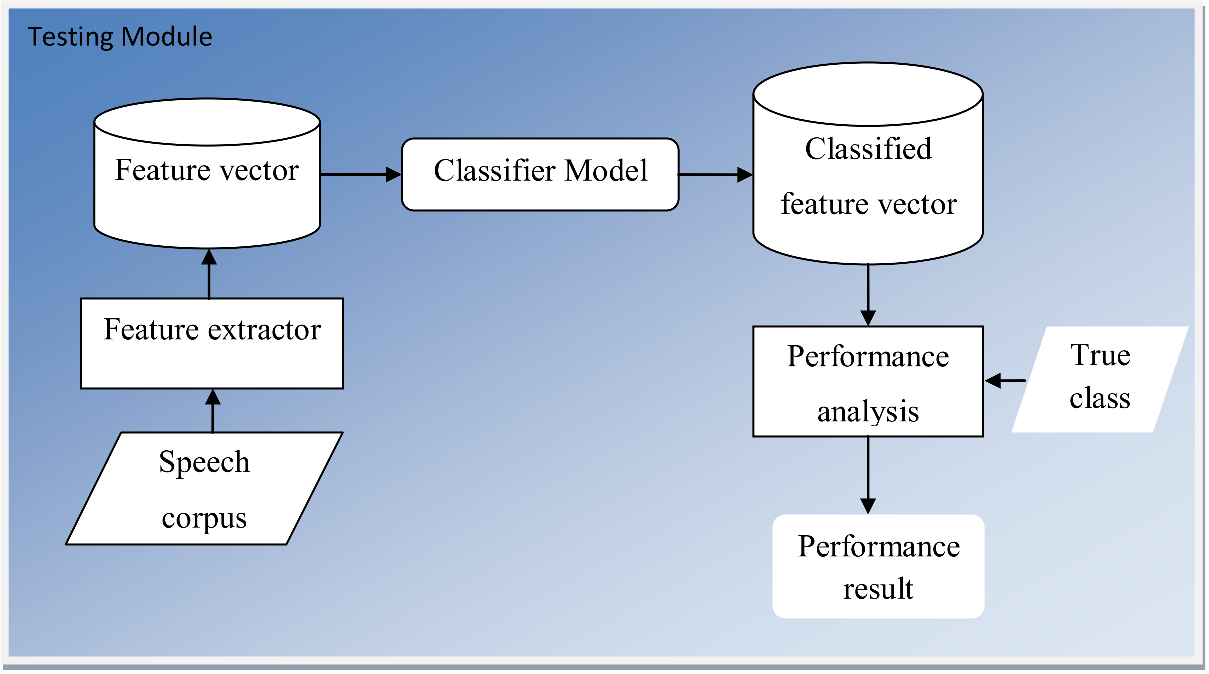


Figure 4.2: Classifier testing and evaluation process

The over all flow chart of the training and classification processes is shown in Figure 4.3 and vivdly described below.

As shown in Figure 4.3, at the first stage, the speech signal is segmented into intervals in frame by frame without overlapping. In the frame-by-frame processing stage, the speech signal is segmented into a non-overlapping frame of samples. It is processed into frame by frame until the entire speech signal is covered. And then, a feature vector was obtained for each 20, 25, 30 and 35 ms segment of speech. The feature vector was a combination of 13 MFCC, 13 LPC and two waveform parameters; the zero-crossing rate and speech energy.

The LPC were derived from 12 pole linear predictive analysis of the speech signal. The autocorrelation methods of autoregressive (AR) modelling were used in calculating the LP coefficients.

The 13 Mel-frequency cepstral coefficients (MFCC) are calculated by taking the absolute value of the short time fourier transform (STFT), warpping to a Mel frequency scale, taking the discrete cosine transform (DCT) of the log-Mel-spectrum and return the first n cepstral components (n=13 in this case).

The rectangular window is used in calculating energy (En), zero-crossing rate (ZCR), 13 LPC and 13 MFCC.

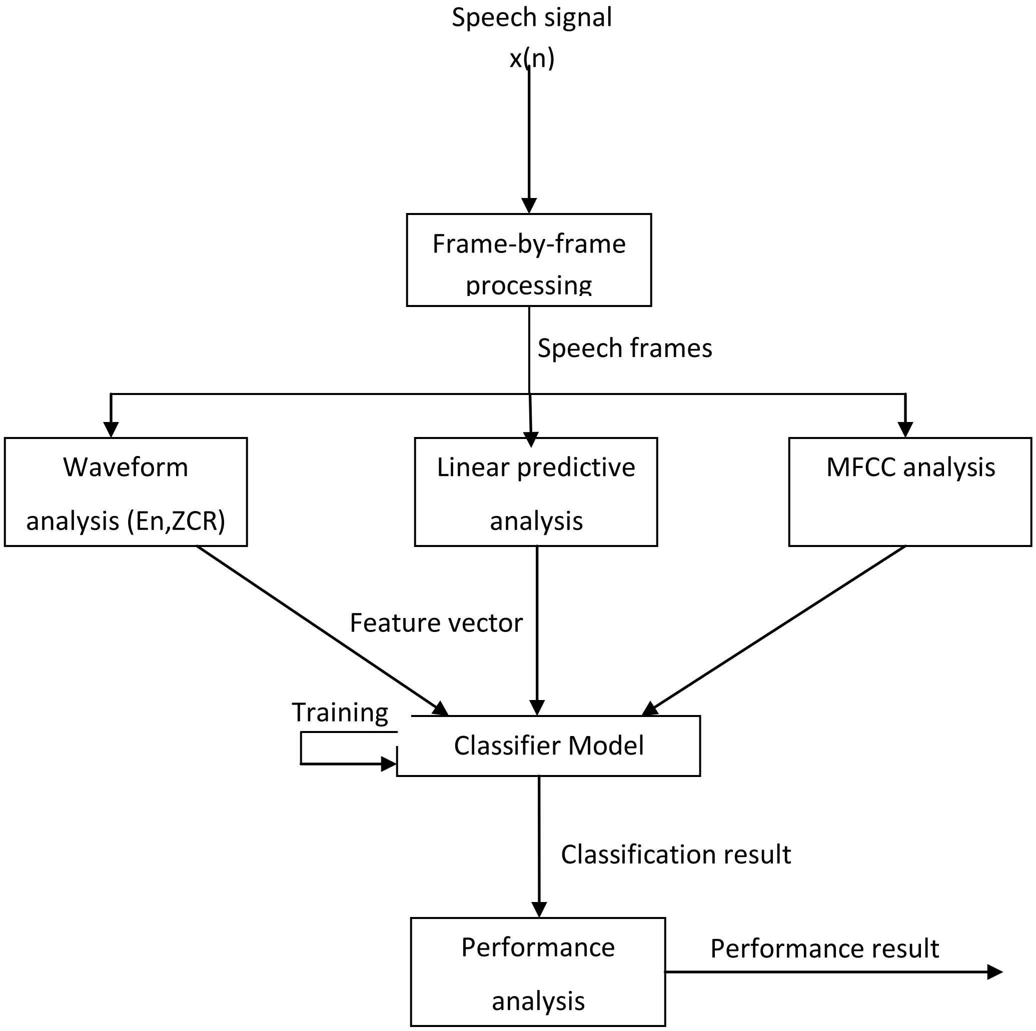


Figure 4.3: Flow chart of classifier training and classification processes

Both the training and testing feature vectors are fed to the classifier model for training and testing (evaluating) the performance of the model respectively. The voiced/unvoiced/silence classification was made for each input feature vector after training was completed using the test set. The classification output is compared with the same data to the test set which is mannualy classfied during the performance analysis. Five different types of classifiers and six combinations of features are tested in this process for 20, 25, 30 and 35 milliseconds of speech segment. The classification performance of each classifier as a function of the frame length and feature vector content was also evaluated and compared with each other. A conclusion about which classifier and feature performs well, is derived based on the result of the comparison of each classifier.

The objective here is to empirically select a classifier that has a simple architecture and reasonably high classification performance and a feature that best represents each speech category. For this work, an extensive search for an optimal classifier and feature selection is undertaken.

For testing and analysing the classifiers and the features weka 3.6.0 is used. The classifiers used in this thesis work include two rule-based (Decision Tables and JRip), two decision tree (J48 and simple CART) and a neural network (with single and double hidden layers having different numbers of neurons on the hidden layers). The six feature combinations used in this process are illustrated in Table 4.1.

Table 4.1: Feature combinations

SNo	*Code*	*Features Used*	*Columns*	*Remark*
1	Feature Vector 1	Energy, Zero-crossing rate	2	
2	Feature Vector 2	LPC	13	
3	Feature Vector 3	Energy, Zero-crossing rate, LPC	15	
4	Feature Vector 4	MFCC	13	
5	Feature Vector 5	Energy, Zero-crossing rate, MFCC	15	
6	Feature Vector 6	Energy, Zero-crossing rate, MFCC	15	Principal component analysis (PCA) is applied[2]

Classification rates as a function of the type of classifier, feature combination and frame size used were obtained for all the classifiers. Each classifier is trained by a set of 600 (out of 900) randomly selected training speech sentences and tested using 300 (out of 900) randomly selected testing speech sentences. The performance of each classifier is described in Chapter 5 of this work.

4.1.4 Summary

Voiced/unvoiced/silence detection and discrimination is the process of assigning speech categories to a sequence of speech segments in a sentence. This problem can possibly be tackled using different approaches. Such approaches are rule based, statistical and neural network which have their own pros and cons. As far as a voicing detection with possibly a higher performance of classification is desired, there is a need to make an empirical selection of one or more than one approach and take an advantage from an approach thereby remedying the shortcomings of the approach.

4.2 Implementation of the Voicing Detection

Here the details regarding the implementation of the classifier architecture and corpus preparation are explained.

To begin with, MATLAB 7.9, Audicity 1.3.13, wavesurfer 1.8.5 and weka 3.6 machine learning tool are used in the entire implementation of the voicing detector. The rationale behind the choice of these tools is, they are suitable for different speech processing tasks and machine

[2] PCA-is a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called *principal components*.

learning process. MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numerical computation. It can be used in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modelling and analysis, and computational biology. Moreover, it contains different add-on toolboxes (collections of special-purpose MATLAB functions) that extend the MATLAB environment to solve particular classes of problems in these application areas. The second tool used is Audicity which is free, open source software for recording and editing sounds. It is available for Mac OS X, Microsoft Windows, GNU/Linux, and other operating systems [32]. The other tool we used is WaveSurfer which is also an open source tool for sound visualization and manipulation. Typical applications of wavesurfer include speech/sound analysis and sound annotation/transcription [30]. Moreover, wavesurfer can be extended by plug-ins as well as embedded in other applications. Finally, we used Weka which is a collection of state-of-the-art machine learning algorithms and data pre-processing tools [33]. The algorithms can either be applied directly to a dataset or called from a Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

In the following sections, the details starting from the corpus preparation to the implementation of the classifiers is discussed.

4.2.1 Corpus Preparation

Corpus, plural corpora is a large collection of linguistic data, either written texts or a transcription of recorded speech, which can be used as a starting point of linguistic description or as a means of verifying hypotheses about a language (corpus linguistics) [7]. A corpus can be a flat text i.e. a text with no additional linguistic information or a text whereby each word in the text is attached with linguistic information [12]. The corpus with additional linguistic information can be called as *annotated/tagged corpus*. Such linguistic information in the annotated corpus can be part of speech information, sound category, sentiment information that specifies the word‟s word class category and sentiment category respectively. The annotated corpus can be used in many NLP applications like part of speech tagger training and testing, parsing, sentiment analysis etc. In this thesis work, the annotated corpus used is considered to be a speech tagged with the corresponding speech sound category.

In fact, the tagged speech i.e. the annotated corpus is thought to represent all the domains of the language. The domains can be text of news category, fiction category, editorial category, scientific category etc. A corpus with all possible categories is called a *balanced corpus* [12]. Though it is difficult to prepare a balanced corpus, it is an important element in most natural language processing applications in general and voicing detection in particular.

Development of balanced corpus takes time, effort/skills of language experts and money as it needs data to be collected from different domains. The essence of developing a balanced corpus is, in fact, to increase the performance of the classifier when it classifies any speech segment taken from any category which implies directly that balanced corpus contains as many speech categories as possible from different categories. However, a category specific corpus contains words that are mostly used in that category and if a text from other category to be classified is given to the classifier trained on this corpus, the performance of the classifier may be degraded. But if the text taken is from that category, the performance is assumed to perform as expected.

The overall corpus preparation process of the voicing detection is shown in Figure 4.4.

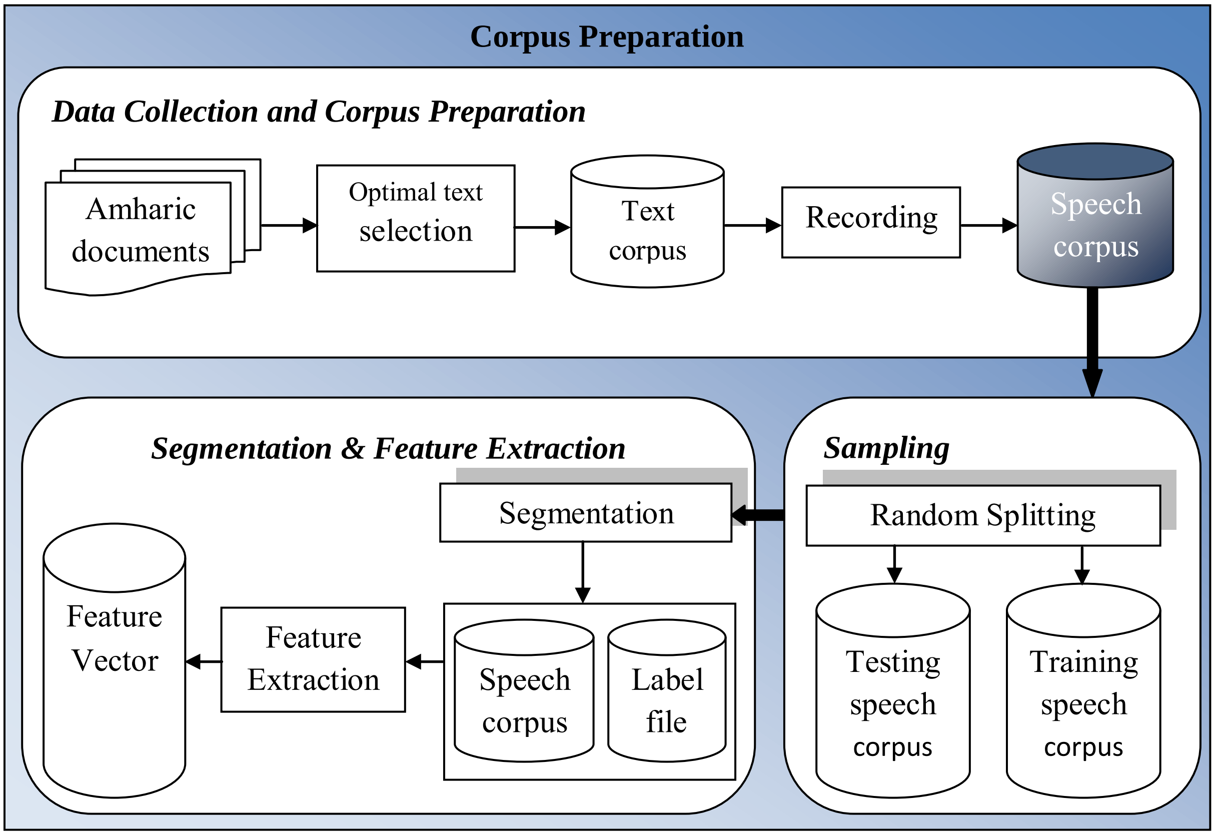


Figure 4.4: Data preparation process of the voicing detection system

The speech corpus prepared for training and testing contains a total of 900 sentences. In order to build phonetically balanced speech corpus, text data is collected from various sources of Amharic documents as indicated in Section 4.2.1.1. Sample text sentence collected for this research work is attached in Appendix A. Once the texts are collected, they are automatically transcribed to their corresponding Latin representation as per ASCII transliteration table attached in Appendix D.

The transcribed text is randomly split in to two. These are training and testing set with 600 sentences and 300 sentences, respectively. These two sets of texts are known as prompt files and are considered as text corpuses which need to be recorded. Other main parts of corpus preparation and description, namely, data recording, creating label files and coding the acoustic data are discussed vividly in the following subsections.

4.2.1.1 Data Collection

The first stage of the voiced/unvoiced/silence speech segment detection research is data (speech corpus) preparation. Speech data is needed both for training and testing. In this research, optimal text selection technique is used to prepare text corpus from various sources. These documents are used as data sources to get phonetically rich and balanced collections of sentences. Amharic Bibles, health news, political news, sport news, economy news, penal code, federal Negarit Gazeta and Amharic fiction named "Fiker Eskemekaber" are data sources used for text corpus preparation. This text corpus contains 900 Amharic sentences.

Afterwards, all of the sentences are recorded by one person (male) in the age range of 29. Then the speech data is split into two sets: training (600 Sentences) and test set (300 sentences) using systematic random sampling. The test data provides the feature vectors against which the classifiers performance can be measured. The training data is used in conjunction with the class (category) labels needed to start the training process. Here for convenience, the sentences needed for training and testing are taken from eight different Amharic sources. A sample text corpus is shown in Appendix A.

4.2.1.2 Recording the Data

Before we begin, we make sure that the resident we were recording in was as quiet as possible. In addition, we turned off speakers while recording to avoid acoustic feedback in audio files. Both the training and test data are recorded with one male speaker in the age range of 29. The profile of the speaker is indicated in Table 4.2. It is important to note that no analysis is made in determining the age range. Only availability of speaker was considered. The training and test sentences are recorded by a speaker whose first language is Amharic. The data is recorded using Audacity, which is free, open source software for recording and editing sounds, at a sampling rate of 48 KHz, with 16 bits/sample and mono channel. The microphone used was a headset with close-speaking, noise cancelling and mono-phone features.

Nine hundred sentences listed in the prompt file (transcribed text corpuses) are recorded for both training and testing data by the same speaker.

Table 4.2: Speaker profile for speech recording

Speaker code	*Sex*	*Age*	*No of sentences recorded for training*	*No of sentences recorded for testing*
Speaker 1	M	29	600	300

After selecting the above speaker and the environment for the recording, the following audacity preferences are also set:

- Set the microphone volume to 1.0.
- Set the default Sampling Rate to 48 KHz.
- Set the default Sample Format to 16-bit.
- Set the Channels to 1 (Mono).
- Set the Uncompressed Export Format to WAV (Microsoft 16 bit PCM) or export the audio using FLAC format.

After getting the proper setting of audacity, the texts are recorded by the speaker properly within a quite environment. The wave files of 600 sentences are used for training and the rest 300 sentence wave files are used for testing purpose.

4.2.1.3 Sampling

The research used a total of 900 sentences out of which 600 are chosen for training and 300 sentences are chosen as evaluation test sentences. Systematic random sampling technique is used for splitting the speech corpus into training and test sets. The training set consists of 600 speech sentences of the total speech corpus with their corresponding label files. The rest 300 speech sentences with their corresponding label file are test data set of the classifier. The end results of systematic random sampling technique are two speech corpuses: the training set and test set. The training data set is used for building a classifier model where as the test data set is used to evaluate the performance and accuracy of the classifier trained using the training set.

4.2.1.4 Segmentation and Creating Label files

To create a feature vector and assign an appropriate speech class (voiced, unvoiced or silence) to each speech frame in the speech signal every speech sentence should have an associated phone level transcription (label file). To make this task easier, creating a word level transcription before creating the phone level transcription is required. Using the created word level transcription as a base, phone level transcription file is created using manual labelling. The final output of this process is the creation of a label file for each speech sentence. The format of a label file along with a sample data is illustrated in Table 4.3.

Speech segmentation is the process of identifying the boundaries between words, syllables, or phonemes in spoken natural languages. The lowest level of speech segmentation is the breakup and classification of the sound signal into a string of phones. The difficulty of this problem is compounded by the phenomenon of co-articulation of speech sounds, where one may be modified in various ways by the adjacent sounds: it may blend smoothly with them, fuse with them, split, or even disappear. This makes speech segmentation particularly phoneme level segmentation a very difficult task. Speech segmentation can be accomplished using two methods: manual or automatic. Manual segmentation refers to the process whereby an expert transcriber segments and labels a speech file by hand, referring only to the spectrogram and/or waveform. The manual method is believed to be more accurate over automatic segmentation. Automatic segmentation refers to the task of detecting the boundaries between words, syllables, or phonemes in a speech signal using carefully chosen procedures. The process may use speech signal information extracted from different properties of speech signal like formant trends, pitch, stress, vowel duration, power spectral density, zero crossing rate, and rhythm or intonation patterns.

In this research work, manual segmentation of speech is used to segment and label the speech file. For this operation wavesurfer 1.8.5 is used as a major tool. The segmentation process is done for all the 900 sentences. Every speech sentence is carefully examined and labelled manually using the selected tool. After the labelling is completed a label file is created for each sentence separately, which in turn is used for feature extraction. Figure 4.5 shows waveform of a sample Amharic speech sentence ("ብዙነ ሽ በ መምባ ይ ማራቶን ለ ድል ትጠበ ቃለ ች") along with the segment labels generated using wavesurfer. In addition, the corresponding label file generated for the above sentence is illustrated in Table 4.3. The label file has three columns which represent the initial and end time of a segment and the segment label (phoneme label).

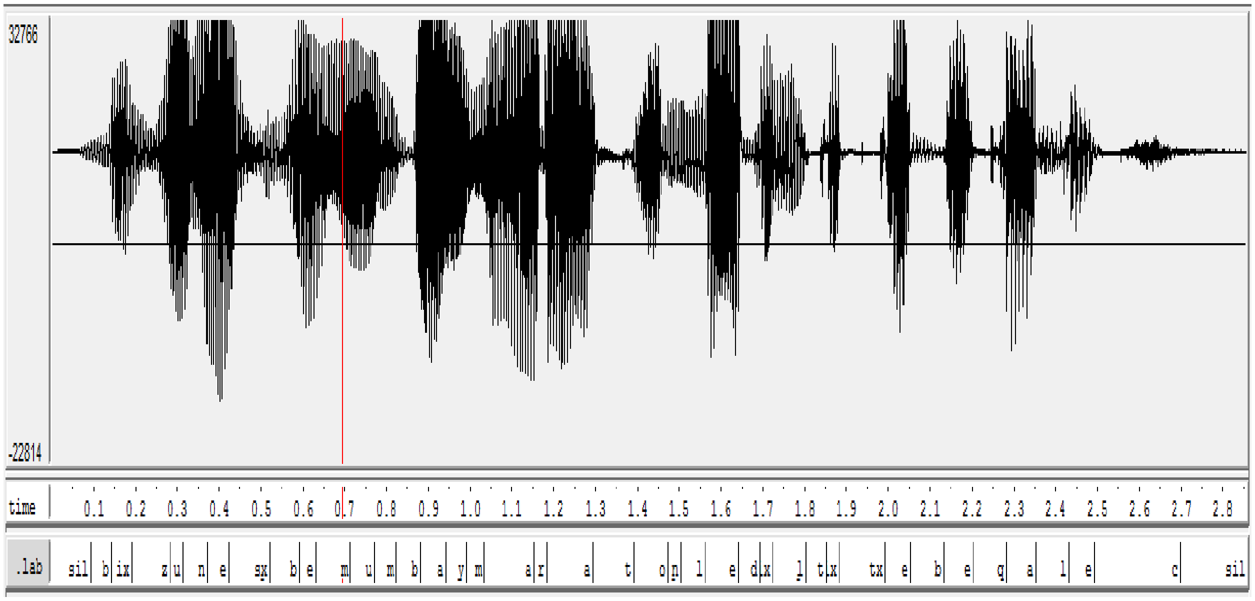


Figure 4.5: Waveform of a sample Amharic speech sentence

Table 4.3: Sample label file ("ብዙነሽ በሙምባይ ማራቶን ለድል ትጠበቃለች")

Start Time	*End Time*	*Label(Phoneme)*
0	0.0925	sil
0.0925	0.1425	b
0.1425	0.1925	ix
0.1925	0.2825	z
0.2825	0.3125	u
0.3125	0.3725	n
0.3725	0.4225	e
0.4225	0.5225	sx
0.5225	0.5925	b
0.5925	0.6325	e
0.6325	0.7125	m
0.7125	0.7725	u
0.7725	0.8225	m
0.8225	0.8825	b
0.8825	0.9425	a
0.9425	0.9925	y
0.9925	1.0325	m
1.0325	1.1525	a
1.1525	1.1825	r
1.1825	1.2925	a
1.2925	1.3925	t
1.3925	1.4725	o
1.4725	1.5025	n
1.5025	1.5625	l
1.5625	1.6425	e
1.6425	1.6925	d
1.6925	1.7225	ix
1.7225	1.8025	l
1.8025	1.8525	t
1.8525	1.8825	ix
1.8825	1.9925	tx
1.9925	2.0525	e
2.0525	2.1325	b
2.1325	2.2025	e
2.2025	2.2825	q
2.2825	2.3525	a
2.3525	2.4325	l
2.4325	2.4925	e
2.4925	2.6975	c
2.6975	2.86	sil

4.2.1.5 Frame by Frame Processing

In the frame-by-frame processing stage, the speech signal is segmented into a non-overlapping frame of samples. It is processed into frame by frame until the entire speech signal is covered. At the beginning, we set the frame size as 960 samples (20 ms) with 48 KHz sampling rate. After the frame by frame processing is completed a feature extraction process is performed on each frame. The result of the feature extraction is the production of a feature vector containing 13 MFCC, 13 LPC coefficients, short-time energy and short-time zero crossing rate for each frame of the speech signal. Afterwards the same process is repeated with the frame size as 1200 samples (25 ms), 1440 samples (30 ms) and 1680 samples (35 ms) with 48 KHz sampling rate. Table 4.4 shows the contents of the datasets for the 20, 25, 30 and 35 millisecond frame size. It indicates the total number of frames, voiced frames, unvoiced frames and silence frames both for the training and test set.

Table 4.4: Dataset contents

Frame Size	*Contents*	*Training Set*	*Test Set*	*Total*
20 millisecond	No. of Sentences	600	300	900
	No. of Frames	143,884	71,986	215,870
	Voiced Frames	106,336	53,457	159,793
	Unvoiced Frames	28,107	13,889	41,996
	Silence Frame	9,341	4,640	13,981
25 millisecond	No. of Sentences	600	300	900
	No. of Frames	115,049	57,566	172,615
	Voiced Frames	84,924	42,764	127,688
	Unvoiced Frames	22,700	11,110	33,810
	Silence Frame	7,425	3,692	11,117
30 millisecond	No. of Sentences	600	300	900
	No. of Frames	95,875	47,965	143,840
	Voiced Frames	70,776	35,634	106,410
	Unvoiced Frames	18,917	9,262	28,179
	Silence Frame	6,182	3,069	9,251
35 millisecond	No. of Sentences	600	300	900
	No. of Frames	82,173	41,118	123,291
	Voiced Frames	60,658	30,509	91,167
	Unvoiced Frames	16,206	7,969	24,175
	Silence Frame	5,309	2,640	7,949

4.2.1.6 Coding the acoustic data and Feature Extraction

Digital signal processing techniques are applied to convert analogue signal into their digital signal representations. Since the speeches we have recorded are continuous or analogue; they have to be converted into discrete or digital representation via sampling and quantization. To convert the analogue speech signal into its corresponding digital representation and extract features, we used matlab 7.9. After the analogue signal is digitized feature extraction is conducted using the digitized speech corpus and the corresponding label file created during segmentation using the feature extraction module.

Recorded speech signals are parameterized into sequence of feature vectors and used in the voiced/unvoiced/silence classification process. The feature extraction process is the process of spectral parameter extraction (parameterization) which involves conversion of speech samples into feature vectors to provide spectral patterns of the speech. The feature extraction is performed using matlab signal processing functions and user defined function written using matlab by taking the label file and speech (wave file) as an input to the process. During feature extraction process we usually assume that the characteristics of the speech signal are stationary over a short time period, typically of the order of 20-35 milliseconds. The result of the feature extraction process is the production of a feature vector which will be used as the main input to the classifier.

For this research work, four main speech signal features are extracted during the feature extraction process namely, short-time energy, short-time zero-crossing rate, 13 MFCC coefficients and a 12-pole LPC coefficients for each 20, 25, 30 and 35 ms of the speech segment. The feature vector used is designed to be of different length depending on the type and combination of features used as indicated in Table 4.1. The structure of the feature vector along with a sample data is shown in Appendix B.

4.2.2 Implementation of Feature Extractor

The feature extraction component of a voiced/unvoiced/silence speech segment classification system maps the speech waveform into a sequence of feature vectors. This sequence of feature vectors is subsequently used to train acoustic model and decode input speech waveform.

To apply digital signal processing techniques to the speech waveform, the analogue signal is first converted into a digital signal. This is done via sampling and quantization of the waveform. Once the digital signal has been obtained, a variety of techniques are used to extract features which are useful for the speech segment classification task. These speech analysis techniques usually assume that the characteristics of the speech signal are stationary over a short time period, typically of the order of 20-35 milliseconds. The resulting features are a representation of the speech signal over this short time period. Parameterization is performed not only for size reduction of the original speech signal data but also for pre-processing of the signal that fits into the classification stage. An important property of feature extraction is the suppression of information that is irrelevant for a correct classification. The most popular features that are used for voiced/unvoiced/silence speech segment classification include, Mel Frequency Cepstral Coefficients (MFCC), Linear Predictive (LP) Coefficients, short-time energy, short-time zero-crossing rate, average magnitude, and auto correlation function.

We have used a combination of four features for our experiment in developing our research containing, short-time energy, short-time zero-crossing rate, 13 MFCC and a 12-pole LPC. The feature extraction module is implemented using matlab. The overall feature extraction and feature vector preparation process is depicted in Figure 4.6.

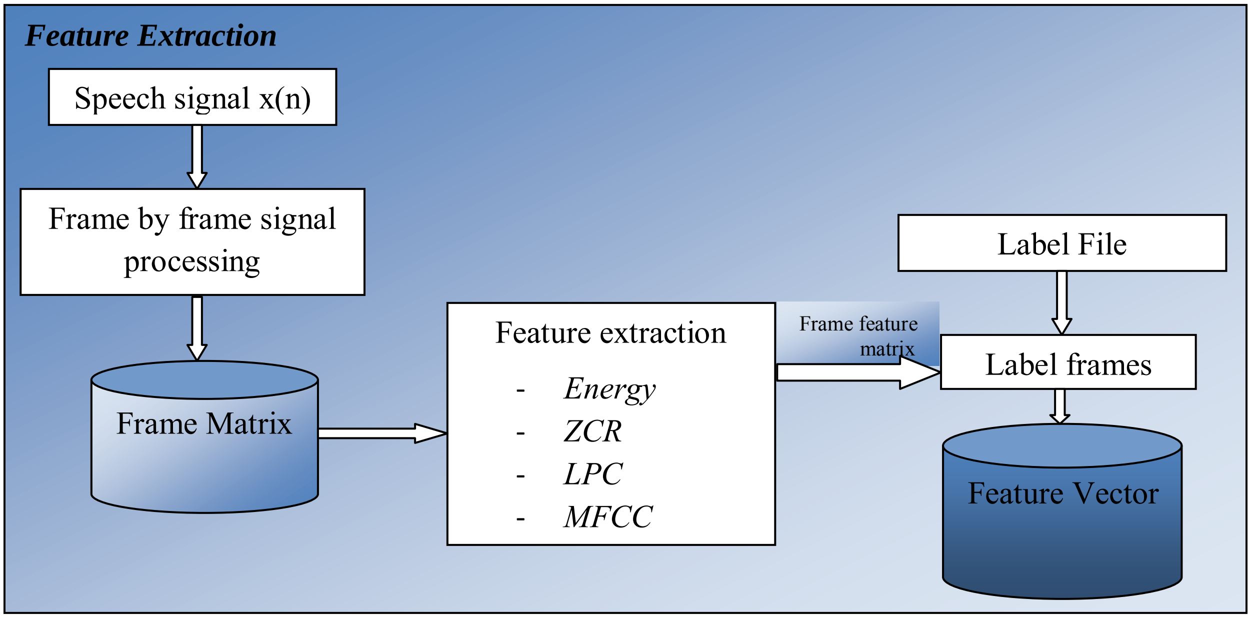


Figure 4.6: Feature extraction process

The input to the feature extraction module is the speech signal prepared during data preparation process. The speech signal is processed frame by frame (divide the signal into frames of 20, 25,30 and 35 ms segment), resulting a frame matrix containing frame by frame representation of the speech signal. The frame matrix in turn is used as input along with the corresponding label file for feature extraction of the speech signal. At this stage a feature vector (energy, ZCR, MFCC and LPC Coefficient) is extracted for every 20, 25, 30 and 35 ms of speech segment. Finally, the output of the feature extraction (frame feature matrix) is labled with an appropriate class label (Voiced,Unvoiced or Silence) to produce the final output: the feature vector. The final output of this process is the feature vector containing 12 LPC, an LPC gain, 13 MFCC, a short-time energy, and a short-time zero-crossing rate which is used as an input to the classifier. Sample matlab codes used for frame-by-frame processing, energy calculation, ZCR calculation, MFCC computation and LPC calculation are included in Appendix C.

4.2.3 Implementation of Voiced/Unvoiced/Silence Detector

An extensive search for an optimal classifier is undertaken to empirically select a classifier that has a simple architecture and reasonably high classification performance. Five different types of classifiers are tested in this research work namely; two rule-based (Decision Tables and JRip), two decision trees (J48 and simple CART) and a neural network (single and double hidden layer with different number of neurons on the hidden layer). The classification performance of each of these models is described in Chapter 5 of this research work. From all the models tested the neural network model shows a good classification performance on different sets of feature combinations used.

For this research work a neural network model with single hidden layer having 25 neurons on the hidden layer, is selected and discussed as it gives a good classification performance on 35 ms speech segment. From all the feature vector combinations tested, the combination having energy, zero-crossing rate and 13 MFCC coefficients performs well on the selected architecture. The network architecture selection process is briefly described below.

The starting number for the hidden layer nodes was set to 8 and was increased from 8 to 30 for a single hidden layer network. Classification rates were obtained for these single hidden layer networks as well as for double hidden layer networks. Each network is trained by a set of 600 (out of 900) randomly selected training speech sentences and tested using 300 (out of 900) randomly selected training speech sentences. As described above, the network with the architecture of 15-25-3 (15 neurones on the input layer, one hidden layer with 25 neurons on the hidden layer and 3 neurons on the output layer) was a preferable choice in terms of the network simplicity and classification rate. The network classifier in this thesis work is implemented using weka machine learning tool.

The network was trained using the generalized delta rule for back propagation of error with a learning rate of $\alpha = 0.3$. A momentum term was added in updating the weights ($\beta = 0.2$). The training loop would not terminate until the total number of epochs to train through is 500. The input and output layers of the network had fixed number of neurons. There were 15 input layer neurons that matched the dimension of the feature vector (13 MFCC coefficients, Energy and zero-crossing rate). There were 3 neurons on the output layer that matched the dimension of the classes (voiced, unvoiced and silence). The overall architecture of the network, i.e., the number of hidden layer and the number of neurons per hidden layer, was a parameter determined in the experimental evaluation of the network. The network performance as a function of the frame size of the training set was also evaluated and compared to other classifiers.

The network with the architecture of 15-25-3 was a preferable choice in terms of the network simplicity and classification rate. In fact, the classification rate was not significantly altered when the number of neurons or the number of hidden layer was increased. This 15-25-3 network was used for comparing the performance of the network classifier and other classifiers.

4.3 Classifier Model Design Issues

4.3.1 Designing Multilayer Perceptron Voiced/Unvoiced/Silence Detection Model

A multilayer perceptron (MLP) is a feed-forward artificial neural network model that maps sets of input data onto a set of appropriate output [23]. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called *back-propagation* for training the network.

Back-propagation is a supervised learning method, and is a generalization of the delta rule [29]. It requires a supervisor that knows, or can calculate, the desired output for any input in the training set. It is most useful for feed-forward networks (networks that have no feedback, or simply, that have no connections that loop). The term is an abbreviation for "backward propagation of errors". Back-propagation requires that the activation function used by the artificial neurons (or "nodes") be differentiable.

In computational networks, the activation function of a node defines the output of that node given an input or set of inputs [1]. A standard computer chip circuit can be seen as a digital network of activation functions that can be "ON" (1) or "OFF" (0), depending on input. A general architecture of a multilayer perceptron is shown in Figure 4.7.

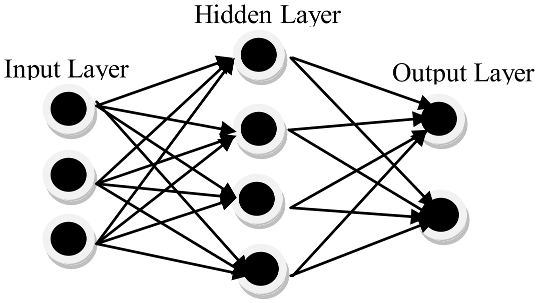


Figure 4.7: General architecture of a multilayer perceptron

Training Multilayer Perceptron Networks

The goal of the training process is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible [22]. There are several issues involved in designing and training a multilayer perceptron network:

- ***Selecting how many hidden layers to use in the network***

For most problems, one hidden layer is sufficient. Two hidden layers are required for modelling data with discontinuities such as a saw tooth wave pattern. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to local minima [17].

- ***Deciding how many neurons to use in each hidden layer***

One of the most important characteristics of a perceptron network is the number of neurons in the hidden layer(s). If an inadequate number of neurons are used, the network will be unable to model complex data, and the resulting fit will be poor. If too many neurons are used, the training time may become excessively long, and, worse, the network may over fit the data. When over-fitting occurs, the network will begin to model random noise in the data. The result is that the model fits the training data extremely well, but it generalizes poorly to new, unseen data. Validation must be used to test for this.

- ***Deciding how many neurons to use in the input and output layer***

On the other hand, the number of neurons in the input and output layers depend on the number of attributes used in the feature vector and the number of classes to be classified respectively. The number of attributes in the feature vector need not be notoriously redundant (much data, but not much information) since it affects the time required to complete the classification process. The number of classes used depends on the problem domain and it in turn determines the number of neurons in the output layer. Therefore, both the number of input components (features) and the number of output neurons is in general determined by the nature of the problem [17].

❖ ***Finding optimal weight***

A typical neural network might have a couple of hundred weights whose values must be found to produce an optimal solution. If neural networks were linear models like linear regression, it would be a breeze to find the optimal set of weights. But the output of a neural network as a function of the inputs is often highly nonlinear; this makes the optimization process complex.

Multilayer perceptrons use a gradient descent algorithm, called the error back-propagation (EBP) algorithm for learning and finding an optimal solution (weight). EBP is a method for calculating the first derivative of the error function with respect to each network weight.

4.3.2 Designing Decision Tree Voiced/Unvoiced/Silence Detection Model

In data mining, a decision tree is a predictive model which can be used to represent both classifiers and regression models. In operations research, on the other hand, decision trees refer to a hierarchical model of decisions and their consequences. When a decision tree is used for classification tasks, it is more appropriately referred to as a classification tree. When it is used for regression tasks, it is called regression tree [28].

A decision tree is a predictive machine-learning model that decides the target value (dependent variable) of a new sample based on various attribute values of the available data.

The attribute that is to be predicted is known as the dependent variable, since its value depends upon, or is decided by, the values of all the other attributes. The other attributes, which help in predicting the value of the dependent variable, are known as the independent variables in the dataset.

4.3.2.1 Simple CART

CART stands for Classification and Regression Trees. It is characterized by the fact that it constructs binary trees, namely each internal node has exactly two outgoing edges. An important feature of CART is its ability to generate regression trees [28]. *CART,* a recursive partitioning method, builds classification and regression trees for predicting continuous dependent variables (regression) and categorical predictor variables (classification) [10].

The process of computing classification and regression trees can be characterized as involving four basic steps [10]:

❖ ***Specifying the criteria for predictive accuracy***

The classification and regression trees algorithms are generally aimed at achieving the best possible predictive accuracy. Operationally, the most accurate prediction is defined as the prediction with the minimum costs. The notion of costs was developed as a way to generalize, to a broader range of prediction situations, the idea that the best prediction has the lowest misclassification rate.

❖ ***Selecting splits***

The second basic step in classification and regression trees is to select the splits on the predictor variables that are used to predict membership in classes of the categorical dependent variables, or to predict values of the continuous dependent (response) variable. In general terms, the split at each node will be found that will generate the greatest improvement in predictive accuracy. This is usually measured with some type of node impurity measure, which provides an indication of the relative homogeneity (the inverse of impurity) of cases in the terminal nodes. If all cases in each terminal node show identical values, then node impurity is minimal, homogeneity is maximal, and prediction is perfect (at least for the cases used in the computations; predictive validity for new cases is of course a different matter).

❖ ***Determining when to stop splitting***

In principal, splitting could continue until all cases are perfectly classified or predicted. However, this wouldn't make much sense since we would likely end up with a tree structure that is as complex and "tedious" as the original data file (with many nodes possibly containing single observations), and that would most likely not be very useful or accurate for predicting new observations. What is required is some reasonable stopping rule. In CART, two options are available that can be used to keep a check on the splitting process; namely Minimum n and Fraction of objects.

❖ ***Selecting the "right-sized" tree***

The size of a tree in the classification and regression trees analysis is an important issue, since an unreasonably big tree can only make the interpretation of results more difficult. Some generalizations can be offered about what constitutes the "right-sized" tree. It should be sufficiently complex to account for the known facts, but at the same time it should be as simple as possible. It should exploit information that increases predictive accuracy and ignore information that does not. It should, if possible, lead to greater understanding of the phenomena it describes. The options available in CART allow the use of either, or both, of two different strategies for selecting the "right-sized" tree from among all the possible trees. One strategy is to grow the tree to just the right size, where the right size is determined by the user, based on the knowledge from previous research, diagnostic information from previous analyses, or even intuition. The other strategy is to use a set of well-documented, structured procedures developed by Breiman et al. (1984) for selecting the "right-sized" tree.

4.3.2.2 J48

J48 is an open source Java implementation of the C4.5 algorithm in the Weka data mining tool. C4.5 is a well-known machine learning algorithm used widely, but its runtime performance is sacrificed for the consideration of the limited main memory at that time [14].

4.3.3 Designing Rule Based Voiced/Unvoiced/Silence Detection Model

4.3.3.1 Decision Table

Decision tables, like decision trees or neural nets, are classification models used for prediction. They are induced by machine learning algorithms. A decision table consists of a hierarchical table in which each entry in a higher level table gets broken down by the values of a pair of additional attributes to form another table [4].

A decision table representation with a default rule mapping to the majority class called DTM (Decision Table Majority) has two components a ***schema*** which is a set of features that are included in the table and a ***body*** consisting of labelled instances from the space defined by the features in the schema. Given an unlabelled instance a decision table classifier searches for exact matches in the decision table using only the features in the schema note that there may be many matching instances in the table. If no instances are found, the majority class of the DTM is returned otherwise the majority class of all matching instances is returned [18].

❖ ***Treatment of Attribute Values***

Attributes must have some set of discrete values for the classifier to be effective. If an attribute is continuous, the inducer uses entropy based discretization so that the distributions of classes in adjacent bins are as different as possible. This discretization is done globally. The ordering of the bins for the continuous attributes is explicit, but for categorical attributes the values can be ordered in three meaningful ways: alphabetically, numerically by record weights, or numerically by correlation with one of the classes to be predicted [4].

4.3.3.2 JRip

JRIP is a prepositional rule learner, i.e. Repeated Incremental Pruning to Produce Error Reduction (RIPPER). Initial rule set for each class is generated using IREP. The Minimum Description Length (MDL) based stopping condition is used. Once a rule set has been produced for each class, each rule is reconsidered and two variants are produced [25].

4.3.4 Summary

In this voiced/unvoiced/silence detection work, MATLAB 7.9, Audicity 1.3.13, wavesurfer 1.8.5 and Weka 3.6 machine learning are used as implementation tools due to their ease of application in speech and signal processing. They are easy to use and process speech with different integrated components.

A fairly general framework based on a neural network approach to voiced/unvoiced/silence classification has been described in which a set of measurements are made on the interval being classified, and an MLP classifier is used to select the appropriate class. Almost any set of measurements can be used as long as there is some physical basis for assuming that the measurements are capable of reliably distinguishing between these three classes.

In this paper, we described a method which uses a neural network approach for classifying a given speech segment into three classes: voiced speech, unvoiced speech, and silence. The neural network approach provides an effective method of combining the contributions of a number of speech measurements-which individually may not be sufficient to discriminate between the classes-into a single measure capable of providing reliable separation between the three classes.

CHAPTER FIVE

EXPERIMENTS AND PERFORMANCE ANALYSIS

5.1 Introduction

Different experiments have been conducted on the voiced/unvoiced/silence classifier. The whole speech corpus (900 sentences) is divided into two sets: the training set (600 sentences) and the testing set (300 sentences). The former one comprises 66.67% of the corpus while the remaining 33.33% are used for testing purpose.

An extensive search for an optimal classifier is undertaken to empirically select a classifier that has a simple architecture and reasonably high classification performance. Five different types of classifiers are tested in this research work namely; two rule-based (Decision Tables and JRip), two decision trees (J48 and simple CART) and a neural network (single and double hidden layer with different number of neurons on the hidden layer). The classification performance of each of these models is described below starting from Section 5.2.

Furthermore, as described in chapter 4 section 4.1.4 six different feature combinations have been tested for this thesis work. The features are energy, zero-crossing, 13 MFCC coefficients and 12-pole LPC coefficients. These combined features are described in Table 4.1.

An extensive search for an optimal classifier model, frame size and feature combination, for voiced/unvoiced/silence speech segment detection is conducted. In this chapter, the detail experiments conducted for this thesis work are discussed briefly.

5.2 Experiments with Selected Classifiers

In addition to the different experiments conducted with MLP voiced/unvoiced/silence classifier, the researchers have conducted other experiments with the same data set used for the MLP classifier. The other classifiers used include decision tree (J48, Simple CART) and rule based (JRip, Decision Tables). In fact in addition to these models other classifier models such as RBFNetwork and NaiveBayes are also tested but because of their poor classification performance they are omitted from this discussion. Table 5.1, 5.2, 5.3, 5.4, 5.5 and 5.6 show the performance (in percent) of these classifiers for the same dataset for feature vector one, feature vector two, feature vector three, feature vector four, feature vector five and feature vector six respectively for 20, 25, 30 and 35 millisecond frame size. Moreover, their corresponding performance curve is shown in Figure 5.1, 5.2, 5.3, 5.4, 5.5 and 5.6 respectively.

Table 5.1: Classifiers performance using feature vector one (see Table 4.1)

Energy & ZCR				
Classifier	20ms	25ms	30ms	35ms
Decision Table	74.26%	74.29%	74.32%	74.24%
Simple CART	74.29%	74.29%	74.29%	74.20%
Tree (J48)	74.28%	**74.33%**	74.32%	74.23%
Rules(Jrip)	74.26%	74.29%	74.23%	74.20%
MLP10[3]	74.26%	74.28%	**74.33%**	74.18%
MLP10,5[4]	74.26%	74.31%	74.29%	74.20%

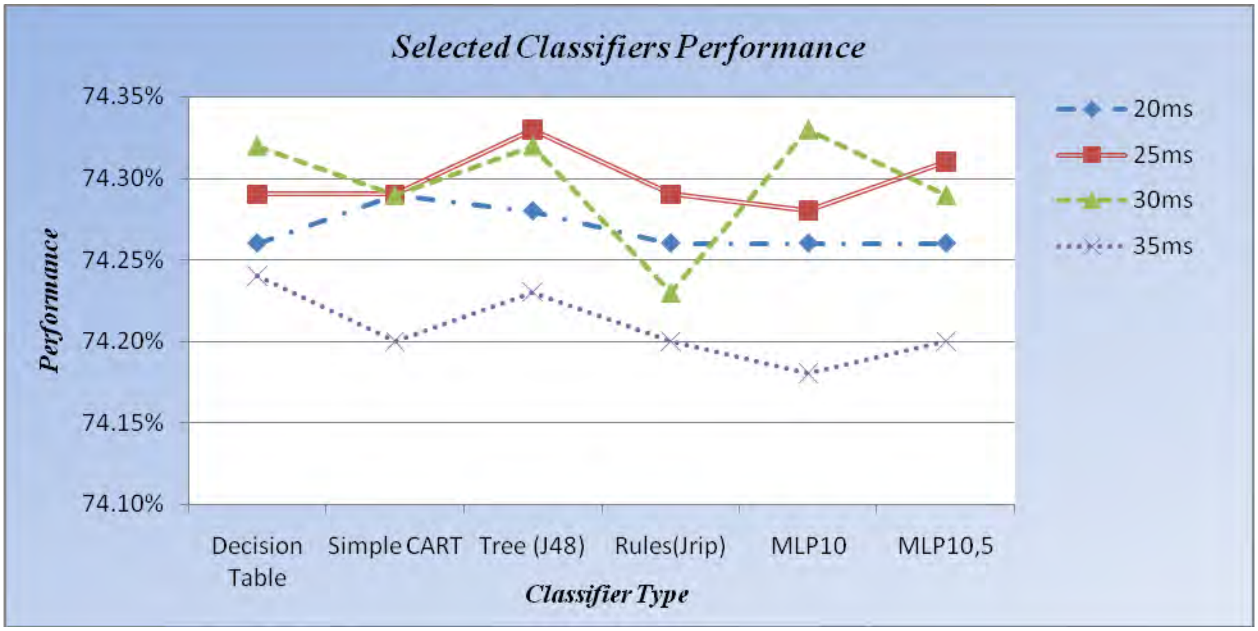


Figure 5.1: Classifiers performance curve analysis using feature vector one

Table 5.2: Classifiers performance using feature vector two

Energy, ZCR & LPC				
Classifier	20ms	25ms	30ms	35ms
Decision Table	81.80%	83.98%	82.94%	82.96%
Simple CART	83.58%	83.98%	84.17%	**84.29%**
Tree (J48)	82.60%	82.63%	83.69%	83.60%
Rules(Jrip)	82.60%	82.72%	83.54%	84.19%
MLP10	83.27%	84.30%	84.77%	84.30%

[3] Represents a neural network (MLP) with 10 neurons on the hidden layer (MLP 15-10-3).

[4] Represents a neural network (MLP) with 2 hidden layers with 10 and 5 neurons on the first and second hidden layer, respectively. (MLP 15-10-5-3).

MLP10,5	83.10%	84.16%	**85.11%**	84.16%

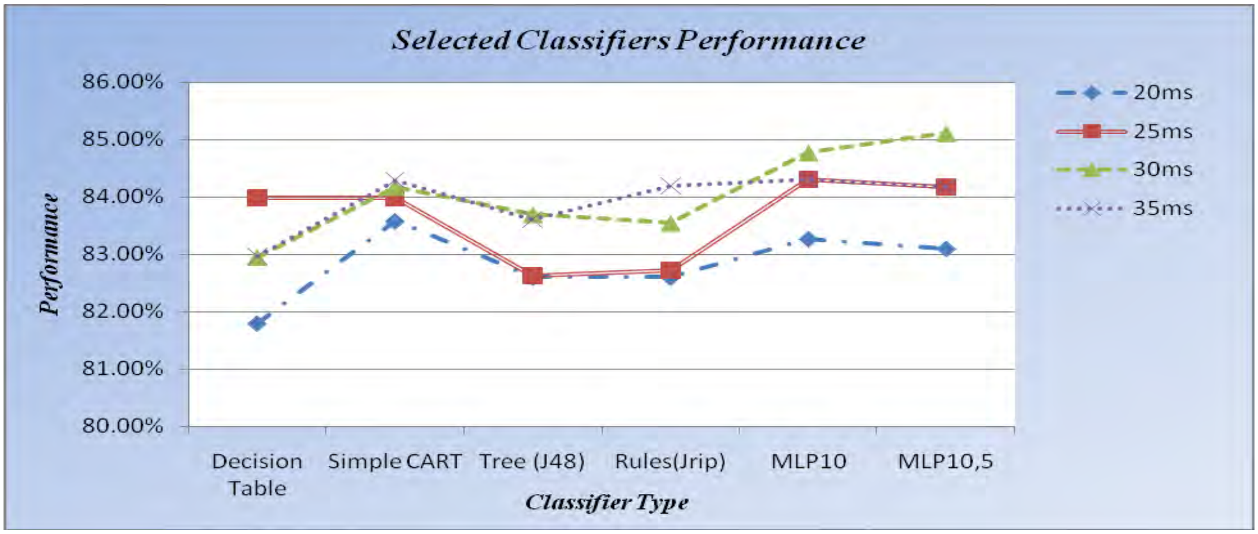


Figure 5.2: Classifiers performance curve analysis using feature vector two

Table 5.3: Classifiers performance using feature vector three

LPC				
Classifier	20ms	25ms	30ms	35ms
Decision Table	81.83%	82.47%	82.79%	82.96%
Simple CART	82.96%	83.42%	84.55%	84.03%
Tree (J48)	82.15%	82.62%	**87.67%**	83.15%
Rules(Jrip)	81.86%	81.86%	83.11%	83.63%
MLP10	82.80%	82.74%	**84.22%**	83.93%
MLP10,5	83.34%	83.22%	81.97%	83.95%

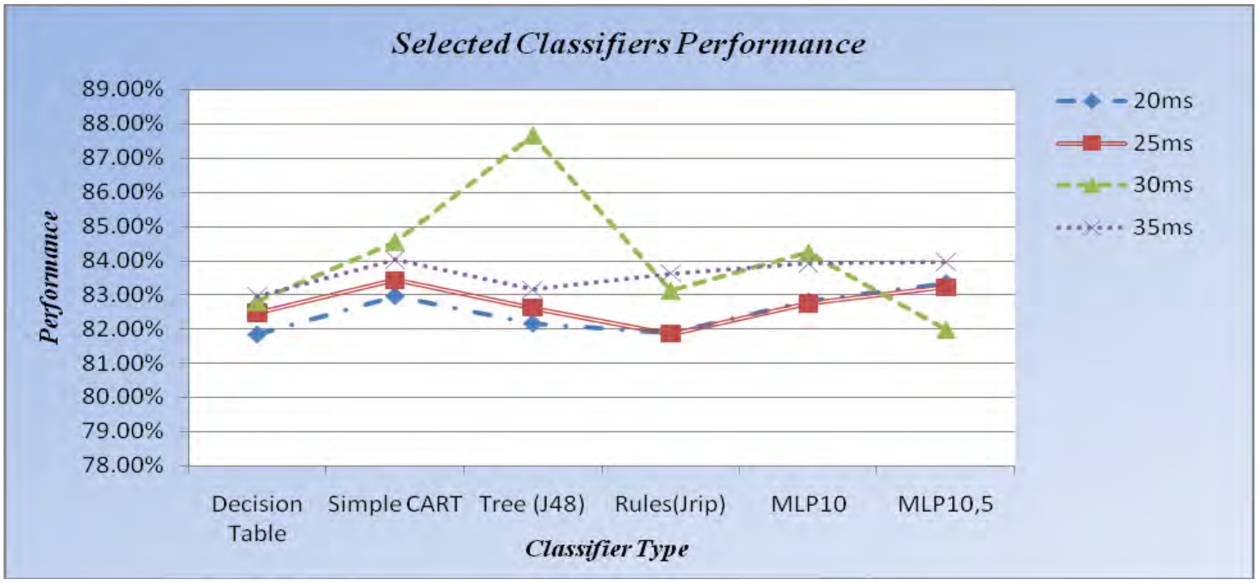


Figure 5.3: Classifiers performance curve analysis using feature vector three

Table 5.4: Classifiers performance using feature vector four

MFCC				
Classifier	20ms	25ms	30ms	35ms
Decision Table	86.07%	85.82%	86.02%	85.28%
Simple CART	**88.40%**	87.98%	88.08%	87.90%
Tree (J48)	87.60%	87.60%	87.32%	87.36%
Rules(Jrip)	-[5]	-[5]	87.94%	87.76%
MLP10	88.77%	88.78%	88.84%	**89.08%**
MLP10,5	88.64%	88.87%	88.80%	88.88%

[5] Indicates JRip requires a maximum (additional) java heap size than used for the other classifiers

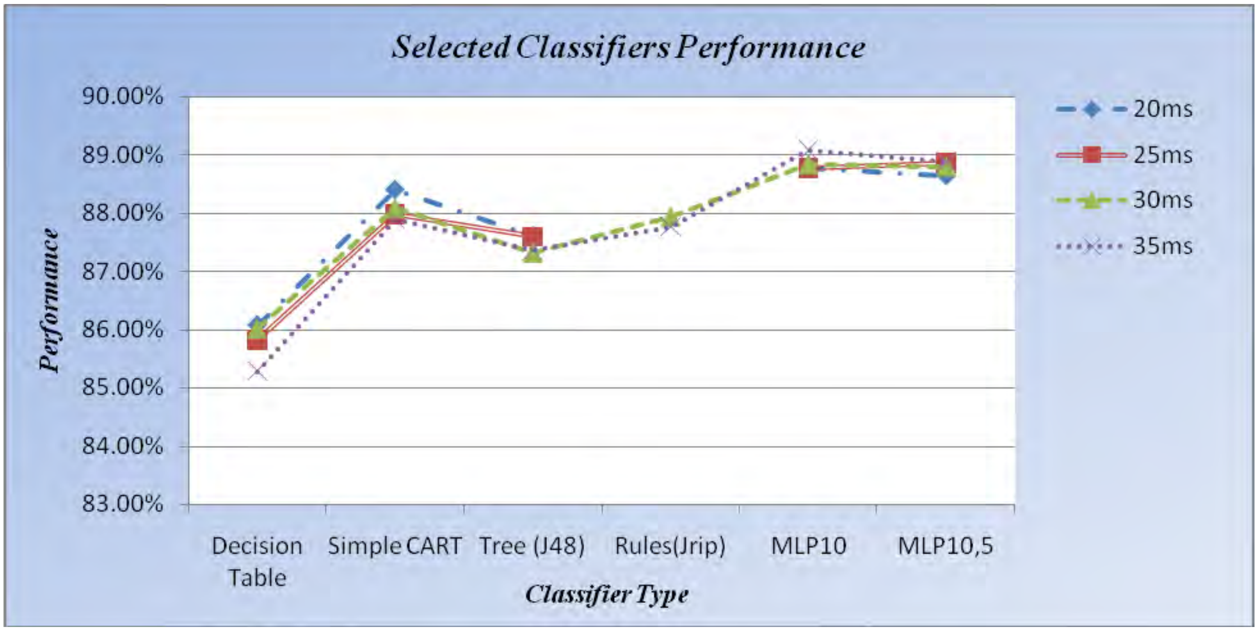


Figure 5.4: Classifiers performance curve analysis using feature vector four

Table 5.5: Classifiers performance using feature vector five

Energy, ZCR & MFCC				
Classifier	20ms	25ms	30ms	35ms
Decision Table	86.07%	85.82%	86.02%	85.28%
Simple CART	**88.41%**	88.14%	88.15%	87.83%
Tree (J48)	87.37%	87.34%	87.08%	87.25%
Rules(Jrip)	-[5]	-[5]	-[5]	87.62%
MLP10	88.64%	88.96%	88.79%	**89.33%**
MLP10,5	88.77%	88.96%	88.66%	89.01%

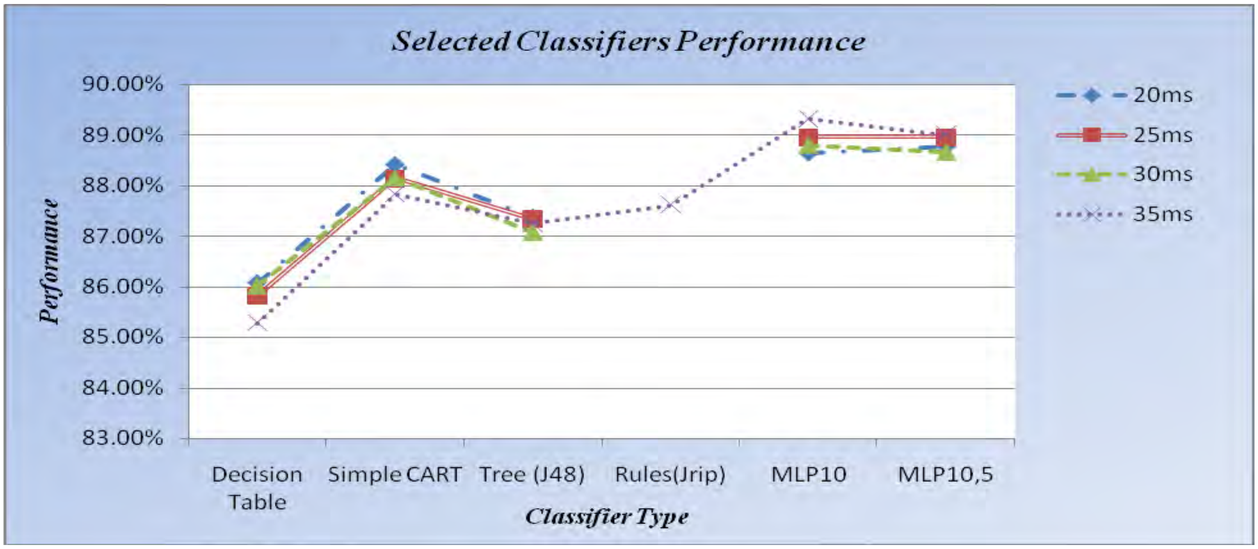


Figure 5.5: Classifiers performance curve analysis using feature vector five

Table 5.6: Classifiers performance using feature vector six

Energy, ZCR MFCC-PCA				
Classifier	20ms	25ms	30ms	35ms
Decision Table	84.99%	83.79%	83.74%	84.21%
Simple CART	**88.32%**	87.59%	87.40%	87.37%
Tree (J48)	87.20%	86.64%	86.36%	86.85%
Rules(Jrip)	-[5]	-[5]	87.26%	87.50%
MLP10	88.24%	88.57%	88.43%	88.57%
MLP10,5	**88.58%**	88.31%	88.24%	88.40%

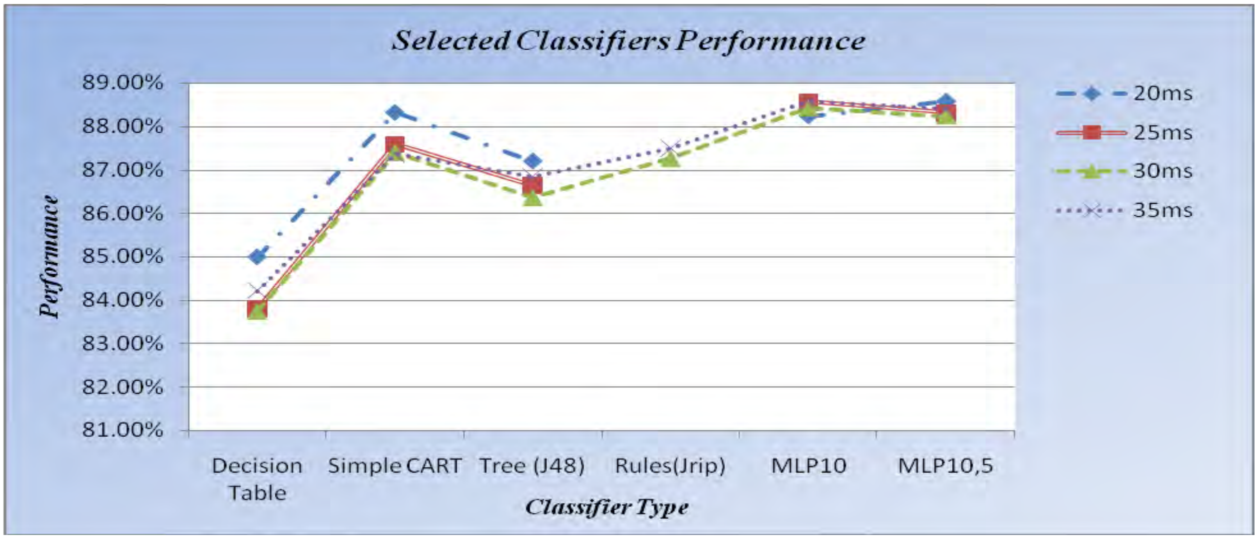


Figure 5.6: Classifiers performance curve analysis using feature vector six

As it can be seen on the learning curve on Figure 5.1, 5.2, 5.3, 5.4, 5.5 and 5.6 most of the classifiers perform better with feature vector containing MFCC coefficients than on feature vectors containing energy and ZCR alone or LPC.

Feature vector one containing energy and ZCR works well on 25 and 30 ms of speech segment using decision tree (J48) and MLP10 classifiers with a classification performance of **74.33%**. Feature vector two containing LPC coefficients works well on 30 ms of speech segment using decision tree (J48) classifier with a classification performance of **87.67%**. Feature vector three containing energy, ZCR and LPC coefficients works well on 30 ms of speech segment using MLP10-5 classifier with a classification performance of **85.11%**. Feature vector four containing MFCC works well on 35 ms of speech segment using MLP10 classifier with a classification performance of **89.08%**. Feature vector five containing energy, ZCR and MFCC coefficients works well on 35 ms of speech segment using MLP10 classifier with a classification performance of **89.33%**. Feature vector six containing energy, ZCR and MFCC coefficients after applying PCA works well on 20 ms of speech segment using MLP10-5 classifier with a classification performance of **88.58%**. Table 5.7 summarizes the classification performance of the classifiers as a function of features used.

Table 5.7: Performance Summary

Feature Vector	*1st Best Performance*	*2nd Best Performance*	*1st Best Model*	*2nd Best Model*
Energy & ZCR	74.33%	74.32%	MLP10,J48	DT,J48
LPC	87.67%	84.55%	J48	CART
Energy, ZCR & LPC	85.11%	84.77%	MLP10-5	MLP10
MFCC	89.08%	88.88%	MLP10	MLP10-5
Energy, ZCR & MFCC	89.33%	89.01%	MLP10	MLP10-5
Energy, ZCR & MFCC-PCA	88.58%	88.57%	MLP10-5	MLP10

As it can be seen from the performance summary on Table 5.7 the feature vector containing the combinations of energy, ZCR and 13 MFCC coefficients shows relatively high classification performance on 35 milliseconds of speech segment with the classification accuracy of **89.33%**. This classification accuracy is obtained using MLP10 classifier.

As it can be seen from performance summary on Table 5.7 a multilayer perceptron (MLP) classifier dominates the other classifiers tested. In view of this fact, other experiments were also conducted using MLP classifiers with different number of hidden layers and neurons on each hidden layer. From the MLP classifiers experimented, an MLP with one hidden layer having 25 neurons on the hidden layer, has shown better classification accuracy of **89.69%**. Further experiments conducted by tunning the parameters of the MLP classifier are conducted using only feature vector five due to the fact that it shows relatively high classification performance over the other feature vectors tested in section 5.2. A vivid description of the experiments conducted on MLP classifiers are documented below (see section 5.3).

5.3 Experiments with MLP Classifier

The weka multilayer perceptron function with some adjustments on the parameters is used for conducting experiments on the voiced/unvoiced/silence classifier. Different experiments are conducted on the MLP classifier by varying the frame size, number of hidden layer and the number of neurons per hidden layer using the training set to see the goodness of the classifier based on the observation that can be made on the learning curve on feature vector 5. The researchers have started training the system using an MLP with single hidden layer having 8 nodes and frame size of 20 millisecond using energy, ZCR and 13 MFCC coefficients as features. After the classifier is trained, its performance is measured on the testing set. Having got a low performance of the classifier trained with these parameters, the researchers‟ kept on adding the number of nodes per layer, changing the frame size until optimal performance of the classifier is obtained. In fact, the desired performance of the classifier is considered to be the performance measured from the learning curve shown in Figure 5.7. Table 5.8 shows the performance obtained for the different experiments conducted by changing the frame size and the number of hidden layers and neurons per hidden layer using feature vector 5 with the corresponding performance of the classifiers for a 20, 25, 30 and 35 millisecond speech segments. The corresponding performance curve is shown in Figure 5.7. The curve shows that the frame size, number of hidden layers and nodes per layer is almost sufficient. Moreover, the performance obtained shows the attainable performance of the MLP classifier.

In this section, the detail description of the experiments conducted and classification performance of the algorithm is presented.

Table 5.8: Single and Double Layer MLP classifier performance (%)

	8	***10***	***15***	***20***	***25***	***30***	***10,5***	***12,5***	***15,5***	***20,5***	***20,10***	***20,15***
20ms	82.22	88.64	88.97	89.04	89.15	**89.27**	88.77	88.76	88.94	89.09	89.24	89.05
25ms	88.76	88.96	89.13	89.20	89.14	**89.53**	88.96	89.13	89.21	89.13	89.27	89.21
30ms	88.33	88.79	89.01	89.14	89.09	**89.21**	88.66	88.70	89.07	89.16	89.10	89.14
35ms	88.80	89.33	89.41	89.64	**89.69**	89.63	89.01	89.18	89.34	89.36	89.38	89.36

The numbers on the column headers represent the number of neurons per hidden layer. For example, 8 is an MLP with single hidden layer and 8 neurons on the hidden layer and 10,5 represents an MLP with two hidden layers having 10 and 5 neurons on the first and second hidden layers respectively.

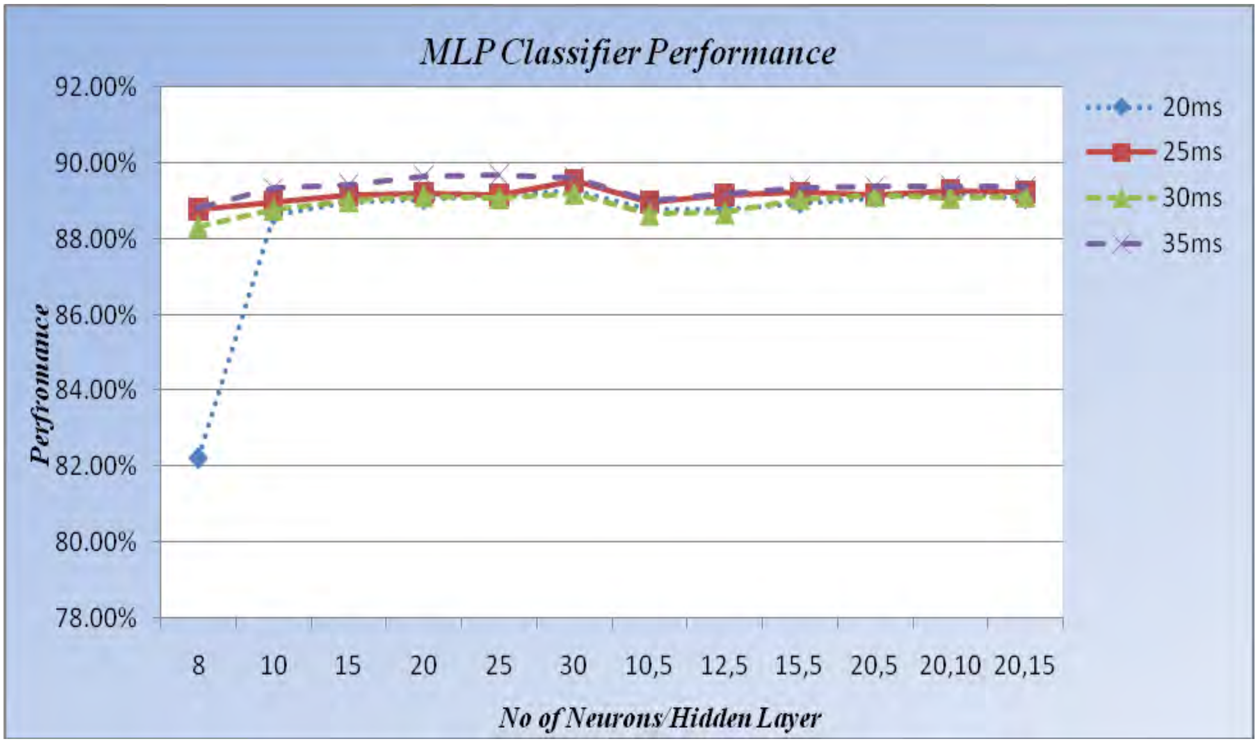


Figure 5.7: Single and Double Layer MLP classifier performance curve analysis

The experiment conducted on 20 ms of speech segment shows a relatively high classification performance on MLP30 (MLP with single hidden layer and 30 neurons on the hidden layer) classifier with a classification performance of **89.27%**. The second frame length, 25 ms speech segment works well on MLP30 classifier with a classification performance of **89.53%**. Experiments conducted on 30 ms speech segment show a relatively high classification performance on MLP30 classifier with a classification performance of **89.21%**. The last parameter tunning experiment conducted on 35 ms of speech segment shows a relatively high classification performance on MLP25 classifier with a classification performance of **89.69%**. In fact, this is the highest classification accuracy obtained from all the experiments conducted for this thesis work. The learning rate used for all the MLP classifiers tested is α=0.3. Table 5.9 summarizes the classification performance of the classifiers as a function of the frame size used.

Table 5.9: Single and Double Layer MLP Performance summary (α=0.3)

Learning Rate (α=0.3)		
Frame Size	***Best Performance***	***Model Type***
20 millisecond	89.27%	MLP30
25 millisecond	89.53%	MLP30
30 millisecond	89.21%	MLP30
35 millisecond	89.69%	MLP25

Furthermore, another experiment is conducted on the above top four high performance classifiers by changing the learning rate (α) of the MLP classifier from 0.3 to 0.5. The results obtained for this experiment is summarized in Table 5.10.

Table 5.10: Single and Double Layer MLP Performance summary (α=0.5)

Learning Rate (α=0.5)		
Frame Size	***Best Performance***	***Model Type***
20 millisecond	89.15%	MLP30
25 millisecond	89.36%	MLP30
30 millisecond	89.09%	MLP30
35 millisecond	89.25%	MLP25

As it can be seen from the performance summary on Table 5.10 the performance of all the classifiers decreases as the learning rate is increased from 0.3 to 0.5.

A detailed analysis on the performance of the MLP classifier (MLP25) that has shown high classification accuracy is indicated in section 5.4. Moreover, a detailed accuracy by class and the confusion matrix of this classifier is illustrated.

5.4 Performance Analysis

In order to analyse the performance of the MLP25 classifier for the different sound categories, the frequency of the sound categories in the entire corpus, training set and testing set is considered. Moreover, confusion matrix is developed for this voiced/unvoiced/silence classifier.

5.4.1 Feature Analysis

The most popular features that are used for voiced/unvoiced/silence speech segment classification include, Mel Frequency Cepstral Coefficients (MFCC), Linear Predictive (LP) Coefficients, short-time energy, short-time zero-crossing rate, average magnitude, and auto correlation function.

In this research work, we have used a combination of four features for our experiment containing; short-time energy, short-time zero-crossing rate, 13 MFCC coefficients and a 12-pole LPC coefficient (13 LPC coefficients). Moreover, these features are extracted for each 20, 25, 30 and 35 ms of the speech segment.

First we conducted our experiment on the feature vector containing the combination of energy and ZCR. This resulted with a maximum classification accuracy of 74.33%. Next an experiment is conducted on the feature vector containing 13 LPC coefficients resulting with a maximum classification accuracy of 87.67%. The third experiment conducted contains the combinations of energy, ZCR and 13 LPC coefficients which resulted with a maximum classification accuracy of 85.11%. The next experiment is conducted on a feature vector containing 13 MFCC coefficients resulting with a classification accuracy of 89.08%. After wards an experiment is conducted on a feature vector containing the combination of features energy, ZCR and 13 MFCC coefficients which resulted with a classification accuracy of 89.69%. Finally, the last experiment is conducted using the same feature vector with experiment five, except that a principal component analysis is performed on the feature vector resulting with a classification accuracy of 88.58%. Table 5.11 summarizes the maximum performance obtained from the experiment conducted on each of the features used regardless of the frame size and classifier model used.

Table 5.11: Feature vector performance summary

Feature Vector	Performance
Energy & ZCR	74.33%
LPC	87.67%
Energy, ZCR & LPC	85.11%
MFCC	89.08%
Energy, ZCR & MFCC	89.69%
Energy, ZCR & MFCC-PCA	88.58%

As it can be seen from Table 5.11, feature combination of energy and ZCR shows the least classification accuracy: 74.33%. This is due to the fact that the value of any of these parameters usually overlaps between categories, particularly when the speech is not recorded in a high fidelity environment. On the other hand, all the feature vectors containing MFCC values show high classification accuracy, specifically the feature vector containing the combinations of energy, ZCR and 13 MFCC coefficients: 89.69%. The advantage of the MFCC approach has been an automatic way to reduce the amount of information in a Fourier transform (FT) of a frame of speech (which is always assumed to reasonably capture the essential information about vocal tract shape at any specific point in time) to a small set of parameters, e.g., 10–16. The data reduction factor is about the same as for LPC, except that the MFCC is able to utilize some auditory factors in warping frequency scales to model the human ear better than LPC can [16].

5.4.2 Frame Size Analysis

Table 5.12 summarizes the maximum performance obtained from the experiment conducted on different frame sizes (20, 25, 30, and 35) in millisecond regardless of the frame features and classifier models used.

Table 5.12: Frame size performance summary

Frame Size	Performance
20	89.27%
25	89.53%
30	89.21%
35	89.69%

As shown on Table 5.12, from the four frame sizes experimented, the experiment on 20 millisecond speech segments shows the least classification accuracy of 89.27%. On the other hand the experiment conducted on 35 millisecond speech segment shows the highest classification accuracy of 89.69%. This is due to the fact that small frame sizes might not be able to capture all the required measurements (missing necessary details) in order to map a speech frame to a specific category.

5.4.3 Selected MLP Classifier Model Analysis

The performance of the MLP classifier (MLP25) illustrated in this analysis is obtained on a feature vector containing energy, ZCR and 13 MFCC coefficients. In addition, from the entire frame sizes tested (20, 25, 30 and 35 ms)[6], the performance described below for this classifier is obtained on 35 ms speech segment.

The classification summary of the MLP (15-25-3) classifier is stated below along with a detailed accuracy by class and a confusion matrix shown on table 5.13 and table 5.14 respectively.

```
Time taken to build model: 6512.76 seconds
=== Evaluation on test set ===
=== Summary ===
Correctly Classified Instances       36879          89.6906 %
Incorrectly Classified Instances     4239           10.3094 %
Kappa statistic                      0.7379
Mean absolute error                  0.0932
Root mean squared error              0.2329
Relative absolute error              34.1071 %
Root relative squared error          63.1707 %
Total Number of Instances            41118
```

Table 5.13: Detailed Accuracy by Class (MLP25)

	TP Rate	*FP Rate*	*Precision*	*Recall*	*F-Measure*	*ROC Area*	*Class*
	0.778	0.011	0.827	0.778	0.802	0.978	Silence
	0.956	0.225	0.924	0.956	0.94	0.936	Voiced
	0.709	0.043	0.799	0.709	0.751	0.919	Unvoiced
Weighted Avg.	0.897	0.176	0.894	0.897	0.895	0.935	

[6] Frame size between 20-35 milliseconds is selected because during such a time humans cannot significantly change the shape of the vocal tract. And no analysis is made in determining the incremental value between the frame sizes used.

Table 5.14: Confusion Matrix (MLP25)

<table>
<tr><th><em>a</em></th><th><em>b</em></th><th><em>c</em></th><th><em>&lt;-- classified as</em></th></tr>
<tr><td>2055</td><td>304</td><td>281 |</td><td><strong><em>a = Silence</em></strong></td></tr>
<tr><td>194</td><td>29173</td><td>1142 |</td><td><strong><em>b = Voiced</em></strong></td></tr>
<tr><td>237</td><td>2081</td><td>5651 |</td><td><strong><em>c = Unvoiced</em></strong></td></tr>
</table>

The MLP classifier summary shows that it classifies 36,879 instances correctly and 4,239 instances incorrectly in the testing set. The performance of the classifier varies for the different sound classes with a higher performance for voiced sounds followed by silence sounds and unvoiced sounds for the given testing set trained on the training set as shown on table 5.13 and 5.14.

5.5 Summary

Different experiments are conducted for the voiced/unvoiced/silence speech segment classifier. Accordingly, different performances are obtained: the MLP classifier with a single hidden layer having 25 neurons on the hidden layer performs better than the other classifiers. The performance of the MLP voiced/unvoiced/silence classifier is **89.69%**.

CHAPTER SIX

CONCLUSION AND RECOMMENDATION

6.1 Conclusion

The interest in speech segment category (voiced/unvoiced/silence) discrimination has intensified lately due to the increasing demand for potential use in commercial or non-commercial systems and a number of speech processing systems. Current personal communication systems such as a cellular phone are examples of commercial systems that integrate speech coding and speech recognition capabilities in their operation. These systems normally require voice commands to control them. The spoken commands need to be accurately extracted from the background to process them.

Assigning speech categories to speech segment in a speech sound is an important component of many speech processing systems. An accurate classification of a speech segment as voiced/unvoiced/silence with voicing detection system is often used as a prerequisite for developing other higher level and efficient applications of speech processing systems such as speech coding, speech analysis, speech synthesis, automatic speech recognition, noise suppression and enhancement, pitch detection, speaker identification, and the recognition of speech pathologies.

Voiced/unvoiced/silence speech segment detection is a method of assigning and labelling a specific speech category (voiced/unvoiced/silence) to a speech segment to discriminate a speech sound from a background noise based on acoustic characteristics of the speech segment. It is a hot research area in the field of natural language processing for different languages. Moreover, voiced/unvoiced/silence speech segment detection can be conceived as the problem of assigning speech categories to a speech segment in a sentence. This problem can be solved using different techniques among which an Artificial Neural Network (ANN) specifically a Multilayer Perceptron (MLP) approach is assumed to perform better than other approaches.

A multilayer perceptron with a single hidden layer having 25 neurons on the hidden layer is designed for voiced/unvoiced/silence speech segment detection. The choice behind the MLP classifier is, it performs better than the other models experimented.

Data corpus is an important component in natural language processing in general and voiced/unvoiced/silence speech segment detection in particular. And hence, a corpus with a total of 900 sentences is collected from different Amharic text sources. Three speech categories are identified as tag sets that are used in annotating these total sentences to create an annotated corpus for training the MLP classifier as a supervised learning approach is used.

The corpus is divided into two (training set and testing set) for testing and training purpose. The training set consists 67.67% of the corpus (600 sentences) and the testing set consists 33.33% of the corpus (300 sentences). Matlab 7.9, Audicity 1.3.13, wavesurfer 1.8.5 and weka 3.6 software tools are used in the implementation and experiment of the voiced/unvoiced/silence speech segment classifier. Five different types of classifiers are tested in this research work namely; two rule-based (Decision Tables and JRip), two decision tree (J48 and simple CART) and a neural network (single and double hidden layer with different number of neurons on the hidden layer). Accordingly, 86.07%, 88.41%, 87.67%, 87.94% and 89.69% performances are obtained for Decision table, simple CART, J48, JRip and MLP25 classifiers respectively. Therefore, it is possible to conclude that a multilayer perceptron with single hidden layer having 25 neurons on the hidden layer performs better than other classifier. In addition, this research work has got promising results on the experimented five models.

Table 6.1: Classifier performance summary

Classifier	Performance (%)
Decision Table	86.07%
Simple CART	88.41%
Decision Tree (J48)	87.67%
Rules(Jrip)	87.94%
MLP25	89.69%

6.2 Recommendation

There are lots of research areas in natural language processing that can be done for different languages in Ethiopia. The same thing holds true for Amharic language. Therefore, to assist researchers, it will be of great paramount if a standard speech corpus for Amharic language is developed that will be available for NLP researchers in Amharic language.

Finally this research work suggests the following items as a future work:

- Comparative study of three different approaches (HMM based, rule based, ANN based classifiers with more training and testing data)
- Extending this work by training in large corpus and using balanced tag-sets for the different sound categories
- Comparison of hybrid approaches
- Experimenting phoneme identification following voiced/unvoiced/silence recognition
- Modelling and researching ANN based speech recognition system.

References

[1] Activation Function. Retrieved November 07, 2011, from http://en.wikipedia.org/wiki/Activation_function

[2] Atal, B. S., & Rabiner, L. R. (1976). A Pattern Recognition Approach to Voiced/Unvoiced/Silence Classification with Application To speech Recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-24, NO. 3.

[3] Bachu, R. G., Kopparthi, S., Adapa, B., & Barkana, B. D. (2010). Voiced/Unvoiced Decision for Speech Signals Based on Zero-Crossing Rate and Energy. USA.

[4] Becker, B. G. (2011). Visualizing Decision Table Classifiers. Shoreline Blvd, MS-500 Mountain View, CA 94043-1389.

[5] Beritelli, F., Casale, S., Russo, A., & Serrano, S. (2009). Adaptive V/UV Speech Detection Based on Characterization of Background Noise.

[6] Camastra, F., & Vinciarelli, A. (2008). Machine Learning for Audio, Image, and Video Analysis Theory and Applications. London: Springer-Verlag London Limited.

[7] Crystal, D. (2008). A dictionary of linguistics and phonetics (6 ed.). USA, UK, Australia Blackwell Publishing Ltd.

[8] David., G. (2003). Technical Report: Pitch Extraction and Fundamental Frequency- History and Current Techniques. Regina, Saskatchewan, CANADA

[9] Dhananjaya, N., & B.Yegnanarayana. (2010). Voiced/nonvoiced detection based on robustness of voiced epochs IEEE Signal Processing Letters.

[10] Electronic Statistics Textbook. (2011). Retrieved February 08, 2012, from http://www.statsoft.com/textbook

[11] Elleithy, K. (2010). Advance Techniques in Computing Sciences and Software Engineering. London, New York: Springer Science+Business Media B.V.

[12] Gebregziabher, T. (2010). Part of Speech Tagger for Tigrigna Language. Master‟s Thesis, Addis Ababa University.

[13] Getahun, A., *ዘመናዊ የአማረኛ ሰዋስው በቀላል አቀራረብ*. (1989). Addis Ababa: Commercial printing press.

[14] HE, P., CHEN, L., & XU, X. H. (2007, 19-22 August). FAST C4.5. Paper presented at the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong.

[15] Holmes, J., & Holmes, W. (2001). Speech Synthesis and Recognition (2 ed.). London, US, Canada: Taylor & Francis.

[16] Jacob Benesty, M. M. S., Yiteng Huang. (2008). Springer Handbook of Speech Processing. Springer-Verlag Berlin Heidelberg.

[17] Kecman, V. (2001). Learinig and Soft Computing Support Vector Machines, Neural Networks and Fuzzy Logic Models. Cambridge, Massachusetts: The MIT Press.

[18] Kohavi, R. (1995). The Power of Decision Tables. Paper presented at the European Conference on Machine Learning (ECML).

[19] Ladefoged, P. (2001). A course in Phonetics (4 ed.). USA: Heinle & Heinle, a Division of Thomson Learning Inc.

[20] Lodge, K. (2009). A Critical Introduction to Phonetics. New York: Continuum International Publishing Group.

[21] Meddins, B. (2000). Introduction to Digital Signal Processing. Oxford: Newness an Imprint of Butterworth-Heinemann.

[22] Multilayer Perceptron Neural Networks. Retrieved November 07, 2011, from http://www.dtreg.com/mlfn.htm

[23] Multilayer Perceptron. Retrieved November 07, 2011, from http://en.wikipedia.org/wiki/Multilayer_perceptron

[24] Qi, Y., & Hunt, B. R. (1993). Voiced-Unvoiced-Silence Classification of Speech Using Hybrid Features and a Network Classifier. IEEE Transactions on Speech and Audio Processing, Vol. 1, NO. 2,.

[25] R.P.Datta, & Saha, S. (2011). An Empirical comparison of rule based classification techniques in medical databases. In R. Bhattacharyya (Ed.), Working Paper Series. New Delhi, Kolkata: Indian Institute of Foreign Trade.

[26] Rabiner, L. R., & Sambur, M. R. (2003). Voiced-Unvoiced-Silence Detection Using the Itakura LPC Distance Measure. Murray Hill, New Jersey 07974.

[27] Rabiner, L. R., & Schafer, R. W. (1978). Digital Processing of Speech Signals. USA: Prentice Hall International Inc.

[28] Rokach, L., & Maimon, O. (2008). Data Mining with Decision Trees: Theory and Applications (Vol. 69). Singapore: World Scientific Publishing Co. Pte. Ltd.

[29] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning Internal Representations by Error Propagation (Vol. 1: Foundations): MIT Press.

[30] Speech, Music and Hearing part of School of Computer Science and Communication. Retrieved November 08, 2011, from http://www.speech.kth.se/wavesurfer/index2.html

[31] Tatarinov, J., & Pollak, P. Hidden Markov Models in voice activity detection. Czech Republic

[32] The Free, Cross-Platform Sound Editor. Retrieved November 08, 2011, from http://audacity.sourceforge.net/

[33] Witten, I. H., Eibe, F., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques (3 ed.). 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA Morgan Kaufmann Publishers is an imprint of Elsevier.

Appendices

Appendix A: Sample Amharic text corpus

1. የነቀምቴ ስታድየም ግንባታ ስልሳ በመቶ ተጠናቀቀ::
2. አጸደ በዱባይ ማራቶን ለድል ከሚጠበቁት አትሌቶች አንድዋ ነች::
3. ጥሩነሽ በዔደንብራ አገር አቋራጭ ውድድር አሸነፈች::
4. በኢትዮጵያ የኔትቦል ስፖርት እንዱስፋፋ እንግሊዝ ትደግፋለች::
5. የመቀሌ ስታድየም የመጀመሪያው ምአራፍ ግንባታ ተጠናቀቀ::
6. ኃይሉ በቶኪዮ ማራቶን አሸነፈ::
7. ጠይባ በቦስተን ማራቶን ለድል ትጠበቃለች::
8. የፌደራል ማረሚያ ቤቶች ስፖርት ክለብ አዲስ የስራ አስኪያጅ ኮሚቴ መረጠ::
9. በቃና የሂዩስተን ማራቶንን ክብረ ወሰን በማሻሻል አሸነፈ::
10. ኃይሌ በማንቸስተር የጎዳና ሩጫ ለድል ይጠበቃል::
11. ብዙነሽ በሙምባይ ማራቶን ለድል ትጠበቃለች::
12. ፌደሬሽኑ ባጸደቃቸው መመሪያዎች ከባለድርሻ አካላት ጋር ተወያየ::
13. ኃይሌ በኒውዮርክ ግማሽ ማራቶን ውድድር ይካፈላል::
14. ሁነኛው በስፔን የአገር አቅዋራጭ ውድድር አሸነፈ::
15. በላሊበላ ከተማ ታላቁ ሩጫ ተካሄደ::
16. ከአፍሪካ ሃያ አምስት ምርጥ ስፖርተኞች ኢትዮጵያውያን ግንባር ቀደም ስፍራ ይዘዋል::
17. ገንዘቤ በጌንት የአንድ ሺ አምስት መቶ ሜትር ሩጫ አሸነፈች::
18. ሲራጅ በሮም የማራቶን ውድድርን በባድ እግሩ በማጠናቀቅ ታሪክ አስመዘገበ::
19. ኢትዮጵያ ለዶኃ ውድድር በመሰረት ደፋር የሚመራ ቡድን ትልካለች::
20. ታዋቂ አትሌቶች በሚገኙበት በኃዋሳ ከተማ የሩጫ ውድድር ይካሄዳል::
21. ለወልድያ ስታድየም ግንባታ የሚውል ገቢ ማሰባሰብ ተጀመረ::
22. ፀጋየ ከበደ በለንደን ማራቶን አሸነፈ::
23. ኃይሌ የታላቁ ማንቸስተር ሩጫ ውድድርን አሸነፈ::
24. ቀነኒሳ በዶኃው የዳይመንድ ሊግ ውድድር አይሳተፍም::
25. ኢትዮጵያ በኦሎምፒክ ለመሳተፍ ዝግጅት እያደረገች ነው::
26. በአገር አቀፍ ደረጃ በተለያዩ የስፖርት አይነቶች ስልጠና እየተሰጠ ነው::
27. ስፔን የአለም ዋንጫን ያሸነፈችባት ኩዋስ በጨረታ ሰባ አራት ሺህ ድላር አወጣች::
28. ለአለም ዋንጫ በኮከብነት አስር ተጫዋቾች ታጩ::
29. ኢትዮጵያ በአስራ ሶስተኛው የአለም ወጣቶች ሻምፒዮና የአምስተኛ ደረጃን አገኘች::
30. ኢትዮጵያ በሞስኮ በተካሄደ የሩጫ ውድድሮች አሸነፉ::
31. የኢትዮጵያ ብሄራዊ የእግር ክዋስ ቡድን አሰልጣኝ እንግሊዛዊ ነው::
32. አመታዊ የአዲስ አበባ የክለቦች ብስክሌት ሻምፒወና የፍጻሜ ውድድር ተካሄደ::
33. የአዲስ አበባ የዱላ ቅብብል ውድድር ሰኔ ላይ ይካሄዳል::
34. የአዳማ ዩኒቨርስቲ ለታዳጊ ወጣቶች የስፖርት ስልጠና እየሰጠ ነው::
35. የአለም የወጣቶች አሎምፒክ ሻምፒዮና በሲንጋፖር እየተካሄደ ነው::
36. ታሪኩ በበርሊን የሶስት ሺህ ሜትር አሸነፈ::
37. ታደሰ በሊዝበን ግማሽ ማራቶን አሸነፈ::
38. ደቡብ አፍሪካ ያስተናገደችው የአለም ዋንጫ ታላቅ ውጤት የተመዘገበበት እንደነበር ፊፋ ገለጠ::
39. የመቀሌ ስታድየም እድሳት እየተደረገለት ነው::
40. መሰረት የአመቱ ምርጥ አትሌትነት ምርጫን በመምራት ላይ ነች::
41. የኢትዮጵያ እግር ክዋስ ፌደሬሽን ጠቅላላ ጉባዔ ተጀመረ::
42. የጋራ ብልጥግና አገሮች የስፖርት ሻምፒዮና ተጀመረ::

43. የዘንድሮው የታላቁ ሩጫ ውድድር ምዝገባ ተጠናቀቀ::
44. መንግስቱ ወርቁ ከዚህ አለም በሞት ተለየ::
45. አዝመራውና ሱሌ የታላቁ ሩጫ ውድድርን አሸነፉ::
46. ኃይሌ ሩጫ የሚያቆምበትን ትክክለኛ ጊዜ እንደማያውቀው ተናገረ::
47. ፌደሬሽኑ ብጥብጥ ባስነሱ ወገኖች ላይ ተገቢውን እርምጃ እንደሚወስዴ ገለጠ::
48. ኃይሌ የአለማችን የምንጊዜም ምርጥ ወንድ አትሌትነትን ምርጫ በድምጥ ብልጫ እየመራ ነው::
49. አቡትሪካ የአፍሪካ የአመቱ ኮከብ ተጫዋች ተብሎ ተሸለመ::
50. ሀገር አቀፉ የከፍተኛ ትምህርት ተቅዋማት የስፖርት ፌስቲቫል በጎንደር ተጀመረ::

Appendix B: Sample Feature Vector

Energy	*ZCR*	*Gain*	*LPC1*	*LPC2*	*LPC3*	*LPC4*	*LPC5*	*LPC6*	*LPC7*	*LPC8*	*LPC9*	*LPC10*	*LPC11*	*LPC12*	*Class*
596.04	0	1	-1.002	0.002	0.001	0.0005	0.0005	-9E-04	0.0003	-3E-04	-0.002	-0.002	-4E-04	0.0037	Silence
593.65	0	1	-0.999	0.0012	-0.003	0.0038	-2E-04	0.0008	-0.003	-0.005	0.0036	-0.003	0.0009	0.0034	Silence
597.69	0	1	-1.002	0.0052	-0.003	0.0012	0.0011	-0.003	-7E-04	-0.001	0.0006	-1E-03	-2E-05	0.0039	Silence
594.38	0	1	-1.001	0.0017	-8E-04	0.0034	-0.004	-7E-04	0.0017	-0.003	-0.002	2E-05	0.0041	0.0009	Silence
599.22	0	1	-1.003	0.0049	-0.003	0.0023	0.0014	-0.003	-4E-04	-0.003	7E-05	0.0032	-0.001	0.0029	Voiced
610.66	0	1	-1.183	0.0801	0.049	0.0506	0.0218	0.0014	0.0076	-0.005	-0.001	-0.011	-0.022	0.0127	Voiced
691.67	50	1	-1.952	0.8818	0.0998	0.0643	-0.018	-0.025	0.0153	-0.028	-0.044	-0.014	-0.013	0.0383	Voiced
723.64	46	1	-2.153	1.5606	-0.535	0.2278	-0.034	-0.011	-0.073	0.1048	-0.035	-0.086	0.065	-0.028	Voiced
750.87	58	1	-1.856	0.8781	0.0446	0.0342	-0.03	-0.052	0.0016	0.092	-0.04	-0.108	0.0364	0.0035	Voiced
737.24	60	1	-1.84	0.8201	0.1123	0.03	-0.056	-0.071	0.0352	0.0586	-0.035	-0.073	-0.022	0.0446	Voiced
669.33	40	1	-1.76	0.6775	0.1706	0.0454	-0.086	-0.038	0.04	-0.018	-0.013	0.0131	-0.098	0.0708	Voiced
611.37	12	1	-1.484	0.362	0.1574	0.0461	-0.012	0.0259	-0.005	-0.039	-0.006	-0.009	-0.016	-0.018	Voiced
581.56	0	1	-1.015	0.0078	0.006	0.0044	0.0063	0.0036	-0.002	-9E-04	-0.002	-0.005	-0.002	5E-05	Unvoiced
566.44	0	1	-1.147	0.0596	0.0592	0.0565	0.0506	0.0387	0.0078	-0.014	-0.031	-0.044	-0.03	-0.005	Unvoiced
568.51	0	1	-1.369	0.1447	0.1411	0.1108	0.0574	0.0199	-0.009	-0.031	-0.038	-0.034	-0.007	0.0153	Voiced
575.9	28	1	-1.897	0.7123	0.2463	0.0458	0.0154	0.0068	-0.07	-0.123	-0.077	0.0411	0.2157	-0.116	Voiced
610.32	22	1	-1.667	0.4131	0.1819	0.1022	0.0986	0.0411	-0.077	-0.131	-0.067	0.0119	0.1024	-0.007	Voiced
570.01	0	1	-1.093	0.0071	0.0186	0.0327	0.0333	0.0223	0.0042	0.0012	-0.004	-0.024	-0.018	0.0204	Unvoiced
577.11	0	1	-1.004	0.0018	0.0029	0.0023	0.0012	-0.003	-0.002	0.0027	-0.003	0.0008	0.0009	-3E-04	Unvoiced
574.49	0	1	-1.001	0.0026	-0.002	0.0051	-0.004	-1E-05	0.0025	-0.005	0.0033	-0.003	-3E-04	0.0019	Unvoiced
573.39	0	1	-1.001	0.0088	-0.004	0.0012	0.0011	-0.004	0.0007	-0.001	-0.003	0.0017	0.0018	-0.001	Unvoiced
591.49	22	1	-1.503	0.4042	0.1122	0.0299	0.0151	0.001	0.0104	0.004	-0.026	-0.061	-0.036	0.0542	Unvoiced
695.14	39	1	-1.746	0.6299	0.1414	0.0372	0.0043	0.0058	0.0129	-0.041	-0.057	-0.016	-0.018	0.0531	Unvoiced
659.17	27	1	-1.827	0.5817	0.2447	0.0962	0.0359	0.0177	-0.051	-0.144	-0.083	0.0564	0.1465	-0.07	Voiced
632.8	20	1	-1.584	0.3094	0.1629	0.1069	0.0763	0.0498	0.0041	-0.063	-0.08	-0.04	0.0227	0.0397	Voiced
617.49	4	1	-1.396	0.1681	0.09	0.1001	0.0497	0.0561	0.0531	-0.042	-0.062	-0.044	-0.045	0.0737	Voiced
578.03	0	1	-1.391	0.138	0.1236	0.109	0.0809	0.045	0.0007	-0.032	-0.055	-0.076	-0.066	0.124	Voiced
584.79	0	1	-1.336	0.1054	0.1083	0.1009	0.0644	0.0349	-8E-04	-0.024	-0.044	-0.066	-0.049	0.1065	Voiced
608.76	16	1	-1.415	0.1704	0.1229	0.1001	0.0809	0.0425	0.0008	-0.038	-0.055	-0.03	-0.005	0.0326	Voiced
651.17	14	1	-1.462	0.2096	0.1188	0.0989	0.0819	0.0307	-0.006	-0.022	-0.018	-0.024	-0.024	0.0255	Voiced
685.28	25	1	-2.037	1.1379	-0.26	0.2025	0.2025	-0.198	-0.145	0.1025	-0.005	-0.097	0.1963	-0.096	Voiced
711.02	19	1	-1.736	0.4991	0.1748	0.1034	0.0518	0.0076	-0.048	-0.071	-0.033	0.0067	0.0622	-0.012	Voiced
603.71	0	1	-1.014	0.0007	0.001	-4E-04	0.0051	0.0035	-0.002	-0.002	0.0002	0.0006	-0.001	0.0082	Voiced
579.57	0	1	-1.032	0.0103	0.0162	0.0112	0.0047	0.0009	-0.002	-0.003	-0.009	-0.007	-0.006	0.0174	Voiced
833.6	40	1	-1.759	0.5929	0.1572	0.0898	-0.005	-0.037	-0.009	-0.026	-0.018	-0.003	-0.01	0.0406	Voiced
790.32	38	1	-1.633	0.4237	0.1538	0.0843	0.0368	-0.016	-0.019	-0.009	-0.017	-0.021	-0.003	0.0325	Voiced
728.16	32	1	-1.482	0.2411	0.1193	0.1369	0.066	-0.034	-0.017	-0.021	-0.012	-0.001	0.0407	-0.024	Voiced

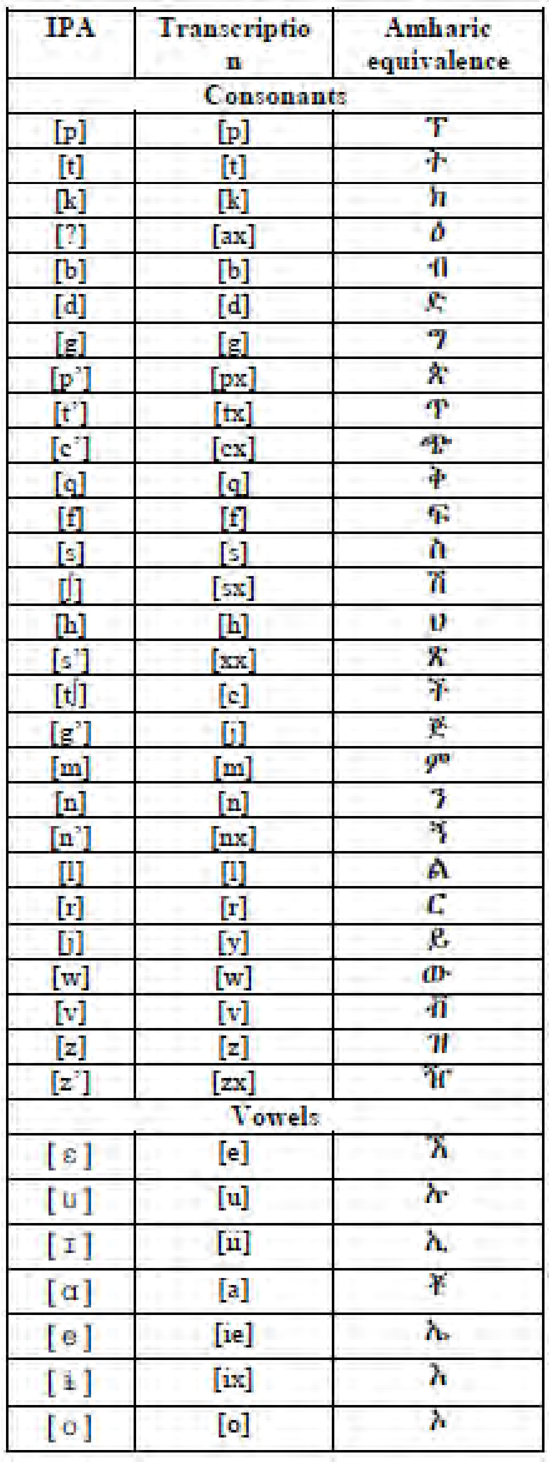
768.3	34	1	-1.588	0.3517	0.1613	0.0865	0.0353	0.0157	0.0045	-0.05	-0.038	0.019	0.0175	-0.006	Voiced
702.53	26	1	-2.263	1.6311	-0.458	0.1549	-0.067	0.018	0.0984	-0.127	-0.077	0.1139	-0.045	0.023	Voiced
581.82	4	1	-1.626	0.4275	0.1717	0.0767	0.0449	0.0092	-0.019	-0.063	-0.061	-0.045	-0.025	0.1102	Voiced
658.45	45	1	-2.014	1.0014	0.0674	0.0214	-0.042	0.0002	0.0348	-0.028	-0.07	-0.051	0.0151	0.0694	Voiced
644.79	31	1	-1.788	0.5469	0.1831	0.1376	0.0488	-0.024	-0.047	-0.078	-0.025	0.0057	0.006	0.0373	Voiced
552.42	0	1	-1.204	0.0535	0.0512	0.0541	0.0402	0.0292	0.0143	-0.002	-0.009	-0.028	-0.024	0.0255	Voiced
545.35	0	1	-1.224	0.0642	0.0663	0.0587	0.0448	0.0343	0.0115	-0.007	-0.016	-0.03	-0.031	0.0283	Voiced
539.02	0	1	-1.041	0.0138	0.0084	0.0122	0.0093	0.0054	0.0054	-5E-04	-0.003	-0.004	-0.004	-0.002	Voiced
535.66	0	1	-1.068	0.0167	0.0156	0.016	0.0116	0.0068	0.0046	0.0071	0.0071	0.0011	0.003	-0.021	Voiced
707.54	36	1	-1.666	0.4539	0.1534	0.0752	0.0376	0.0085	-0.01	-0.032	-0.023	-0.016	-0.023	0.051	Voiced
724.28	32	1	-1.805	0.6685	0.1002	0.0814	0.0298	-0.021	-0.032	-0.031	-0.003	0.0095	0.0244	-0.013	Voiced
645.82	18	1	-1.723	0.5179	0.1309	0.0599	0.0912	0.0338	-0.019	-0.059	-0.076	-0.027	0.0915	-0.016	Voiced
590.54	4	1	-1.46	0.2076	0.1554	0.1045	0.0627	0.0452	0.0118	-0.049	-0.093	-0.087	-0.026	0.1297	Voiced
561.67	0	1	-1.165	0.0368	0.0372	0.0398	0.0318	0.0182	0.0186	0.0048	-0.011	-0.02	-0.035	0.0453	Voiced
554.73	0	1	-1.003	0.0026	-0.002	0.0044	-6E-04	-0.002	-4E-05	-0.002	0.0024	3E-05	0.0005	-7E-04	Unvoiced
546.28	0	1	-1.212	0.1996	0.0142	0.0053	0.0208	-2E-04	-0.004	-0.01	-0.021	-0.036	0.0088	0.0353	Unvoiced
553.38	0	1	-1.075	0.0184	0.0157	0.0188	0.0113	0.0102	0.0038	-0.002	0.0003	-0.008	-0.013	0.0211	Voiced
580.9	10	1	-1.648	0.3436	0.23	0.1466	0.0601	-0.019	-0.061	-0.07	-0.052	-0.009	0.0616	0.0216	Voiced
698.74	34	1	-2.075	1.0881	0.02	0.0923	-0.02	-0.092	-0.032	0.0171	-0.037	-0.03	0.1404	-0.067	Voiced
593.41	4	1	-1.38	0.1715	0.105	0.076	0.0516	0.0248	0.0049	-0.03	-0.02	0.0038	0.005	-0.012	Unvoiced
580.04	0	1	-0.999	0.0018	-0.002	0.0023	-7E-04	-2E-04	-0.003	-0.002	0.0032	-0.004	-1E-03	0.004	Unvoiced
587.57	2	1	-1.448	0.3945	-0.022	-0.053	0.0478	0.0891	0.0497	-0.017	-0.034	-0.023	0.0334	-0.015	Unvoiced
672.6	20	1	-1.443	0.2154	0.0961	0.0763	0.0477	0.0466	0.0149	-0.015	-0.026	-0.006	0.06	-0.057	Voiced
746.75	26	1	-1.517	0.2913	0.1286	0.0931	0.0433	0.0092	-2E-04	-0.031	-0.016	0.0262	0.0004	-0.016	Voiced
639.29	2	1	-1.315	0.1333	0.0669	0.0461	0.0548	0.0376	0.008	0.0047	-0.005	-0.02	-0.007	-8E-05	Voiced
580.72	0	1	-1.04	0.035	0.0261	0.0047	-0.007	-0.022	-0.019	0.0053	0.0059	0.0104	0.0164	-0.016	Unvoiced
571.03	0	1	-1.001	0.0023	-0.002	0.0036	0.0017	-0.002	-0.002	-8E-04	0.0024	-0.003	-1E-04	0.0016	Unvoiced
560.16	0	1	-1.015	0.0086	0.0039	0.0063	-0.005	-2E-04	0.0012	0.0033	0.0033	-0.011	-0.001	0.0057	Unvoiced
587.16	0	1	-1.184	0.1775	0.0162	-0.017	0.0024	-0.014	0.0112	0.0391	0.0258	-0.008	-0.037	-0.01	Unvoiced
658.72	33	1	-1.591	0.3603	0.1713	0.1006	0.0454	0.0042	-0.014	-0.044	-0.062	-0.039	-0.007	0.0821	Voiced

Appendix C: Matlab Codes

```
% opens a dialogue to choose a sound file
file = uigetfile({'*.wav;*.mp3;*.wmv;*.wma;*.lab','Audio files'},'Select a
sound file','MultiSelect','On');
wfile=file{2};
lfile=file{1};
[wavFile,Fs,nbits] = wavread(wfile);
ext='.xlsx';
fn=sprintf('%s''%s',wfile,ext);
      frameSize = 960; % frame size 20ms for 48000Hz sampling rate
      overlap = 0; % frame overlap
      frameMat=buffer(wavFile, frameSize, overlap); % process frames
      frameNum=size(frameMat, 2); % calculate size
% Function call to find the energy
En = calcEnergy(frameMat,frameNum);
% Function call to find the zero-crossing rate
Zcr = calcZcr(frameMat,frameNum);
% Function call to do LPC
lpc=dolpc(frameMat,12);


% Functions
///////////////////////// Calculate Energy /////////////////////////////
function En = calcEnergy(frameMat,frameNum)
% Compute energy
En=zeros(frameNum, 1);
for i=1:frameNum
    frame=frameMat(:,i);
    En(i)=sum(abs(frame));
end
////////////////////////////////////////////////////////////////////////
/////////////////// Calculate Zero-Crossing Rate////////////////////////
function Zcr = calcZcr(frameMat,~)
% Compute zero-crossing rate
for i=1:size(frameMat,2)
    crossing=0;
    frames=frameMat(:,i);
    for j=1:length(frames)-1 %length : get frame size
        if ((frames(j)>=0 && frames(j+1)<0) || (frames(j)<=0 && frames(j+1)>0))
            crossing=crossing + 1;
        end
    end
    Zcr(i)=crossing;
end
///////////////////////////////////////////////////////////////////////
////////////////////////// Calculate LPC //////////////////////////////
function y = dolpc(x,modelorder)
% modelorder is order of model, defaults to 8
if nargin < 2
  modelorder = 8;
end
% Find LPC coeffs
y=lpc(x,modelorder);
//////////////////////////////////////////////////////////////////////
```

Appendix D: Amharic phonetic list, IPA Equivalence and its ASCII Translation table (adopted from [13])



IPA	Transcription	Amharic equivalence
	Consonants	
[p]	[p]	ፕ
[t]	[t]	ት
[k]	[k]	ክ
[ʔ]	[ax]	ዕ
[b]	[b]	ብ
[d]	[d]	ድ
[g]	[g]	ግ
[p']	[px]	ጵ
[t']	[tx]	ጥ
[c']	[cx]	ጭ
[q]	[q]	ቅ
[f]	[f]	ፍ
[s]	[s]	ስ
[ʃ]	[sx]	ሽ
[h]	[h]	ህ
[s']	[xx]	ጽ
[tʃ]	[c]	ች
[g']	[j]	ጅ
[m]	[m]	ም
[n]	[n]	ን
[n']	[nx]	ኝ
[l]	[l]	ል
[r]	[r]	ር
[j]	[y]	ይ
[w]	[w]	ው
[v]	[v]	ቭ
[z]	[z]	ዝ
[z']	[zx]	ዥ
	Vowels	
[ɛ]	[e]	ኧ
[u]	[u]	ኡ
[ɪ]	[ii]	ኢ
[ɑ]	[a]	ኣ
[e]	[ie]	ኤ
[ɨ]	[ix]	እ
[o]	[o]	ኦ

Declaration

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all source of materials used for the thesis have been duly acknowledged.

Declared by:

Name: **Mohammed Abebe Yimer**

Signature: ______________________

Date: __________________________

Confirmed by advisor:

Name: **Sebsbie Hailemariam (PhD)**

Signature: ____________________

Date: ________________________

Place and date of submission: Addis Ababa, February, 2012.