Addis Ababa University
School of Graduate Studies

SELECTING APPROPRIATE AMHARIC UNIT FOR DOMAIN SPECIFIC SPEECH SYNTHESIS: A CASE FOR MOBILE PHONES

By Workagegnehu Petros Fitamo

A thesis submitted to the School of Graduate Studies of Addis Ababa University in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science

October, 2008
Addis Ababa

Addis Ababa University
School of Graduate Studies
Faculty of Informatics
Department of Computer Science

SELECTING APPROPRIATE AMHARIC UNIT FOR DOMAIN SPECIFIC SPEECH SYNTHESIS: A CASE FOR MOBILE PHONES

By Workagegnehu Petros Fitamo

Approved By

Examining Board

1. Sebsibe Hailemariam, Advisor _____
2. _____ _____
3. _____ _____
4. _____ _____
5. _____ _____

Dedication

To my parents

Acknowledgements

Had it not been the help of the following people, the materialization of this paper with such a shape would have been unthinkable.

My first and foremost gratitude goes to my advisor Sebsibe Hailemariam for his guidance and unreserved comments in the whole lot of this thesis work. My heart-felt gratitude also goes to my family for their love, material and moral support in my course and research endeavors.

I am also indebted to the following people: Surafel Gelgelo for his support and moral encouragement; Fitsum Fikru, Abeselom Zemedkun and Girma Bulti for providing me their laptops; Tessema Mindaye, Nesredine Suleyman, Abera Abebaw, Demeke Asres and Melisew Dejene for commenting on the drafts of this paper and for giving me substantial ideas.

All the respondents who participated in filling out the questionnaire also deserve gratitude.

Above all, I praise my God for giving me health and all the courage to finalize this work.

Table of Contents

List of Tables

List of Figures

List of Appendices

List of Acronyms

/x/	Phoneme x
ASSMP	Amharic Speech Synthesizer for Mobile Phone
C	Consonant
CDC	Connected Device Configuration
CLDC	Connected, Limited Device Configuration
DRT	Diagnostic Rhyme Test
DSP	Digital Signal Processing
IPA	International Phonetic Alphabet
J2ME	Java™ 2 Platform, Micro Edition
J2SE	Java 2 Platform, Standard Edition
JCP	Java Community Process
JVM	Java Virtual Machine
MIDP	Mobile Information Device Profile
MOS	Mean Opinion Score
MRT	Modified Rhyme Test
NLP	Natural Language Processing
PDA	Personal Digital Assistant
PDAP	PDA Profile
RIFF	Resource Interchange File Format
SAMPA	Speech Assessment Methods Phonetic Alphabet
TTS	Text-to-Speech
V	Vowel

Abstract

Speech synthesis – the production of artificial speech – has a lot of applications. Applying speech synthesis onto mobile phones for Amharic language will be an important success in language technology.

Mobile phones are characterized by smaller memory and processing capacity. The choice of a unit for concatenation has an impact on the quality of the synthetic speech produced, the size of the database that is used to store the speech units, and also the time required to synthesize a speech. In this thesis, three Amharic units: phonemes, diphones, and syllables are compared.

Analysis is done on these units in terms of naturalness, intelligibility, memory requirement, and processing time. The result shows that diphone based speech synthesis approach is the appropriate alternative since it requires less memory and time, and provides reasonably acceptable naturalness and intelligibility. The overall Mean Opinion Score obtained for intelligibility and naturalness is 4.10 and 3.69, respectively.

Keywords: Speech Synthesis, Mobile Phones, Diphones, Amharic Language

CHAPTER ONE

INTRODUCTION

1.1. Background of the Study

Speech is the primary means of communication among people. Speech synthesis, also called text-to-speech (TTS) synthesis, is the artificial production of human speech and it has been under development for several decades. Recent progress in speech synthesis has produced speech synthesizers with very high intelligibility[1], but, the sound quality and naturalness[2] still remain a major problem. However, the quality of present products has reached an adequate level for several applications, such as multimedia and telecommunications [1, 2].

The text-to-speech synthesis procedure consists of two main phases: text analysis and generation of waveforms. The first one transcribes the input text into a phonetic and some other linguistic representations and the second one produces the acoustic output for the phonetic and prosodic information [1].

According to the model used for speech generation, speech synthesis systems are classified into three common types: *articulatory synthesis*, *formant synthesis*, and *concatenative synthesis*. Articulatory synthesis uses a physical model of human speech production system that includes all the articulators. Formant synthesis uses a source-filter model, where the filter is characterized by slowly varying formant frequencies. Concatenative synthesis generates speech by joining speech segments together [2].

[1] Intelligibility is a measure of how understandable speech is to humans [39].
[2] Naturalness is a measure of how "human" a synthesizer sounds [39].

Concatenative synthesis seems to have won the race for natural sounding artificial speech among the text-to-speech synthesis techniques. Diphones were the most widely used acoustic units till a few years ago, basing their success on their high combinatorial power: a relatively small number of diphones allows the generation of any message in a given language. The drawbacks were the number of junctions and the need to heavily modify the prosodic parameters, resulting in a "robotic" voice. Traditional synthesizers were limited by computational difficulties, largely overcome by nowadays computers. Moreover, the availability of robust automatic speech analysis and labeling tools extends the concept of concatenative synthesis towards what is called corpus-based or unit selection synthesis. Unit selection obtains highly natural-sounding speech by concatenating units longer than diphones and available in many prosodic variants, in order to reduce the number of junctions and the need of prosodic modification. The key factors in this approach are the phonetic and prosodic coverage, the intended domain, and the run-time selection criteria for the acoustic units [4].

The unit selection paradigm is a cluster based technique where units of the same type are clustered based on the acoustic differences. The units can be syllables, phonemes, diphones, and half phones. The clusters are then indexed based on higher level phonetic and prosodic context. During synthesis an appropriate unit is chosen from multiple instances of that unit based on minimization of joining cost and concatenation cost [3, 11]. General unit selection system, when units that get concatenated match prosodic and/or spectrally, offers speech near human quality, but when the units do not match prosodic and/or spectrally, it is usually much worse than a diphone synthesizer. It is hoped that in a limited domain the good quality of unit selection can be obtained while the bad quality is avoided [12].

Limited domain speech synthesis is mean that applications whose vocabulary to be synthesized is very limited. Some common examples are systems telling the time, reading telephone numbers, and fixed weather reports. The reason for popularity of such systems is that they are easy to build small version of unit selection synthesis since there are controlled number of units. Because of this control they get the good quality of unit selection and avoid the pitfalls of more open domain, general unit selection systems [5].

In the developing world, where the vast majority of the world's population lives, mobile phone usage is growing at a phenomenal rate. Many users would value timely information on jobs, health issues, local market prices, etc. but, they have little access to computers and use only their local or national language as a means of information exchange. Voice services and speech technology would address this need but have to be adapted both linguistically and culturally with different service models in order to succeed for these new users, who represent a completely different and new market for voice systems [13].

Mobile phones have got small memory, limited processing capacity, poor display and unhandy input when compared with the common computer. These restrictions pose a lot of challenges to mobile phone application program, which involves several fields such as application, service, system security, etc. But, a huge progress in telecommunications and in mobile phones during the past years enabled mobile phones to launch various multimedia applications [14, 15]. For example, the paper, [15], shows an application for reading short messages, which is a combination of speech synthesis and face animation, for Slovak language. This will be of particular importance if it is adapted to local languages in Ethiopia, for example, Amharic.

Amharic is the official language of Ethiopia. Among the 73 languages which are registered in the country, Amharic is the widely spoken language and it is one of the Semitic languages having its own script. It is estimated to be mother tongue of more than 17 million people, with at least an additional 5 million of second language speakers [32]. The scripts are more or less orthographic representation of the phonemes in the language. The script of Amharic language is phonetic in nature. It has 32 consonants and 7 vowels. The orthographic representation of the language is organized into orders. Each of the 32 consonants has seven orders (derivatives). Six of them are consonant-vowel combinations while the seventh is constant. Moreover, there are extra orthographic symbols in the language. The total number of orthographic symbols of the language exceeds 230 [3].

1.2. Application of the Study

Synthetic speech is used in several applications. The application field of synthetic speech is expanding fast whilst the quality of TTS systems is also increasing steadily. Some of the applications include [1]:

- **Application for the blind**: Probably the most important and useful application field in speech synthesis is the reading and communication aids for the blind. Before synthesized speech, specific audio books were used where the content of the book was read into audio tape.

- **Application for the deafened and vocally handicapped**: Synthesized speech gives the deafened and vocally handicapped an opportunity to communicate with people who do not understand sign language.

- **Educational applications**: Synthesized speech can be used also in many educational environments. A computer with speech synthesizer can teach 24 hours a day and 365 days a year. It can be programmed for special task like spelling and pronunciation teaching for different languages.

- **Applications for telecommunications and multimedia**: The newly emerging applications in speech synthesis are in the area of multimedia. With synthetic speech, e-mail messages may be listened to via telephone line. Synthesized speech may also be used to speak out short text messages.

- Mobile phones with Amharic speech synthesizer can be used by people who do not know how to read/write Amharic.

1.3. Motivation

Nowadays, a great deal of advancement in technology, especially in the area of mobile phones is being noticed. Mobile phones have become important components in the day-to-day activities of human beings. The primary purpose of mobile phones is making communication possible among people. In addition, their being programmable enhances their functionalities. This made us think of loading speech applications onto mobile phones in order to serve the local community (in Ethiopia). To this end, first the mobile phones should support Amharic keypad feature. This problem is alleviated because Nokia has produced mobile phones that support Amharic keypad feature [10]. Second, Amharic speech synthesizer that is particularly suited to mobile phones should be developed.

As it is stated in [47], mobile communication is perhaps the single most transformative technology for rural African villages to improve access to health care and education, create new business opportunities and access to markets, and ultimately to help eradicate extreme poverty. The tremendous opportunities which mobile phones make possible in every kind of community and economic activity - ranging from pastoralists and farmers, to traders, health workers and teachers is exciting.

Hence, the motivation arose to do a research that explores various Amharic speech units and selects one to develop Amharic speech synthesizer for mobile phones. The application will have an impact to enable most Ethiopians to use mobile phones.

1.4. Research Problem

Mobile phones are becoming one of the important components in the day-to-day activities of human beings. Nowadays, the number of Ethiopians using mobile phones is growing. According to [46], the number of mobile users in Ethiopia grew by 77.5% in 2005-6. The introduction of Amharic keypad feature helps people in Ethiopia to use mobile phones using Amharic. However, to address the needs of people, for example, who are visually impaired and who do not know how to read/write Amharic, a need arises to develop Amharic speech synthesizer for mobile phones. In developing the synthesizer, issues such as selection of speech synthesis techniques and choice of a unit in the case of concatenative speech synthesis technique should be addressed keeping in mind the limited processing and memory capacity of mobile phones.

1.5. Objective

1.5.1. General Objective

The main objective of this research is to select appropriate Amharic unit for speech synthesizer to be applied on mobile phones by addressing issues like small memory and limited processing capacity of mobile phones, naturalness, and intelligibility of the synthesized speech.

1.5.2. Specific Objectives

The specific objectives of the research include:

- Collecting data from sources like telephone directories.
- Performing analysis on the collected data.
- Selecting appropriate Amharic unit for mobile application.
- Developing a model for a speech synthesizer for mobile phone.
- Developing a prototype for the speech synthesizer.
- Testing the developed system and analyze the performance of system.

1.6. Scope of the Study

Although text-to-speech synthesis deals with converting arbitrary text into utterance, in this thesis work, the text to be synthesized is composed of a maximum of one word. In addition, Amharic names and numerals (digits) are the text of interest – that is why the phrase “domain specific” is

added on the title of this thesis work. In addition, prosodic effects are not considered during synthesis.

1.7. Methodology

The following methodologies have been used to achieve the general and specific objectives of this thesis work.

1.7.1. Development Methodology

Mobile phone applications are currently being developed using Java technology called Java™ 2 Platform, Micro Edition (J2ME). J2ME has also been used to implement speech synthesizer for mobile phones in this thesis work. Modules, which are used for data analysis, are implemented using Java 2 Platform, Standard Edition (J2SE).

Additional tools have also been used to supplement the development process. Some of these include:

- Microsoft® Sound Recorder software Version 5.1 to collect speech data.
- WaveSurfer to extract diphones from the speech data that has been collected.
- A standalone PC with Intel® Pentium® IV CPU with 1.80 GHz speed, 256 MB of RAM, 40 GB of hard disk capacity, with Microsoft Windows XP Professional Version 2002 Service Pack 2 operating system.
- Microphone for the purpose of recording sounds.

- The Sun Java™ Wireless Toolkit for CLDC 2.5.2 to create application for mobile phone by providing an emulator.

1.7.2. Data Collection Methodology

The main objective of data collection methodology is to prepare appropriate data set to train and test the developed speech synthesis system. This data set is collected from Amharic NLP website, [36], and Ethiopian telephone directory of year 2006-7. Greedy based optimal data selection has been done on the data set to select the data that is used for training, which covers statistically representative data set and all the units required. Testing data has been taken from the data set that is not used for training.

1.7.3. Testing Methodology

Evaluation of the developed system has been done with respect to naturalness and intelligibility of the synthetic speech output by the system. A technique known as Mean Opinion Score (MOS) has been used in the evaluation process to measure naturalness and intelligibility.

1.8. Organization of the Thesis

The rest of this thesis document is organized as follows. Chapter two deals with literature review. It also deals with related works that have been done to produce synthetic speech for mobile phones. Chapter three describes Amharic writing system and Amharic phonetics that are closely related topics to speech synthesis. Chapter four documents the detailed design of Amharic speech synthesizer for mobile phones. Chapter five describes the implementation of speech synthesizer

for mobile phones and also how experimentation is conducted. Finally, chapter six is devoted to conclusion and recommendation.

CHAPTER TWO

LITERATURE REVIEW

2.1. Introduction

This chapter gives us background about this thesis work (especially about speech synthesis). It also includes review of related works (which includes speech synthesis for mobile phones for languages other than Amharic).

As it is already mentioned in the first chapter, speech synthesis, also called text-to-speech synthesis, is the artificial production of human speech from text. The process of converting written text into speech passes through a number of steps, which are broadly classified into two basic modules: the Natural Language Processing (NLP) module and the Digital Signal Processing (DSP) module. The NLP module is responsible for producing a phonetic transcription of the text together with the desired intonation and rhythm. The DSP module transforms the symbolic information it receives from the NLP module into speech [17].

Prior to taking a brief look at the text-to-speech synthesis procedure, it is necessary to have an insight about how the human speech production system works. This helps us understand the whole text-to-speech synthesis procedure well. Section 2.2 describes the human speech production system briefly. Sections 2.3 and 2.4 describe the Natural Language Processing module and the Digital Signal Processing module, respectively. Finally, review of related works and summary are presented.

2.2. Human Speech Production System

The human speech production system, which is illustrated in Figure 2.1, is used to produce human speech. Although Figure 2.1 shows a schematic view of only the major articulators, the gross components of the speech production system are the lungs, trachea, larynx (organ of voice production), pharyngeal cavity (throat), oral and nasal cavities. The pharyngeal and oral cavities are typically referred to as the vocal tract, and the nasal cavity as the nasal tract [1, 2, 16].

Speech is produced by air-pressure waves emanating from the mouth and the nostrils of a speaker. The source of air during speech is the lungs. Speech sounds are usually considered as either voiced or unvoiced, but in some cases they are something between the two. When the vocal folds are held close together and oscillate against one another during speech production, the sound is said to be voiced. When the vocal folds are too slack or tense to vibrate periodically, the sound is said to be unvoiced. The place where the vocal folds come together is called the glottis [1, 2].

Voiced sounds consist of fundamental frequency (F0) and its harmonic components produced by the vocal folds. The vocal tract modifies this excitation signal causing formant frequencies. With purely unvoiced sounds, there is no fundamental frequency in excitation and therefore there is no harmonic structure either. The airflow is forced through a vocal tract constriction that can occur several places between glottis and mouth. Some sounds are produced with complete stoppage of airflow followed by a sudden release [1].

The oral cavity is one of the most important parts of the vocal tract. Its size, shape and acoustics can be varied by the movements of the palate, the tongue, the lips, the cheek and the teeth. Especially, the tongue is very flexible, the tip and the edges can be moved independently and the

entire tongue can move forward, backward, up and down. It is shaped away from the palate for the vowels, placed close to or on the palate and other hard surfaces for consonant articulation. The palate is a long relatively hard surface at the roof inside the mouth, which, when the tongue is placed against it, enables consonant articulation. The lips control the size and the shape of mouth through which the speech sound is radiated. The lips can be rounded or spread to affect vowel quality, and closed completely to stop the oral airflow in certain consonants, e.g., /p/, /b/, and /m/. The teeth, another place of articulation, are used to brace the tongue for certain consonants [1, 2].

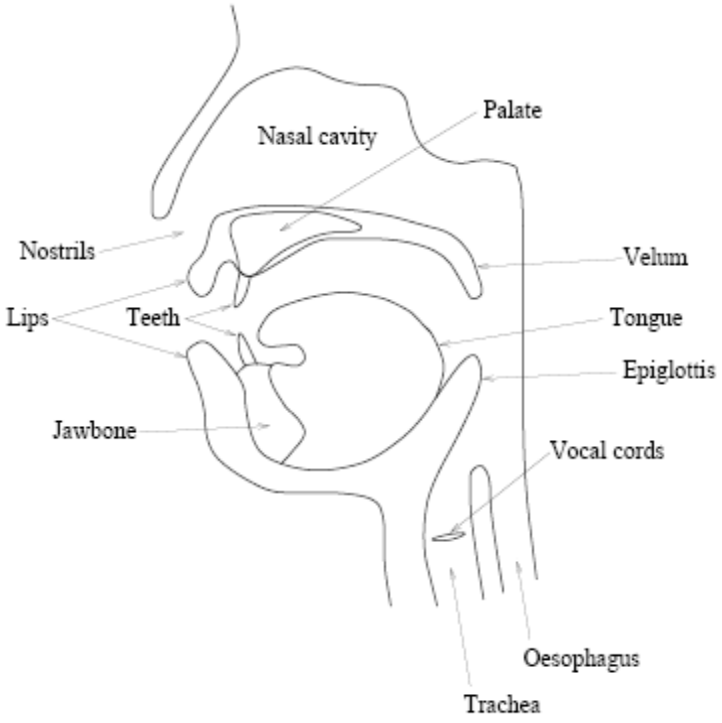

Figure 2.1: The Human Speech Production System [16]

Unlike the oral cavity, the nasal cavity has fixed dimension and shape. The air stream to the nasal cavity is controlled by the velum, which operates as a valve, opening to allow the passage of air through the nasal cavity. Sounds produced with the flap open include /m/ and /n/ [1, 2].

2.3. The Natural Language Processing Module

The NLP module consists of three basic processing stages: *text analysis*, *phonetic analysis*, and *prosody generation* [2, 17, 18]. A brief description of each of these processing stages is given in the following subsections.

2.3.1. Text Analysis

Text analysis component is typically responsible for determining document structure, conversion of non-orthographic symbols, and parsing of language structure and meaning [2].

Document structure is important to provide context for all later processes. In addition, some elements of document structure, such as sentence breaking and paragraph, may have direct implications for prosody [2].

Text normalization – the conversion from the variety of symbols, numbers, and other non-orthographic entities into a common orthographic transcription suitable for subsequent phonetic conversion – is also done by the text analysis. Text normalization is usually a very complex task and includes language dependent problems. For example, in English, numeral 243 would be expanded as two hundred and forty three and 1750 as seventeen fifty (if year) or one thousand seven hundred and fifty (if measure). Related issues also exist for dates and fractions, expansion of ordinal numbers, and abbreviations. For instance, abbreviations may be expanded into full words, pronounced as written or pronounced letter by letter [1, 2, 16, 18, 19].

Text analysis also recovers the syntactic constituency and semantic features of words, phrases, clauses, and sentences, which are important for both pronunciation and prosodic choices in the successive processes [2].

2.3.2. Phonetic Analysis

The task of phonetic analysis component is to convert lexical orthographic symbols to phonemic representation along with possible diacritic information such as stress placement. Phonetic analysis is thus referred to as grapheme-to-phoneme conversion. Grapheme-to-phoneme conversion is trivial for languages where there is a simple relationship between orthography and phonology. Such a simple relationship can be well defined by a set of rules. Languages belonging to this category are referred to as phonetic languages [2, 18]. According to [33], Amharic is a phonetic language.

The following are three services necessary to produce accurate pronunciation [2]:

- **Homograph disambiguation** – Homographs are words that are spelled the same way but they differ in meaning and usually in pronunciation [1]. It is important to disambiguate words with different senses to determine proper phonetic pronunciation.

- **Morphological analysis** – Analyzing the component morphemes provides important cues to attain pronunciation for inflectional and derivational words.

- **Letter-to-sound conversion** – The last stage of phonetic analysis includes letter-to-sound rules (or modules) and a dictionary lookup to produce accurate pronunciation for any arbitrary word.

2.3.3. Prosody Generation

The last component of the NLP module, prosody generator, is where certain properties of speech signal such as intonation, stress, and duration called prosodic features are processed. Intonation

means how the pitch pattern or fundamental frequency changes during speech. The prosody of continuous speech depends on many separate aspects, such as the meaning of the sentence, the speaker characteristics, and emotions. The prosodic dependencies are shown in Figure 2.2. Unfortunately, written text usually contains very little information of these features and some of them change dynamically during speech. However, with some specific control characters it is possible to give this information to a speech synthesizer [1, 17, 18].

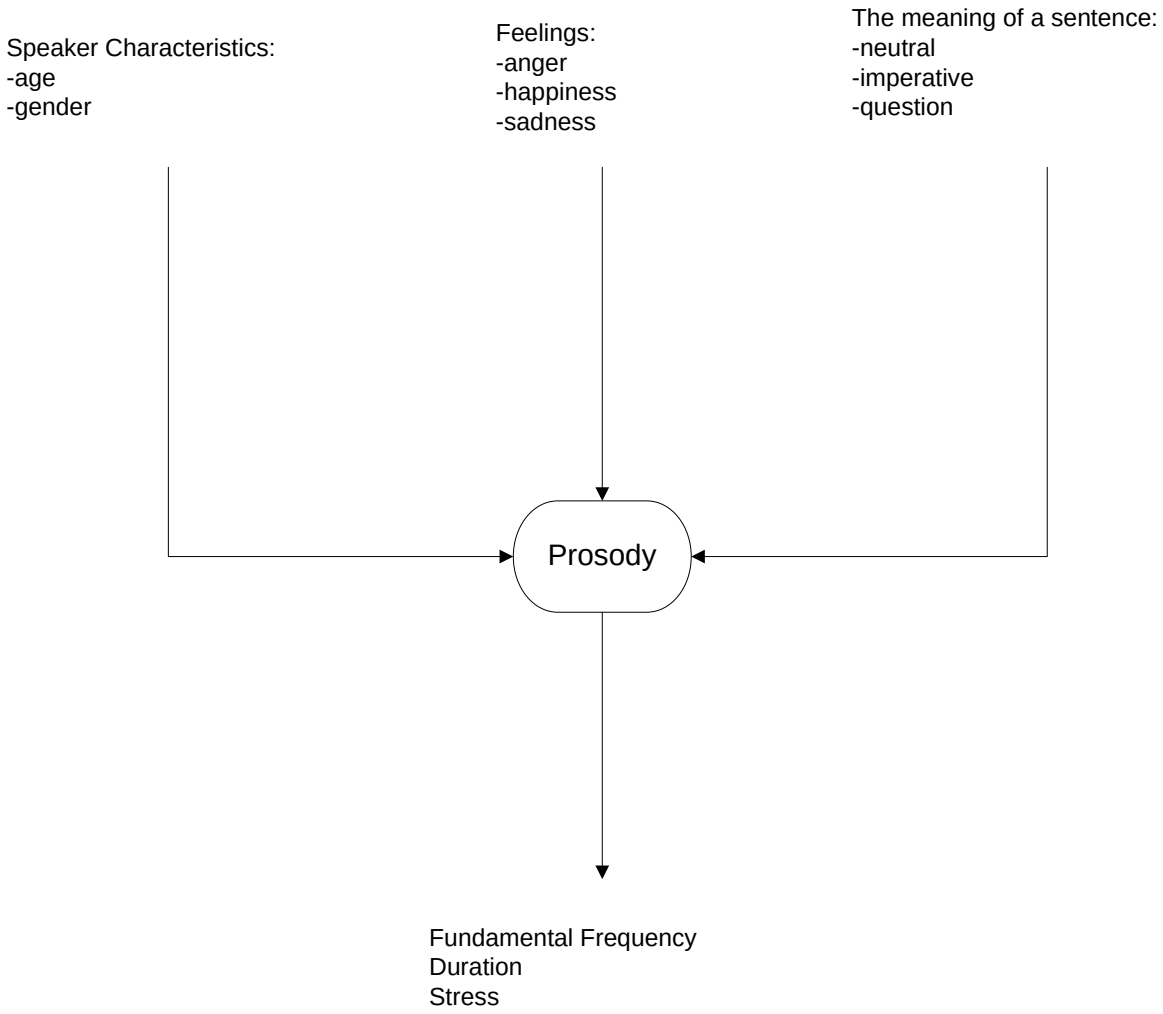

Figure 2.2: Prosodic dependencies

2.4. The Digital Signal Processing Module

In DSP module, transformation of information received from the NLP module into speech is done [17]. As stated in the first chapter, there are three different categories of waveform generation or so called types of speech synthesis: articulatory synthesis, formant synthesis, and concatenative synthesis. Speech synthesis systems can also be classified into *synthesis by rule* and *data driven synthesis* based on the degree of manual intervention in the design of the system. In the former, a set of manually derived rules is used to drive a synthesizer, and in the latter the synthesizer's parameters are obtained automatically from real speech data. Concatenative synthesizers are, thus, data driven, while formant synthesizers use synthesis by rule [2]. Each of these synthesis techniques is described in the following subsections in brief.

2.4.1. Articulatory Synthesis

Articulatory synthesis tries to model the human articulators as perfectly as possible, using parameters that model the mechanical motions of the articulators and the distributions of volume, velocity and sound pressure in the lungs, larynx, and vocal and nasal tracts, so as to be potentially satisfying method to produce high-quality synthetic speech. On the other hand, it is also one of the most difficult methods to implement and the computational load is also considerably higher than the other methods. Thus, it has received less attention than the other synthesis methods and has not yet achieved the same level of success as the other methods have achieved [1, 17, 18].

The DAVO (Dynamic Analog of the VOca1 tract), is the first articulatory synthesizer for English language that was introduced in 1958 by George Rosen at the Massachusetts Institute of technology (M.I.T). It was controlled by tape recording of control signals created by hand [1].

2.4.2. Formant Synthesis

Formant synthesis employs some set of rules to synthesize speech using the formants that are the resonance frequencies of the vocal tract. Since the formants constitute the main frequencies that make sounds distinct, speech is synthesized using these estimated frequencies [17, 18].

Formant synthesis provides infinite number of sounds which makes it more flexible than concatenation methods. In general, formant synthesis produces intelligible but not natural sounding speech [1, 21].

There are examples of formant speech synthesizers developed for English language. Some of these include: Parametric Artificial Talker (PAT) was the first formant synthesizer introduced by Walter Lawrence in 1953, Orator Verbis Electris (OVE I) was another formant synthesizer introduced by Gunnar Fant at about the same time as PAT, and OVE II which is an improved version of OVE I was introduced by Fant and Martony ten years after the introduction of OVE I [1].

There are also formant synthesizers developed for languages other than English. SYNTE3 and AVS are some to list. SYNTE3 is a formant synthesizer developed for Finnish language introduced by Laine [1]. AVS (Amharic Vowel Synthesizer) is a formant synthesizer for Amharic vowels developed by Nadew Tademe [9].

2.4.3. Concatenative Synthesis

Synthesis-by-rule techniques produce intelligible speech, but the speech sounds unnatural because it is very difficult to capture all the nuances of natural speech in a small set of manually derived rules [21]. In concatenative synthesis, an utterance is synthesized by joining together

several speech fragments taken from natural speech. The beauty of this synthesis technique is that unlike synthesis-by-rule, it requires neither rules nor manual tuning. Since each speech segment is natural, the synthesized speech is expected to be natural. However, this approach is greatly affected by coarticulation[1], so if we concatenate speech segments that were not adjacent to each other, there happens discontinuities in the concatenated speech [2].

The following are issues that need to be addressed in designing a concatenative speech synthesis system [2]:

- What type of speech segments to use? Diphones, phonemes, syllables, phrases, sentences, etc. can be used.
- How to design the acoustic inventory, or a set of speech segments, from a set of recordings? This includes extracting the speech segments from the set of recordings as well as deciding how many are necessary.
- How to select the best string of speech segments from a given library of segments, given a phonetic string and its prosody? There may be several strings of speech segments that produce the same phonetic string and prosody.
- How to alter the prosody of a speech segment to best match the desired output prosody?

In the following subsection various speech units, their numbers in a particular language, and the quality of the synthetic speech produced when using the speech units are presented in brief.

[1] Coarticulation – indicates the extent to which two neighboring phonemes have a joint articulation [21].

2.4.3.1. Choice of Unit

As mentioned in the first chapter, the speech units that are used in concatenative speech synthesis can be phonemes, diphones, syllables, words, demisyllables[1], and sometimes even triphones. There are issues that need to be considered in choosing/selecting units for concatenation [1, 2].

As stated in [1], one of the most important aspects in concatenative speech synthesis is to find correct unit length. The selection is usually a trade-off between longer and shorter units. With longer units, high naturalness, less concatenation points, and good control of coarticulation are achieved, but the amount of required units and memory is increased. With shorter units, less memory is needed, but the sample collecting and labeling procedures become more difficult and complex.

According to [2], the issues that should be taken into account when choosing/selecting appropriate units for synthesis include the following:

- The unit should lead to *low concatenation distortion*. A simple way of minimizing this distortion is to have fewer concatenations and thus use long units such as words, phrases, or even sentences. Instead of using long units to minimize distortion, using several instances per unit (as it is done in unit selection synthesis) can be an alternative which allows the choice of units that lead to low concatenation.

- The unit should lead to *low prosodic distortion*. While it is not crucial to have units with slightly different prosody than the desired target, replacing a unit with a rising pitch with

[1] Demisyllables represent the initial and final parts of syllables [1].

another with a falling pitch may result in an unnatural sentence. Altering the pitch or duration of a unit is possible at the expense of additional distortion.

- The unit should be *generalizable*, if unrestricted text-to-speech synthesis is required. If one chooses words or phrases as units, one can not synthesize speech from arbitrary text, because it is guaranteed that the text will contain words not in the inventory. The longer the speech segments are, the more of them one needs to synthesize speech from an arbitrary text.

- The unit should be *trainable*. The training data should be sufficient to estimate all our units. Since the training data is usually limited, having fewer units leads to better trainability in general.

An additional issue e.g., the issue of memory and processing power comes into being if small computing devices such as mobile phones are considered. A practical challenge is how to balance these selection criteria [2]. The following are comparisons between the various speech units including their strengths and weaknesses.

Phoneme

A phoneme is a member of the set of the smallest units of speech that serve to distinguish one utterance from another in a language or dialect. It is the most straightforward unit. Having one instance of each phoneme, independent of the neighboring phonetic context, is very generalizable, since it allows us to generate every word or sentence. It is also very trainable and we could have a system that is very compact. For a language with N phonemes, N = 42 for English, only N unit instances are needed. The problem is that using context-independent

phonemes results in many audible discontinuities making the system unintelligible. These audible discontinuities can be minimized by using context-dependent phonemes [2].

In context-dependent phonemes, if the context is limited to the immediate left and right phonemes, then the unit is known as *triphone* [2]. Triphones contain one phoneme between steady-state points (half phoneme - phoneme - half phoneme) [1]. In a language with N phonemes, there are N^3 triphones, but, not all combinations will occur in practice. As stated in [1], there are more than 10,000 triphones in English. The increased number of units enables us to capture more contextual variations. Due to the larger number of units used, discontinuities can be smaller than the case of diphones while making use of the best available data. Triphones are trainable and generalizable.

Diphone

Diphone, also called dyad, is a type of unit that has been extensively used. A diphone extends from the central point of the steady state part of one phoneme to the central point of the following one, so they contain transitions between adjacent phonemes. Hence, diphone is, on the average, one phoneme long. If a language has N phonemes, there are potentially N^2 diphones. However, in practice, many such diphones never occur in the language so that a smaller number is sufficient. For example, in English only about 1300 diphones are needed [1, 2].

Diphones were the first type of unit used in concatenative systems because they yield fairly good quality. This is because the concatenation point is taken from the most steady state region of the signal, which reduces the distortion from the concatenation points [1]. Diphones are trainable, generalizable, and offer better quality than context-independent phonemes [2].

Syllable

Syllable is a speech unit that consists of a movement from a constricted or silent state to a vowel-like state and then back to constricted or silent state [20]. The number of different syllables in each language is considerably smaller than the number of words, but the size of unit database is usually still too large for TTS systems [1]. As stated in [2], there are about 10,000 syllables in English, so even a context-independent syllable system needs to store at least as many if one instance per syllable is needed for full generalizability. There are also spectral discontinuities even if they are not too noticeable. And it has been observed that discontinuities across syllables stand out more than discontinuities within syllables.

Word and Phrase

The unit can also be as large as a word or even a phrase. As it is tried to mention earlier, using such longer units can increase naturalness significantly. However, generalizability and trainability remain poor and also it is difficult to have all the instances required to synthesize the output utterance. For example, there are hundreds of thousands of different words and proper names in each language, so word is not a suitable unit for any kind of unrestricted TTS system [1, 2].

There are some research works done locally in an attempt to develop speech synthesizers using concatenative synthesis approach for local languages such as Amharic, Tigrigna, and Oromiffa. Laine [6] developed Amharic speech synthesizer using diphone as basic unit. Henock [7] extended the work of Laine by using diphones and syllables as units. Habtamu [8] implemented Amharic speech synthesizer using diphone as a unit. Diphone based speech synthesizer for

Tigrigna is done by Tesfaye [24]. Morka [25] developed diphone based speech synthesizer for Oromiffa.

2.4.3.2. Unit Selection Synthesis

Unit selection synthesis is also concatenative speech synthesis in which there are a number of unit instances (stored in a database) that vary in terms of prosody and other characteristics for each linguistic unit (phoneme, diphone, syllable, etc.). During synthesis, an algorithm selects one unit from the possible choices in attempt to find the best overall sequence of units which matches the input specification [21].

It has been stated in [30, 37, 38, 40] that unit selection synthesis provides more natural sounding utterance than the one that uses only one instance of each unit in a database. It is also pointed out in the works that the naturalness is due to relying less on signal smoothing techniques that cause degradation in the quality of synthetic voice. That is, unit selection synthesis employs almost no signal modification techniques that have impact on the degradation of the quality of synthetic voice [40, 21].

As stated in [5, 15], the quality of voice produced using unit selection synthesis varies substantially. When good, it produces voices that sound like natural speech when best concatenation takes place. However, when poor concatenation takes place the quality perceived will be even much worse than that of diphone synthesizer that uses one instance of a unit. It is also indicated that unit selection offers good quality in a limited domain. This is because there are much smaller and controlled number of units. Examples of limited domains for unit selection speech synthesis include: telling time, fixed weather reports, telling sports results, etc.

Since unit selection synthesis uses several instances of each unit, it needs more storage than the ones that store only one unit instance. In addition to this, searching algorithm employed needs to search the optimal sequence from the stored units in the database making it time-consuming process [21].

Unit selection synthesis technique has been applied to develop synthetic voices for languages other than English. These include: Amharic [3], Arabic [17], and Turkish [18].

The work, [3], built unit selection voices for Amharic using Festvox[1]. To this end, Amharic phoneme set is described based on features like voicing, tongue position, tongue height, place of articulation and manner of articulation. In addition, letter-to-sound rules are defined to map orthographic characters to spoken form. Syllabification rules are also applied that decided the presence of epenthetic vowel in spoken form. Then, a speech database, consisting of 29,480 diphone instances made up of 801 unique diphones and 12,724 syllable instances made up of 1317 unique syllables, is created. Finally, unit selection voice for Amharic is built by applying unit clustering algorithm on the units of the database. Perceptual evaluation of the voice in terms of naturalness and intelligibility is performed using a method known as MOS. The result showed the speech synthesizer developed is categorized as good with MOS score of 2.9.

The work, [17], built a Festival based TTS system for Arabic using diphone concatenation method. To this end, Arabic phone set that suits the Festival framework is built. And, all diphones in the language were extracted from 200 sentences to build diphone database. The TTS system was evaluated using Diagnostic Rhyme Test (DRT) and the Modified Rhyme Test

[1] Festvox is a voice building framework which offers general tools for building unit selection voices for new languages

(MRT)[1] to test the intelligibility. The system is also tested with respect to naturalness, speed, sound quality, pronunciation, and stress/intonation. The results obtained, according to the authors, was satisfactory.

The paper, [18], dealt with designing and developing intelligible and natural-sounding corpus-based concatenative speech synthesis system for the Turkish language. It used unit selection algorithm which is based on Viterbi algorithm to find the best path in the network of speech units. The TTS system is evaluated in terms of intelligibility using DRT. The quality of the synthesized speech is evaluated using MOS method.

2.5. Related Works

Synthetic voices have a lot of applications. As an attempt is made to list out some of the applications of synthetic speech in the first chapter, it was discussed that synthetic voice is of particular importance, if applied to mobile phones.

Mobile phones have limited processing power and resources shared among different applications [14, 22, 41, 42]. However, as it is shown in [15], recent progress in technology helped mobile phones run multimedia applications. There are some works done and products made in the area of synthetic speech for mobile phones, which include Mobile Speak for Windows Mobile Smartphones [26] and Speech Synthesis for Mobile Phones [15].

Mobile Speak for Windows Mobile Smartphones is a screen reader application installed on a Windows Mobile-powered Smartphone which allows someone to use the device even if he/she

[1] The Diagnostic Rhyme Test uses a set of isolated words to test for consonant intelligibility in initial position, while the Modified Rhyme Test, which is a sort of extension to the DRT, tests for both initial and final consonant apprehension [1].

cannot read the visual screen. Information displayed on screen is rendered in synthesized speech output generated using TTS technology and routed through the device's speaker or a headset. Mobile Speak for Windows Mobile Smartphones supports more than 20 languages including English (US and UK), Spanish, Portuguese, Italian, French, German, Dutch (Netherlands and Belgium), Norwegian, Swedish, Finnish, Danish, Turkish, Polish, Czech, Greek, and Russian [26]. However, nothing else is described about the TTS technology.

The work in [15] combines speech application with face animation application to provide mobile phones SMS reading capability for Slovak language. The speech part is represented by speech synthesizer and animation is represented by 3D model of human face.

The system used concatenative speech synthesis approach with diphone unit. Diphone was used compared to other units because of its size (by stating that Slovak language as having 1550 frequently used diphones).

In their implementation, the input text to be synthesized is broken down into Speech Assessment Methods Phonetic Alphabet (SAMPA)[1] alphabet. Then before the actual synthesis takes place, they loaded the data part of the speech database (the database is in WAVE format which has header and data) into memory. The index file with the list of phonemes and diphones along with their boundaries is also loaded. The synthesis algorithm used checks whether a diphone is in the database or not. If the diphone is in the database, its sample is taken from the database and stored. If it is not in the database, it is replaced with two phonemes from which the diphone is composed of. For example, for diphone En (assuming it is not in database) samples for E are taken from "middle" to "end", samples for n are taken from "begin" to "middle". The algorithm repeats this

[1] SAMPA is a phonetic translation which uses only printable ASCII characters [15].

process until it finishes going through the whole text. Finally, audio is created and written in a WAVE file that can be played on mobile phone. The WAVE file format is a subset of Microsoft's RIFF (Resource Interchange File Format) specification for the storage of multimedia files [26]. However, this work did not explore the speech units other than diphones and also it documented nothing about how the system is tested.

2.6. Summary

This chapter discussed speech synthesis. The components of a TTS system have been examined. The commonly used speech synthesis techniques which include articulatory synthesis, formant synthesis, and concatenative synthesis have also been discussed. It is presented that the choice of unit in concatenative synthesis has impact on the quality of the synthetic speech produced and also the size of the database that stores speech segments. There is a trade-off between choosing longer units over shorter units. With longer units, good coarticulation, less concatenation, and good quality of speech are obtained. But the size of the database is large and generalizability can not be achieved. With shorter units, more concatenation, bad coarticulation, and poor quality of speech are obtained. An alternative to using longer units in a concatenative synthesis is storing several instances of a shorter unit from various context and prosody in order to produce natural sounding output at the expense of large database and time to search optimal sequence of the units that best matches the specification.

This chapter also discussed related works, i.e., speech synthesis for mobile phones for languages other than Amharic and also a product used for screen-reading has also been reviewed.

CHAPTER THREE

THE AMHARIC LANGUAGE WRITING SYSTEM AND PHONETICS

3.1. Introduction

This chapter presents the Amharic language with particular attention to its writing system and phonetics. As stated in the first chapter, Amharic is the official language of the Federal Government of Ethiopia having its own script [3, 32]. Among the 73 languages which are registered in the country, Amharic is a widely spoken language and it is one of the Semitic languages having its own script [3].

3.2. Amharic Writing System

Amharic writing system is phonetic, meaning, anyone who wants to write Amharic text can write the text as long as he/she can speak the language and has a good working knowledge of Ethiopic script. Unlike most languages, no one needs to learn how to spell Amharic words, nor to see a word first written in order to know how to spell it [33].

Amharic script (orthographic representation) consists of 33 core characters (shown in Table 3.1) each of which occurs in a basic form and six other forms. These seven forms of a character are known as orders and represent syllable combinations consisting of a consonant and following vowel. This representation of each character in 7 different forms makes the number of core characters to be 231 (33 * 7), also called syllographs (**ፊደል**) [32, 34].

Table 3.1: List of Amharic core characters

Order 1	Order 2	Order 3	Order 4	Order 5	Order 6	Order 7
ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ
መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ
ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ
ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ
ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ
ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ
በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ
ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ
ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ
ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ
ነ	ኑ	ኒ	ና	ኔ	ን	ኖ
ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ
አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ
ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ
ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ
ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ
ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ
ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ
ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ
የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ
ደ	ዱ	ዲ	ዳ	ዶ	ድ	ዶ
ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ
ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ
ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ
ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ
ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ
ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ
ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ
ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ

There are also characters in Amharic other than the core characters mentioned above. Some of these, as stated in [34], include:

- Character ሸ that has also 7 forms or orders as core characters,
- 44 other symbols (of which 20 are most commonly used) that contain a special feature representing labialization,
- Punctuation marks such as ፡ (Ethiopic word space), ። (Ethiopic full stop), ፣ (Ethiopic comma), ፤ (Ethiopic semicolon), and borrowed symbols like ?, !, (,), and
- Numerals consisting of a single character for 1 to 10, for multiples of 10 (20 to 90), for 100 and 1000. The numeral symbols are shown in Table 3.2.

Table 3.2: The numeral symbols of Amharic

1	፩	6	፮	20	፳	70	፸
2	፪	7	፯	30	፴	80	፹
3	፫	8	፰	40	፵	90	፺
4	፬	9	፱	50	፶	100	፻
5	፭	10	፲	60	፷	1000	፼

3.3. Amharic Phonetics

Phonetics is the scientific study of speech whose central concerns are the discovery of how speech sounds are produced, how speech sounds are recorded with written symbols, and how we hear and recognize different sounds [20]. Due to the differences in how speech sounds are produced, Amharic speech sounds are of basically two types: consonants and vowels [35].

The following subsections brief about how consonants and vowels are produced in Amharic language, which are taken from [35], and also the transcription of Amharic scripts.

3.3.1. Consonants

Consonants are articulated in the presence of constrictions in the throat and obstructions in the mouth (tongue, teeth, lips) as we speak. According to [35], there are 27 consonant sounds (phonemes) in Amharic language. The three different characteristics that differentiate the various consonants in the language are: voicing, manner of articulation, and place of articulation. According to the first aspect, that is voicing, consonants are classified into voiced and unvoiced. In view of the manner of articulation, consonant sounds can be classified into: stops, fricatives, affricatives, nasals, liquids, and glides. Based on place of articulation, consonants are classified into: labial, alveolar, palatal, velar, labio-velar, and glottal. Table 3.3 shows Amharic characters categorized based on the above mentioned characteristics. The detailed description about these characteristics and also the characters under different categories is given in the following paragraphs.

Stops

Stop consonants, also called plosives, are produced when the vocal tract is closed causing stop or attenuated sound. When the tract reopens, it causes noise-like, impulse-like or burst sound. The stop can occur at the lips, at the alveolar when the tip of the tongue touches it, at the palate when the middle of the tongue touches it, at the velum when the back of the tongue touches it, or at the glottis. As it can be seen from Table 3.3, Amharic consonants that are found in stop category include: /ፐ/, /ቡ/, /ጵ/, /ት/, /ድ/, /ጥ/, /ክ/, /ግ/, and /ቅ/. These sounds, based on the place of articulation, are classified into: labial – (/ፐ/, /ቡ/, and /ጵ/), alveolar – (/ት/, /ድ/, and /ጥ/), and

palatal – (/ክ/, /ግ/, and /ቅ/). In terms of voicing, /ፕ/, /ጵ/, /ት/, /ጥ/, /ክ/, and /ቅ/ are unvoiced or voiceless, while /ብ/, /ድ/, and /ግ/ are voiced.

Table 3.3: Consonants with their features [3]

		Labial		**Alveolar**		**Palatal**		**Velar**		**Labio-velar**		**Glottal**	
Stop	Voiceless	p	ፕ	t	ት			k	ክ	kx	ኳ		
	Voiced	b	ብ	d	ድ			g	ግ	gx	ጓ		
	Glottalized	px	ጵ	tx	ጥ			q	ቅ	qx	ቋ		
Fricative	Voiceless	f	ፍ	s	ስ	sx	ሽ					h	ሀ
	Voiced	v	ቭ	z	ዝ	zx	ዥ						
	Glottalized			xx	ጽ							hx	ኀ
Affricative	Voiceless					c	ች						
	Voiced					j	ጅ						
	Glottalized					cx	ቈ						
Nasal	Voiced	m	ም	n	ን	nx	ኝ						
Liquid	Voiced			l	ል								
				r	ር								
Glide		w	ው			y	ይ						

Fricatives

Fricatives are produced when the vocal tract is constricted in some place so that the turbulent flow causes noise which is modified by the vocal tract resonances. Amharic consonants classified under this category include: /ፍ/, /ቭ/, /ስ/, /ዝ/, /ጽ/, /ሽ/, /ዥ/, and /ሀ/. The constriction can occur between upper teeth and lower lip, the tip of the tongue and the alveolar, the middle of the tongue and the palate, or the back of the tongue and the velum. Based on this, /ፍ/ and /ቭ/ are labial, /ስ/, /ዝ/, and /ጽ/ are alveolar, /ሽ/ and /ዥ/ are palatal, and /ሀ/ is glottal. Among the fricatives, /ዝ/, /ዥ/, and /ዥ/ are voiced, while /ፍ/, /ስ/, /ጽ/, /ሽ/, and /ሀ/ are voiceless.

Affricatives

Affricatives have the characteristics of both stops and fricatives. An affricative is a type of consonant consisting of a plosive followed by a fricative with the same place of articulation. The

three Amharic consonants categorized under this category are: /ች/, /ጅ/, and /ጭ/. All of them are palatal and among them /ጅ/ is voiced, while /ች/ and /ጭ/ are voiceless.

Nasals

A nasal consonant is the one in which the air escapes only through the nasal cavity during its articulation. For this to happen, two articulatory actions are necessary: firstly, the velum must be lowered to allow air to escape past it, and secondly, a closure must be made in the oral cavity to prevent air from escaping through it. The closure may be at any place of articulation from bilabial at the front of the oral cavity to uvular at the back. Amharic consonants classified under this category are /ም/, /ን/, and /ኝ/. All consonants under this category are voiced and based on place of articulation, /ም/ is labial, /ን/ is alveolar, and /ኝ/ is palatal.

Liquids

Liquids are consonant sounds that are produced when the tip of the tongue closes the oral cavity leaving a side route for the air flow. Amharic consonants under this category include /ል/ and /ር/. Both sounds are alveolar and voiced.

Glides

Glides, also known as semivowels, are sounds with vowel and consonant features. Like vowels there is no major obstruction of air pressure emanating from the lung. Functioning as consonants they precede vowels that form the nucleus of syllables. In Amharic, the two semivowels are /ው/ and /ይ/. In view of place of articulation, /ው/ is labial while /ይ/ is palatal. But both /ው/ and /ይ/ are voiced in manner of articulation.

3.3.2. Vowels

Unlike consonants, there is no major constriction of air flow in the oral cavity during vowel articulation. The tongue plays major role in the articulation of vowels. Its role can be seen in two ways. Firstly, its vertical position (height) in the mouth showing how the mouth is open. It can be high close to the palate, in the middle, or low. Secondly, the movement of the tongue horizontally in which the front, the middle, or the back of the tongue plays the major role. In addition to the tongue, the lip is also important in the articulation of vowels with its shape being either rounded or unrounded. Figure 3.1 shows Amharic vowels based on the height of the tongue and its horizontal movement.

The seven vowels in Amharic language include: /ʾ**ኧ**/, /**ኡ**/, /**ኢ**/, /**ኣ**/, /**ኤ**/, /**እ**/, and /**ኦ**/. All are oral and voiced. When we say the vowels are oral, we mean air flow passes through oral cavity during their articulation.

Based on the vertical position of the tongue, Amharic vowels are categorized into three: high, middle, and low. /**ኢ**/, /**እ**/, and /**ኡ**/ are articulated when the body of the tongue is moved up making them categorized under high. /**ኤ**/, /ʾ**ኧ**/, and /**ኦ**/ are uttered when the body of the tongue is at its resting position making them categorized under middle. The only vowel that is articulated when the body of the tongue is moved down is /**ኣ**/ making it classified as low.

According to the movement of the body of the tongue toward the front or the back of the mouth, Amharic vowels are also categorized into three: front, mid, and back. Front vowels are articulated when the body of the tongue is moved forward, mid vowels are articulated when the body of the tongue is at its resting position, back vowels are uttered when the body of the tongue is moved

backward. Based on this, /ኢ/ and /ኬ/ are front, /እ/, /ኧ/, and /ኣ/ are mid, and /ኡ/ and /ኦ/ are back.

Based on the shape of the lip during the articulation of vowels, Amharic vowels are classified into two: rounded and unrounded. Among the Amharic vowels, /ኡ/ and /ኦ/ are rounded, while /ኧ/, /ኢ/, /ኣ/, /ኬ/, and /እ/ are unrounded.

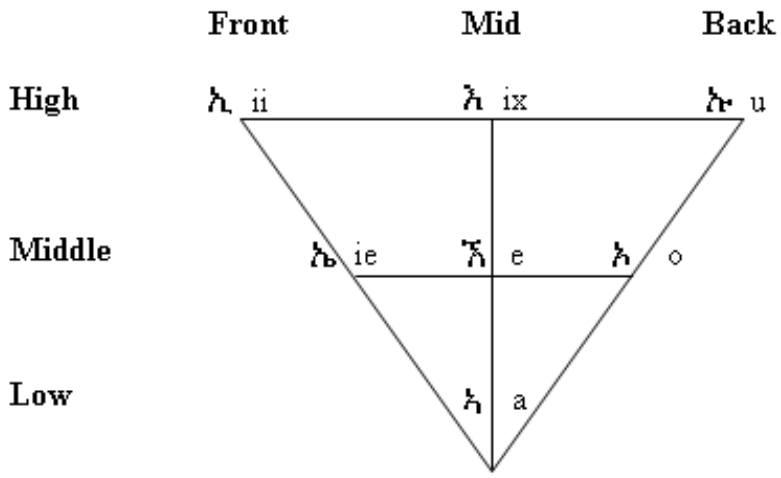

Figure 3.1: Vowels with their features [3]

The features of Amharic vowels can be summarized as follows:

- Amharic vowels are voiced and oral.
- When the vowels are articulated, there is no major constriction of air flow.
- The tongue plays pivotal role in the articulation of the vowels.

3.3.3. Transcription of Amharic Characters

Transcription refers to the writing down of spoken utterance using a suitable set of symbols. In its original meaning the word implied converting from one representation (e.g. written text) into another (e.g. phonetic symbols) [20].

As stated in [1], transcription is needed because written text in most languages does not correspond to its pronunciation. Hence, in order to describe the correct pronunciation some kind of symbolic presentation is needed. There were some efforts made to construct language independent phonemic alphabets during the last decades. Among these, IPA (International Phonetic Alphabet) [29] and SAMPA [31] are some to list. IPA, which is one of the best known language-independent phonemic alphabets, consists of a huge set of symbols for phonemes, suprasegementals, tones/word contours, and diacritics. On the other hand, SAMPA is designed to map IPA symbols to 7-bit printable ASCII characters to alleviate the complexity and the use of Greek symbols that make IPA alphabets unsuitable for computers which usually require standard ASCII as input. Even if there are several other phonetic representations and alphabets used in present TTS systems, there is no single generally accepted phonetic symbol [1].

According to [3], transliteration can be used as an alternative to using the IPA alphabets. The transliteration scheme used in this research work is shown in Table 3.4. It is designed based on the orthographic ordering of the script and acoustic similarity of the letters. The transliteration scheme used in [3] is adopted in this study. Based on the transliteration scheme, Amharic name **ጽናት** is transcribed into [xxixnat]. Square brackets, [], are used during phonetic transcription.

Table 3.4: (I-X NOTATION) Amharic phonetic list, IPA equivalence and its transliteration table [3]

IPA	Transcription	Amharic equivalence
Consonants		
[p]	[p]	ፕ
[t]	[t]	ት
[k]	[k]	ክ
[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
[ʊ]	[u]	ኡ
[ɪ]	[ii]	ኢ
[ɑ]	[a]	ኣ
[e]	[ie]	ኤ
[ɨ]	[ix]	እ
[o]	[o]	ኦ

3.4. Summary

In this chapter, an attempt is made to give a brief description about the Amharic writing system and phonetics. Most Amharic characters (core characters) have seven different forms known as orders. It is shown that the language also contains characters other than the core characters.

Amharic phonetics in particular attention to consonants, vowels and transcription of Amharic characters has been presented. Consonants are sounds that are articulated with the presence of major obstruction of air pressure in the vocal tract, while vowels are produced without the obstruction of air pressure in the vocal tract. Detailed descriptions about how Amharic consonants and vowels are articulated are given in this chapter. The chapter also described the transcription of Amharic characters.

CHAPTER FOUR

DESIGN OF AMHARIC SPEECH SYNTHESIZER FOR MOBILE PHONE

4.1. Introduction

This chapter deals with the design of Amharic Speech Synthesizer for Mobile Phone (ASSMP). Various issues that should be considered in the design of speech synthesizer for mobile phone for Amharic language keeping in mind that the limited processing and storage capacity of mobile phones are presented. In addition, the various components of the speech synthesizer, the function(s) of each component, and also how components communicate each other are also described in detail.

4.2. Design Goal

As it is stated in [28], the majority of speech synthesizers that are in use for commercial applications use the same basic principle – that of concatenation of stored speech units. The fundamental goal of speech synthesis is to produce natural sounding and intelligible speech for any input.

4.3. Design Issues

Traditionally, concatenative speech synthesis was hampered by the CPU and memory requirements of speech synthesizers. However, these issues are not much pronounced in today's computers due to their high processing and memory capacity [28]. But, these same issues should

be taken into account when limited processing power and memory capacity of mobile phones are considered.

Unit size has an impact on the quality of synthetic speech produced, the size of the database needed to store the units and time required to generate the waveform. Choosing one unit over the other is challenging as it requires balancing the quality of the speech, the size of the database and the time needed to generate the waveform. After the choice has been made, the database need to have full coverage of the speech units needed to produce an utterance in a particular domain.

4.3.1. Unit Size

The second chapter detailed that there are different types of speech units that can be used in building a concatenative speech synthesizer. Phonemes, diphones, syllables, words, phrases, and even sentences are some to list. It is also discussed that the choice of a unit has an impact on the size of the database and also the quality of the synthetic speech produced. According to [16], there is a tradeoff between shorter and longer units. Longer units are advantageous over shorter units on preserving naturalness over longer time-scales, and result in fewer concatenation points in the synthetic speech. However, longer units can also be very numerous, to the point of being prohibitively numerous in the case of words, and therefore shorter units, which are much less numerous, are also attractive.

In this thesis work, the three speech units of Amharic language taken into consideration are: phonemes, diphones, and syllables. Words, phrases, and sentences are not taken into consideration because there are a lot of such units in a particular language. Hence, storing such numerous units needs a huge amount of storage space and even the units in the language can not

be exhaustively listed. As stated in the second chapter, such units are neither generalizable nor trainable.

When someone selects one of the aforementioned units as a unit for concatenation, especially in an environment where resource (processing and storage capacity) is limited, there come issues like how much of the unit are there in the language (in this case Amharic language), how much storage space is needed to store the units, and also how CPU intensive the synthesis process (generation of the waveform) is. All these issues should be considered keeping in mind the quality of the utterance – which is the design goal of a speech synthesizer – produced after the synthesis process has taken place.

In the following subsections, phonemes, diphones, and syllables are examined in detail having in mind the quality of the synthetic speech produced and also addressing the limited storage and memory capacity of mobile phones. As stated in chapter one, this thesis work focuses on specific domain. Amharic names and numbers are taken as domain for the research. For this, a list of names (884 unique names) is collected from telephone directory and website. Then transliteration is done based on the transliteration scheme depicted in Table 3.4. An epenthetic vowel (ix) is inserted into names where it is needed according to the rules stated in [3].

4.3.1.1. Phonemes

Chapter two detailed that there are context-dependent and context-independent phonemes. An algorithm has been developed and based on it a code that lists the phonemes has been written. In addition, context information is also taken into consideration, which includes: left contexts, right contexts, and both left and right contexts. Table 4.1 shows the 34 phonemes obtained from the collected data.

Table 4.1: List of phonemes

No.	Phoneme	No.	Phoneme
1	e	18	ii
2	a	19	o
3	ix	20	k
4	m	21	f
5	r	22	q
6	y	23	z
7	n	24	sx
8	s	25	tx
9	l	26	xx
10	t	27	nx
11	b	28	c
12	d	29	j
13	u	30	p
14	h	31	cx
15	g	32	px
16	ie	33	zx
17	w	34	v

It has also been discussed in the second chapter that using context-independent phonemes results in many audible discontinuities making the system unintelligible. This problem can be minimized by using context-dependent phonemes. Table 4.2 lists some phonemes with their left contexts. The table does not list all the phonemes along with their left contexts because the size of the list is too large to display as the whole list contains 451 phonemes with their left contexts. The table also shows that there is additional character (#) used other than the characters used to represent the phonemes. It is used here as a delimiter that shows the start and the end of a word, in this case Amharic names. Phoneme /p/, e.g., as shown in Table 4.2, has four phonemes to the left of it in the collected data. These are: /s/, /e/, /ii/, and /ix/. Similar explanation holds true regarding context information given in Table 4.3, Table 4.7 and Table 4.8.

Table 4.2: List of phonemes with their left contexts

No.	Phoneme	Left Context	No.	Phoneme	Left Context
1	p	s	12	a	#
2	p	e	13	e	m
3	p	ii	14	a	m
4	p	ix	15	e	y
5	p	#	16	m	e
6	t	k	17	n	a
7	k	l	18	a	h
8	b	t	19	e	t
9	d	g	20	ii	g
10	g	t	21	ie	u
11	px	o	22	o	sx

Table 4.3 shows some phonemes along with their right contexts. The whole list actually contained 449 unique phonemes with their right contexts.

As discussed in chapter two, in context-dependent phonemes, if the context is limited to the immediate left and right phonemes, then the unit is known as triphone. Table 4.4 shows some phonemes with left and right contexts. The number of triphones obtained from the data is 1943.

Table 4.3: List of phonemes with right contexts

No.	Phoneme	Right Context
1	ix	#
2	m	e
3	y	e
4	a	n
5	h	a
6	a	y
7	a	b
8	u	g
9	m	xx
10	ie	f
11	o	y

Table 4.4: List of phonemes with left and right contexts (triphones)

No.	Left Context	Phoneme	Right Context
1	#	y	e
2	e	m	e
3	b	ix	r
4	e	r	e
5	e	b	r
6	a	r	ix
7	h	a	n
8	a	w	ii
9	y	a	n
10	u	ie	l
11	w	o	l

4.3.1.2. Diphones

Diphones, as mentioned in chapter two, are speech units that extend from the central point of the steady state part of one phoneme to the central point of the following one, so they contain transition between adjacent phonemes. Hence, diphones are, on average, one phoneme long. It is also mentioned that in a language with N phonemes, there are potentially N^2 diphones. In practice, the number of diphones is smaller than that. Table 4.1 listed that there are 34 phonemes (of Amharic language) extracted from the data. Theoretically, the number of diphones is expected to be 1156 (34*34). Actually, the number of unique diphones obtained from the data is 420 of which some are listed in Table 4.5. According to [3], 801 diphones are obtained from the data they collected. The variation is due to the data used. The number of diphones would have increased if a large amount of data had been used. Note that 884 unique names have been used for the purpose of analysis in this thesis work.

Table 4.5: List of diphones

No.	Diphone
1	m-e
2	a-n
3	t-e
4	e-s
5	a-m
6	g-e
7	a-b
8	u-g
9	k-s
10	ii-f
11	o-y

4.3.1.3. Syllables

It has been stated in [35] that the syllabic structure of Amharic has the following template: V[1], VC[2], VCC, CV, CVC, and CVCC. It has been cited in [3] that there are some scholars arguing that the syllabic structure is only composed of CV and CVC only. In this research, the analysis is done only on CV syllables because they commonly occur in the writing system of the language. Table 4.6 shows some syllables from the list of 158 unique syllables obtained from the collected data. According to [3], 1317 unique syllables are obtained from the data using the syllable structure stated in [35]. The variation is due to the use of only CV syllables in this research.

[1] V – Vowel.
[2] C – Consonant.

Table 4.6: List of syllables

No.	Syllable
1	a
2	me
3	ye
4	ha
5	re
6	ge
7	da
8	txa
9	nxo
10	jo
11	zxie

Table 4.7: List of syllables with their left contexts

No.	Syllable	Left Context
1	a	#
2	me	#
3	ne	ix
4	re	ix
5	ka	mii
6	re	fe
7	ya	rii
8	wii	ma
9	yo	sii
10	za	ge
11	zxie	ga

Even though the whole list contained 1403 unique syllables with their left contexts, Table 4.7 shows sample syllables along with their left contexts.

Table 4.8 depicts sample syllables along with their right contexts. The whole list contained 1422 unique syllables with their right contexts.

Table 4.8: List of syllables with their right contexts

No.	Syllable	Right Context
1	ix	#
2	ix	ha
3	mii	ka
4	ka	sa
5	te	ma
6	ge	nxe
7	be	re
8	de	mii
9	bii	ya
10	tie	wo
11	ba	re

4.3.2. Selection of Appropriate Unit

In the above subsections, three speech units along with their contexts have been presented. This section presents the way how the unit, which is appropriate for mobile phone for Amharic language, is selected. The unit that is going to be used as a unit for concatenation is chosen among the other units based on the quality of speech, the size of the database needed to store the unit, and processing time needed to generate the waveform. And the chosen unit is termed to be appropriate unit.

As it is discussed earlier, the main goal of speech synthesis is producing natural sounding and intelligible speech output. It is also discussed that the choice of a unit has an impact on the quality of speech produced. In addition, it has an impact on the size of the database, which is an issue in a resource-constrained environment. It has also been briefed that someone can store only one instance of each unit or multiple instances of a unit from various contexts. In the latter case, the type of synthesis is called unit selection synthesis as discussed in chapter two. In the following paragraphs, each unit with respect to quality, size, and also time is briefed.

The most appealing unit in terms of size is context-independent phoneme if only one instance of each phoneme is used. The 34 phonemes listed in Table 4.1 only need to be stored.

However, as it has been discussed in the second chapter, using context-independent phonemes results in audible discontinuities producing a speech with poor quality. The problem of having audible discontinuities can be alleviated by using context-dependent phonemes.

The other unit to use is phoneme with both left and right contexts, which is called triphone. As mentioned in Section 4.3.1, there are more than 1900 unique instances of triphones in the collected data. According to [44] and [45], the number of unique names (which is our domain) in a particular language tends to be very large. This makes the number of unique triphones to be much larger than the figure indicated here, which in turn makes the database grow larger. As discussed in chapter two, triphones offer a speech of good quality which is better than that of diphones and context-independent phonemes.

One can use diphone as a unit for concatenation. As mentioned in Section 4.3.1, there are 420 unique diphones. A diphone is on average one phoneme long. Diphones are taken from a stable region of the speech making them produce speech with good quality.

Another unit mentioned in Section 4.3.1 is syllable. As stated in chapter two, coarticulation effect within a syllable is well-handled than across syllables. According to [1], using syllables as unit of concatenation is not very reasonable since coarticulation effect is not included across syllables. To address coarticulation, syllable with either context can be used.

An alternative to storing one instance of a unit, is storing several instances of the same unit taken from different contexts[1]. The synthesis technique (called unit selection synthesis as mentioned previously) can use phoneme, diphone, or syllable as a basic unit. During synthesis optimal sequence is searched and concatenation takes place. Searching the optimal sequence requires a lot of processing time [21].

To give some idea about searching in unit selection synthesis, assume an input text contains T units (e.g., phonemes) and there are N instances of each phoneme. Then there are N^T unique possible sequences of units. Hence an algorithm employed in unit selection synthesis needs to search an optimal sequence from N^T unique possible sequences, which takes an astonishing length of time regardless of computer speed. However, one can employ dynamic programming algorithm, for example Viterbi algorithm, which operates in N^2T time. This algorithm takes linear time in T, which is an improvement on the exponential N^T time of exhaustive search. Searching takes much time in unit selection when compared to searching a unit from a database where only one instance of each unit is stored [21]. As stated in the second chapter, unit selection synthesis gives us more natural sounding speech.

In the above paragraphs, the various units with respect to quality, size, and time have been presented. As discussed previously, even though context-independent phonemes are fewer in number, they result in audible discontinuities. Triphones produce speech with quality better than the one produced using diphones, but, they are much larger in number than diphones. In the case of syllables coarticulation effect across syllables is not well-handled. To handle coarticulation, context information can be included. Unit selection synthesis is not chosen, since it requires larger database and much more processing time than the ones that use single instances even if it

[1] Context information given in Table 4.2, Table 4.3, Table 4.7 and Table 4.8 can be used in unit selection synthesis.

produces relatively better synthetic speech in terms of naturalness and intelligibility. Hence, diphone has been selected as an appropriate unit for mobile phone.

4.3.3. Coverage

The database (which stores diphones in our case) should contain all instances of the unit chosen in order to utter arbitrary text in a particular domain, even if, according to [23], obtaining full coverage of diphone inventory is very hard. In this research work the main focus, as discussed in the first chapter, was on Amharic names that are going to be saved in mobile phones. Hence, an attempt is made to store diphones necessary to utter every name among the list of 884 names collected.

4.4. Architecture of ASSMP

In this section the basic components of ASSMP are presented. As any speech synthesizer does, ASSMP consists of two basic modules: the NLP module and the DSP module. As stated in the second chapter, the NLP module is responsible for performing text analysis, phonetic analysis, and prosody generation. The DSP module, on the other hand, is responsible for generating waveform, that is, for producing synthetic speech. The architecture of the ASSMP is depicted in Figure 4.1. Each of the components (shown in Figure 4.1) and the intercommunication of components are presented in the following subsections.

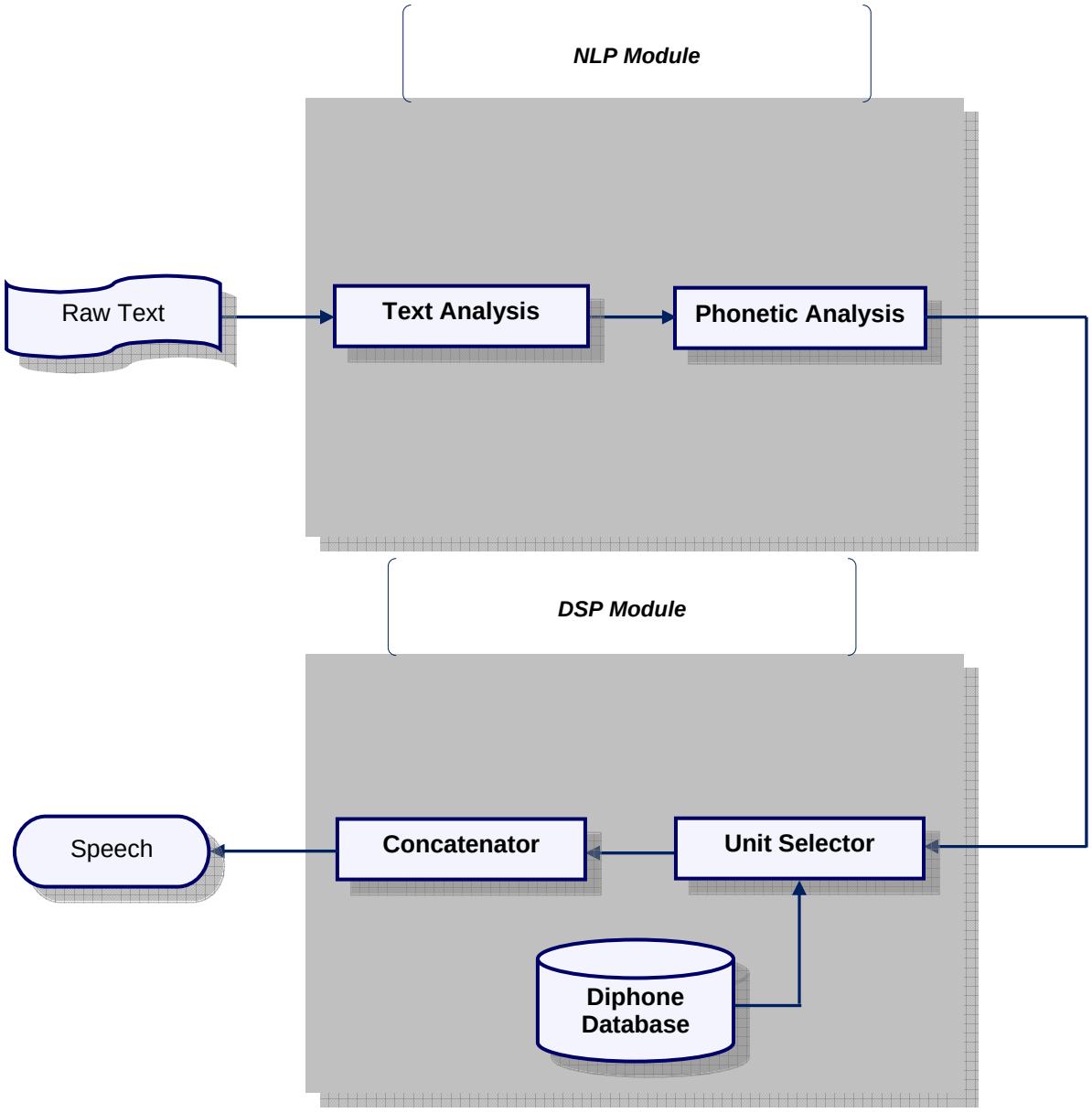

Figure 4.1: Architecture of Amharic Speech Synthesizer for Mobile Phone

4.4.1. The NLP Module

As it is discussed in the second chapter, the NLP module is responsible for performing text analysis, phonetic analysis, and prosody generation. Even though a component responsible for

modeling prosody is not included in the architecture explicitly, appropriate selection of training data set minimizes its impact on the quality of the synthesized speech.

The NLP module accepts raw text in a transliterated form. The text analysis component of the NLP module is responsible for converting the raw text into orthographic representation. The output of the text analysis component is fed into the phonetic analysis component of the NLP module. The phonetic analysis component converts the text to a set of phonemes. The flowchart of the algorithm employed for the conversion is depicted in Figure 4.2. The algorithm is developed based on the nature of characters used in the transliteration scheme. The set of phonemes is given to the unit selector component of the DSP module.

The algorithm shown in Figure 4.2 first checks the last character of a string. If the last character is ‘x’ or ‘i’, it extracts the last character along with its previous character. If the last character is ‘e’, it checks whether the preceding character is ‘i’. If so, the algorithm extracts character ‘i’ and ‘e’ together, otherwise, the algorithm extracts character ‘e’ only. If the last character is different from ‘x’, ‘i’, or ‘e’, the algorithm extracts only the last character. This whole process listed in this paragraph is repeated until all the characters in the string have been processed.

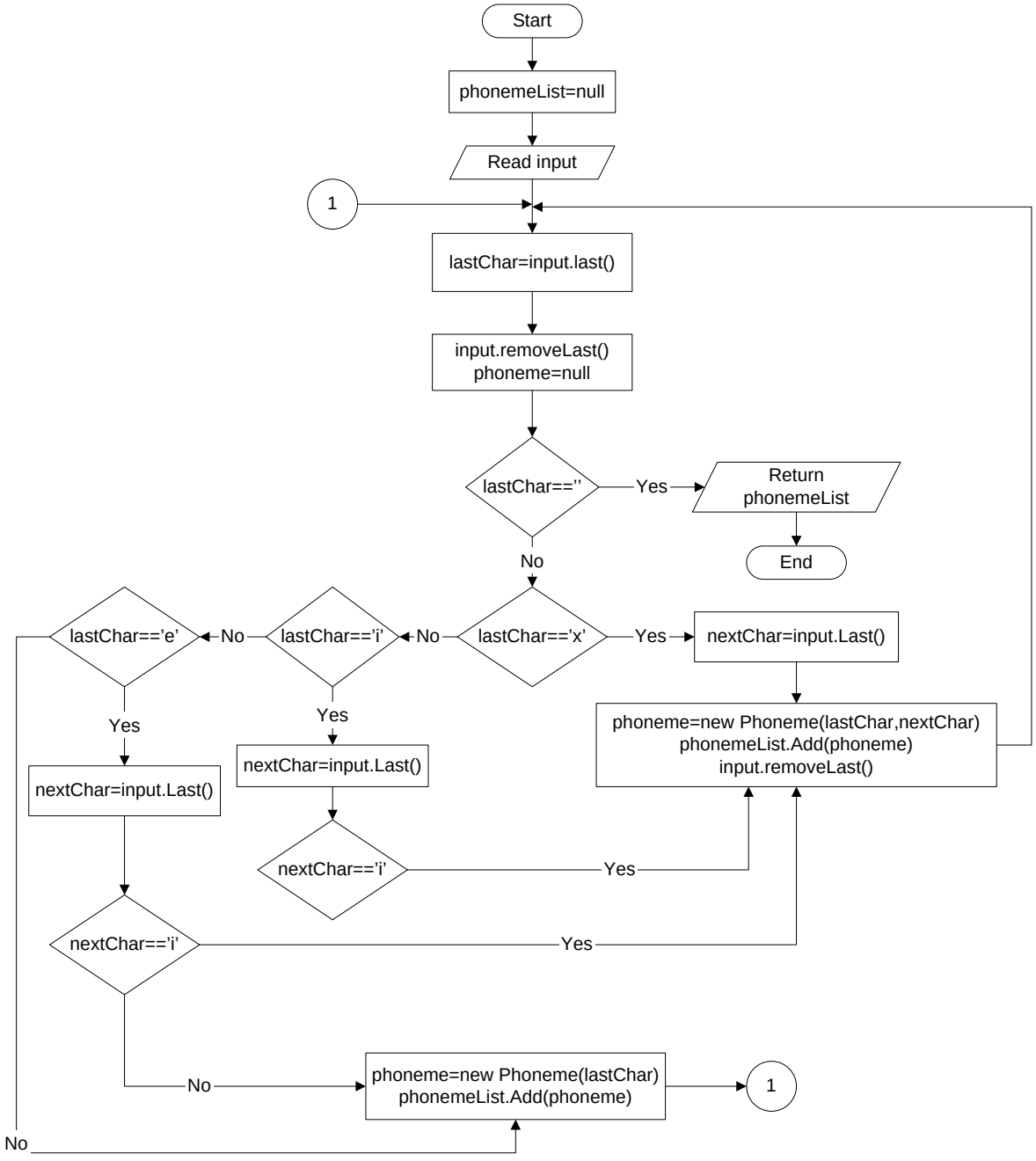

Figure 4.2: Flowchart of the algorithm used for converting a text to a set of phonemes

4.4.2. The DSP Module

As discussed in the second chapter, the DSP module of a speech synthesizer is responsible for the generation of waveform. The DSP module, as it can be seen from Figure 4.1 contains two components: the unit selector component and the concatenator component.

The unit selector component accepts a set of phonemes as an input and then converts it to a set of diphones. After the conversion has taken place, the unit selector component fetches each diphone (the wave file) from the database and gives the diphone to the concatenator component until all the diphones in the list are processed.

	Field Name	Field Size (bytes)
Header	ChunkID	4
	ChunkSize	4
	Format	4
	Subchunk1ID	4
	Subchunk1Size	4
	AudioFormat	2
	NumChannels	2
	SampleRate	4
	ByteRate	4
	BlockAlign	2
	BitsPerSample	2
	Subchunk2ID	4
	Subchunk2Size	4
Raw Data	Data	Subchunk2Size

Figure 4.3: Wave Format [adapted from 26]

The concatenator component accepts the first diphone and strips off the header part of the wave file and then calculates the values of all fields. The concatenator also stores the raw data (wave

file without the header). Figure 4.3 shows the structure of a wave file in RIFF format. The header part of a wave file is 44 bytes in size and contains 13 fields. If one takes two diphones extracted from a speech recorded with the same sampling frequency, bit rate, etc., then each corresponding field of the two diphones has the same value except the following two fields: ChunkSize, and Subchunk2Size. The value of Subchunk2Size tells us how many bytes the raw data (actual sound) contains. The value of ChunkSize is equal to the sum of Subchunk2Size and 36.

For each successive diphones sent from the unit selector component, the concatenator component first strips of the header, calculates the value of the Subchunk2Size, adds the value of the SubchunkSize2 to both previous values of SubChunk2Size and ChunkSize, and finally appends the raw data on the previous raw data. After all the diphones have been processed, the concatenator component creates a wave file that can be played on mobile phones. The flowchart of the algorithm used for concatenating diphones and then creating another wave file is given in Figure 4.4.

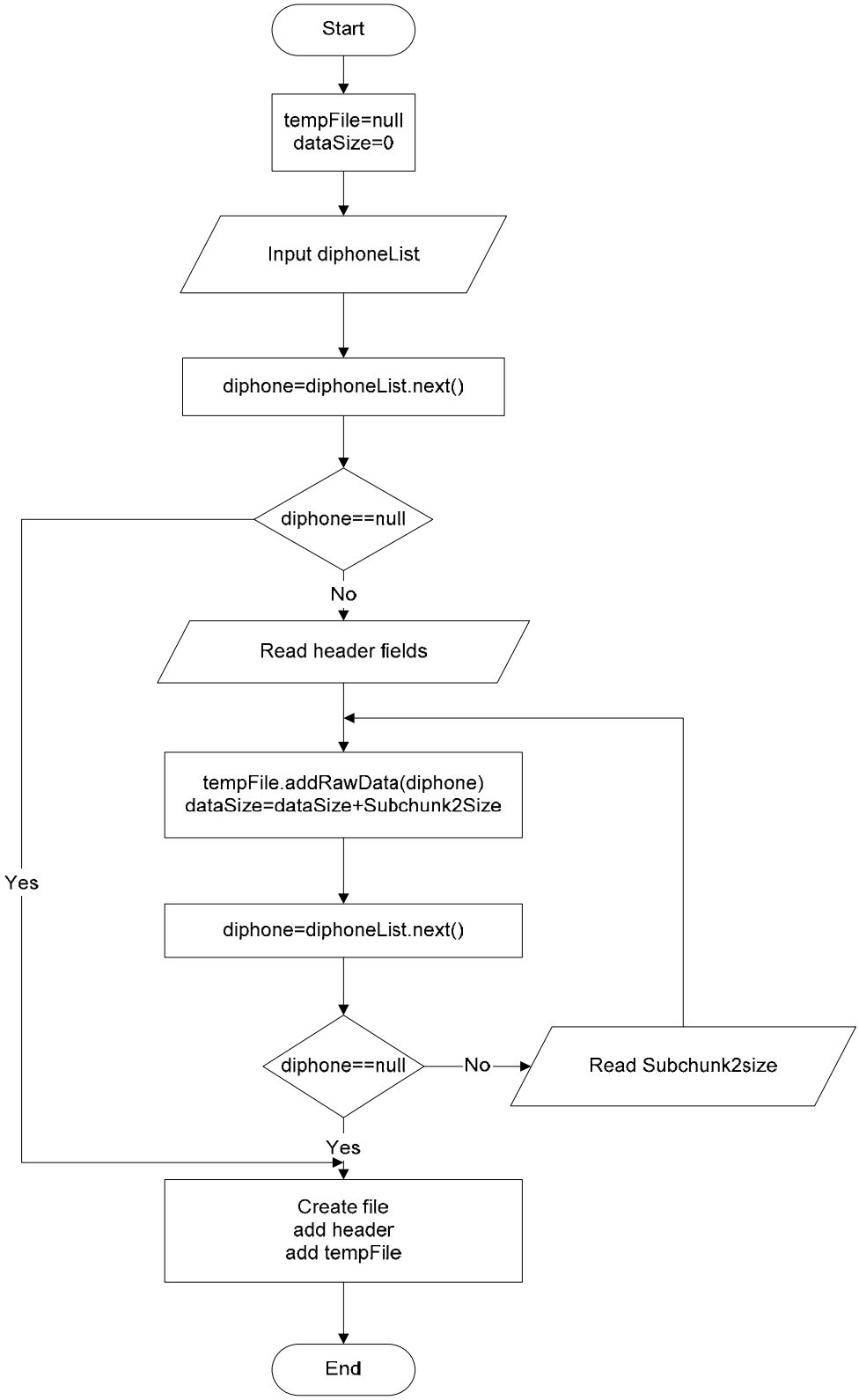

Figure 4.4: Flowchart of the algorithm used for concatenating diphones and creating a wave file

4.5. Summary

This chapter addressed the various issues in the design of Amharic speech synthesizer for mobile phones. The various units that can be used as a unit for concatenation have been discussed. Comparison among the units in terms of quality of speech produced using the unit as a unit for concatenation, the size of the database in relative terms, and also processing time which is also discussed in relative terms has been done. After considering various issues we come up with diphones as unit for concatenation.

The chapter also documented the architecture of the speech synthesizer. Then the various components found in each of the two modules: the NLP module and the DSP module have been detailed. The process of converting raw text into synthetic speech is also explained in this chapter.

CHAPTER FIVE

IMPLEMENTATION AND EXPERIMENT

5.1. Introduction

This chapter describes the implementation of the designed algorithms for Amharic speech synthesizer for Mobile Phone whose details are explained in the pervious chapter. The chapter also discusses how experimentation is done and the results found.

As it is stated in the previous chapter, ASSMP has two modules: the NLP module and the DSP module. The NLP module accepts a raw text that has already been transliterated as an input. The DSP module is, as it is discussed in Chapter Two and also in Chapter Four, responsible for the generation of actual waveform. To this end, it accepts the output of the NLP module as an input.

5.2. The Development Environment

The implementation has taken place on a standalone PC whose specification is given in the first chapter. Additional tools, stated in chapter one, have been used to supplement the development process. Appropriate programming language to implement the algorithms described in the previous chapter has been searched and J2ME has been selected due to the following reasons:

- Portability – applications written using J2ME will be directly portable to mobile device.
- Security – J2ME is well-known for its ability to safely run downloaded codes.
- Our experience of the language, Java (J2SE), made us easily learn J2ME.

J2ME is not a specific piece of software or specification. All it means is Java for small devices. Small devices range in size from pagers, mobile phones, and personal digital assistants (PDAs), all the way up to things like set-top boxes [42].

J2ME is divided into *configurations*, *profiles*, and *optional APIs*, which provide specific information about APIs and different families of devices. A configuration is designed for a specific kind of device based on memory constraints and processor power. It specifies a Java Virtual Machine (JVM) that can be easily ported to devices supporting the configuration. It also specifies some subset of the J2SE APIs that will be used on the platform, as well as additional APIs that may be necessary. Profiles are more specific than configurations. A profile is based on a configuration and adds APIs for user interface, persistent storage, and whatever else is necessary to develop running applications.

There are two types of configuration: the Connected Device Configuration (CDC) and the Connected, Limited Device Configuration (CLDC).

A connected device has, at a minimum, 512 KB of read-only memory (ROM), 256 KB random access memory (RAM), and some kind of network connection. The CDC is designed for devices like television set-top boxes, car navigation systems, and high end PDAs. The CDC specifies that a full JVM (as defined in Java Virtual Machine Specification, 2nd edition) must be supported.

CLDC is a configuration that encompasses mobile phones, pagers, PDAs, and other devices of similar size. It is designed for devices with 160 KB to 512 KB of memory available for the Java platform. The “limited connection” simply refers to a network connection that is intermittent and probably not very fast. (Most mobile telephones, for example, typically achieve data rates of 9.6 Kbps.) Between the small screen size, limited memory, and slow network connection,

applications designed in the CLDC space should be very sparing with the use of the network connection. The CLDC is based around a small JVM called the KVM whose name comes from the fact that it is a JVM whose size is measured in kilobytes rather than megabytes. Because of its small size, the KVM can not do everything a JVM does in the J2SE world.

A profile is layered on top of a configuration, adding the APIs and specifications necessary to develop applications for a specific family of devices. There are several different profiles being developed under the Java Community Process (JCP). These include: the Foundation Profile, the PDA Profile (PDAP), the Mobile Information Device Profile (MIDP), etc.

The Foundation Profile is a specification for devices that can support a rich networked J2ME environment. It does not support a user interface; however, other profiles can be layered on top of the Foundation Profile to add user interface support and other functionality. The PDA profile, which is built on top of CLDC, is designed for palmtop devices with a minimum of 512 KB combined ROM and RAM (and a maximum of 16 MB).

The Mobile Information Device Profile provides a standard Java runtime environment for today's most popular mobile information devices. A Mobile Information Device has the following characteristics:

- 128 KB of non-volatile memory for the MIDP implementation
- 32 KB of volatile memory for the run time heap
- 8 KB of non-volatile memory for persistent data
- A screen of at least 96 * 54 pixels

- Some capacity for input, either by keypad, keyboard, or touch screen
- Two-way network connection, possibly intermittent

According to the above specification, mobile phones and pagers are Mobile Information Devices.

5.3. The NLP Module

This section describes, how the NLP module whose design given in the previous chapter, is implemented. Before implementing the classes needed to build the synthesizer, an attempt is made to display Amharic characters onto the emulator. To this end, the emulator is customized so that it can be able to display Amharic characters. Some of the customizations made here are adapted from [43].

The customization actions taken in order to display Amharic characters onto mobile phone's emulator are:

- Copying the folder named *DefaultColorPhone* under C:\WTK2.5.2\wtklib\devices[1].
- Renaming the folder with *Amharic* and placing it back in the same directory.
- Renaming the properties[2] file to *Amharic.properties* to make it match the folder name.
- Changing the font in the properties file in order to display Amharic characters as shown in Figure 5.1.
- Copying *ktools.properties* file from C:\WTK2.5.2\wtklib\Windows to C:\Documents and Settings\user\j2mewtk\2.5.2\wtklib and appending the text shown in Figure 5.2.

[1] It is used to store different emulator skins.
[2] Emulator skins are defined in properties file.

```
font.default=Power Geez Unicode1-plain-10
font.softButton=Power Geez Unicode1-plain-11

font.system.plain.small: Power Geez Unicode1-plain-9
font.system.plain.medium: Power Geez Unicode1-plain-11
font.system.plain.large: Power Geez Unicode1-plain-14

font.system.bold.small: Power Geez Unicode1-bold-9
font.system.bold.medium: Power Geez Unicode1-bold-11
font.system.bold.large: Power Geez Unicode1-bold-14

font.system.italic.small: Power Geez Unicode1-italic-9
font.system.italic.medium: Power Geez Unicode1-italic-11
font.system.italic.large: Power Geez Unicode1-italic-14

font.system.bold.italic.small: Power Geez Unicode1-bolditalic-9
font.system.bold.italic.medium: Power Geez Unicode1-bolditalic-11
font.system.bold.italic.large: Power Geez Unicode1-bolditalic-14

font.proportional.plain.small: Power Geez Unicode1-plain-9
font.proportional.plain.medium: Power Geez Unicode1-plain-11
font.proportional.plain.large: Power Geez Unicode1-plain-14

font.proportional.bold.small: Power Geez Unicode1-bold-9
font.proportional.bold.medium: Power Geez Unicode1-bold-11
font.proportional.bold.large: Power Geez Unicode1-bold-14

font.proportional.italic.small: Power Geez Unicode1-italic-9
font.proportional.italic.medium: Power Geez Unicode1-italic-11
font.proportional.italic.large: Power Geez Unicode1-italic-14

font.proportional.bold.italic.small: Power Geez Unicode1-bolditalic-9
font.proportional.bold.italic.medium: Power Geez Unicode1-bolditalic-11
font.proportional.bold.italic.large: Power Geez Unicode1-bolditalic-14
```

Figure 5.1: Amharic phone properties

```
wtk.locale = am-ET
microedition.locale = am-ET
microedition.encoding = UTF-8
microedition.encoding.supported: UTF-16, UTF-16BE, UTF-8, ISO-8859-1,
                                 ISO-8859-2, UCS-2
javac.encoding = UTF-8
```

Figure 5.2: ktools properties

After customizing the emulator, a project named ASSMP and a MIDlet[1] class named Synthesizer are created using the Sun Java™ Wireless Toolkit for CLDC 2.5.2. When this is done, a folder

[1] MIDP applications are called MIDlets, a continuation of the naming theme begun by applets and servlets.

named ASSMP will be created under C:\Documents and Settings\user\j2mewtk\2.5.2\apps containing other four folders named: *bin*, *lib*, *res (resource)*, and *src (source)*. The source folder stores the source codes of the classes.

A MIDlet Synthesizer is a class that starts automatically when the MIDP application is started. It calls other classes needed in order to generate the synthesized utterance. It also contains user interface objects such as ListBoxes, TextBoxes, Command buttons that enable us navigate through the menu items on the mobile phone. The captions displayed in these objects are Amharic characters with Unicode encoding.

After placing all the necessary classes implementing the algorithms in the source folder, the next step is choosing the emulator skin (in this case Amharic) and setting the locale to am-ET[1]. Then the project is built and run. When this is done, the emulator, shown in Figure 5.3 with the project name displayed onto the screen, is displayed. When someone chooses the Launch button and then again chooses the menu button, text box for entering text into the synthesizer will be displayed as shown on the left screen of Figure 5.4. The right screen shows the screen that is displayed when the **ማውጫ** "menu" command on the left screen is selected.

[1] am-ET – stands for Amharic Ethiopia

Figure 5.3: Screenshot of the ASSMP

Figure 5.4: Screenshots of ASSMP with text box for text entry and an open menu (***ማውጫ***)

As it is mentioned earlier, the text that is going to be synthesized is entered into the text box labeled with **ጽሑፍ አስገባ** "enter text". The user enters Amharic text in a transliterated form and chooses the **አጫውት** "play" command in order to listen to the synthesized speech.

The text is then passed to a class implementing the text analysis component of the NLP module described in the previous chapter. The text analysis module converts the text into its orthographic representation. As discussed in the previous chapter, the text that is going to be synthesized can be Amharic name or phone number.

The class implementing the phonetic analysis component of the NLP module accepts a text in orthographic form and converts the text into array of phonemes. The array of phonemes is then passed to the class implementing the unit selector component of the DSP module.

5.4. The DSP Module

The DSP module accepts the array of phonemes from the class implementing the phonetic analysis component of the NLP module. The class implementing the unit selector component converts the array of phonemes to an array of diphones. For each element (diphone) in the array, the diphone sound is fetched from the database and passed to the class implementing the concatenator component. The concatenator joins together the diphones and creates a wave file that will be uttered by the mobile phone.

Before the concatenation takes place, the diphone database should be prepared and stored in the resource folder of the ASSMP project. The following is a sequence of steps taken to prepare the diphone database:

- Optimal data selection
- Recording speech
- Segmenting diphones

The following subsections describe each of the above steps briefly.

5.4.1. Optimal Data Selection

During optimal data selection, a minimum set of words that contain all the diphones are chosen. This is because of the reason that diphones chosen from these minimum set of words have got higher probability of being taken from the same word. This in turn helps us get synthesized speech of higher quality.

As stated in the previous chapter, 884 unique names were collected for the purpose of analysis. It is also stated that the number of unique diphones obtained from these names was 420. All the 420 diphones were obtained from 167 names out of the 884 names.

5.4.2. Recording Speech

After optimal data selection is done, the next step is recording all the 167 names. To this end, Microsoft ® Sound Recorder is used. A male voice with 22,050 KHz sampling frequency, 16 bit Mono, and PCM encoding is recorded and saved in RIFF format. The recording has taken place in a quiet room, although there were some noises from the computers. Figure 5.5 and Figure 5.6

show the waveform and spectrogram[1] of the name [mulugieta] recorded with the above specification, respectively.

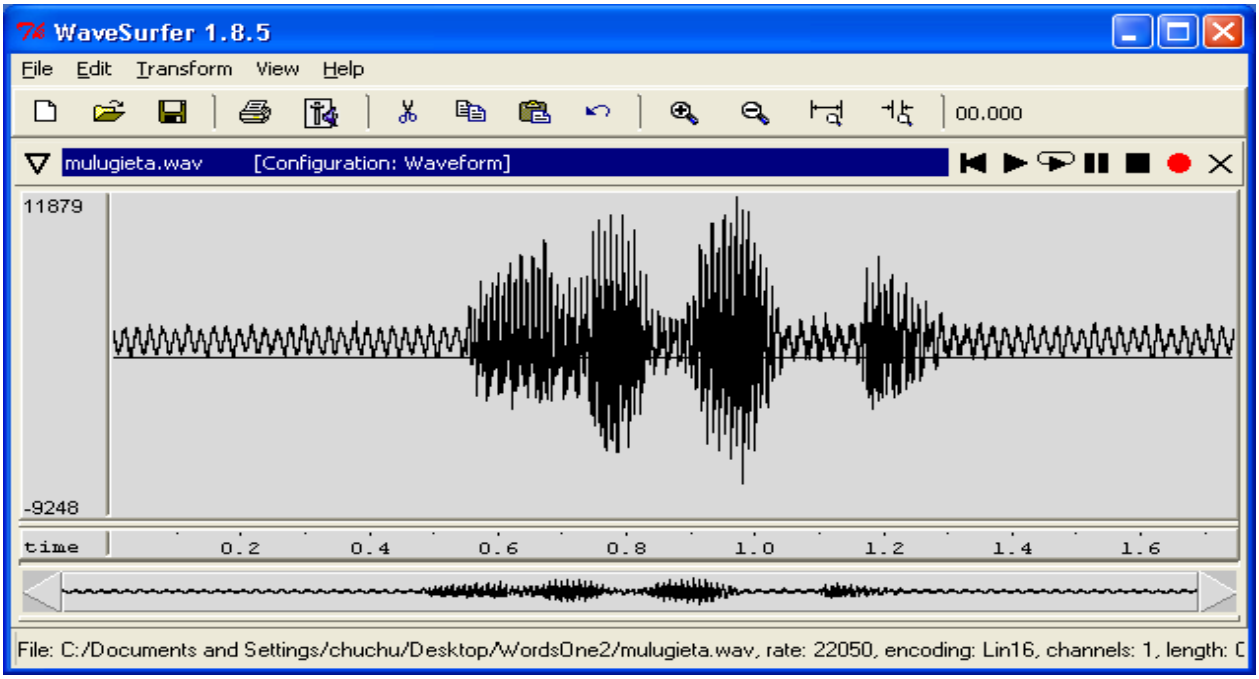

Figure 5.5: Waveform of the name **ሙሉጌታ** [mulugieta]

[1] Spectrogram is a commonly used method to describe a speech signal. It is a time-frequency-amplitude presentation of a speech signal. Higher amplitudes are presented with darker-grey levels [1].

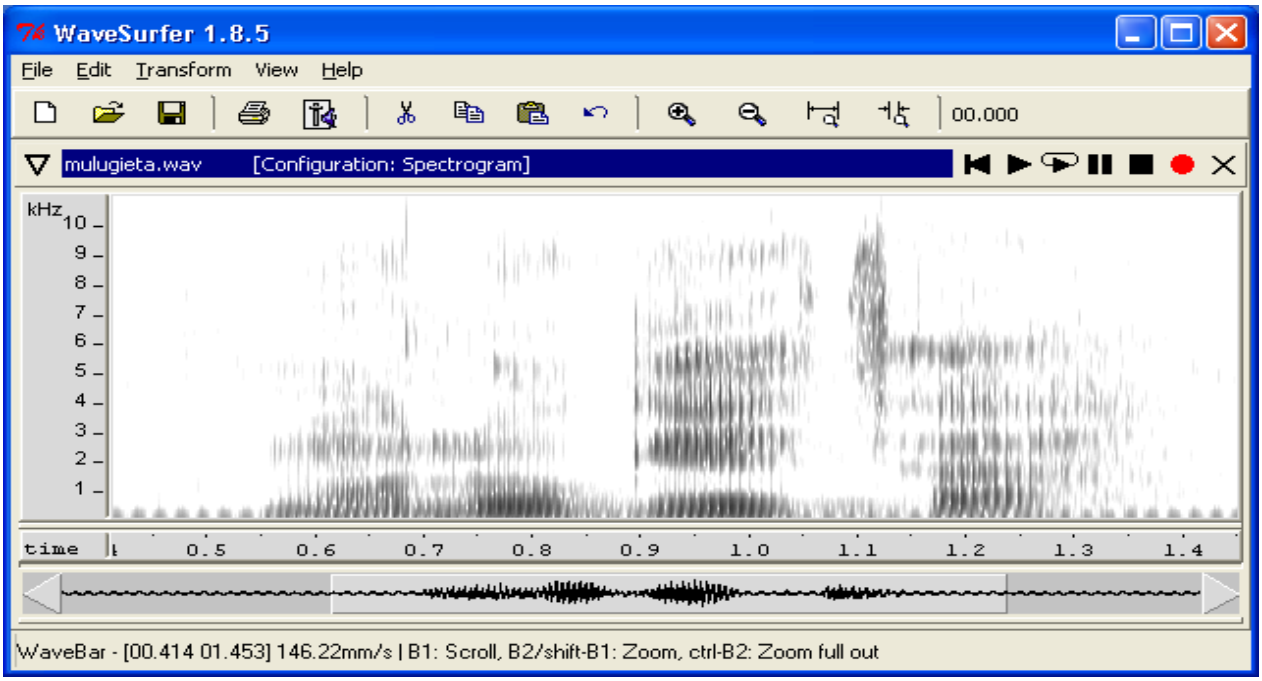

Figure 5.6: Spectrogram of the name **ሙ·ሉ·ጊ·ታ** [mulugieta]

5.4.3. Segmenting Diphones

Diphones were segmented manually using the WaveSurfer. Each diphone segment is stored in the resource folder of ASSMP project. Figure 5.7 shows the waveform of the name [mulugieta] labeled with the seven diphones which the name is comprised of: m-u, u-l, l-u, u-g, g-ie, ie-t, and t-a.

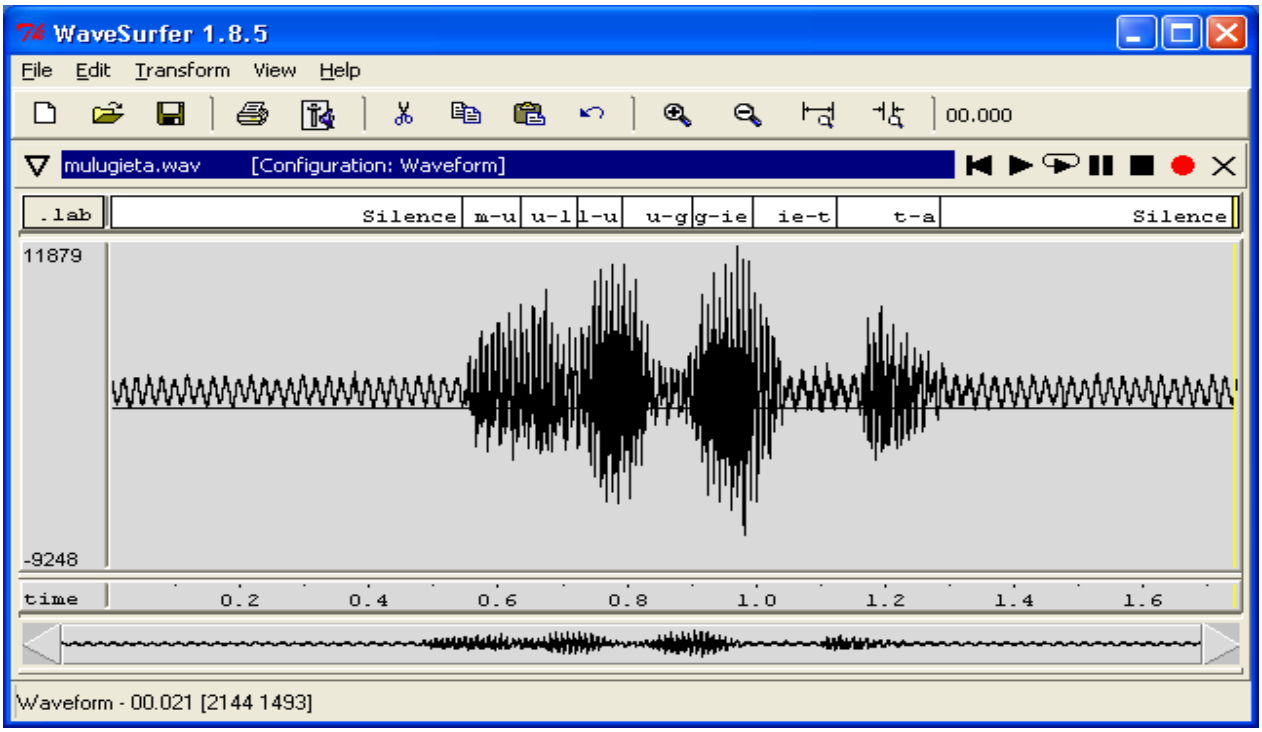

Figure 5.7: Diphones of the name መሉጌታ [mulugieta]

5.5. Experimentation

Synthetic speech can be compared and evaluated with respect to intelligibility, naturalness, and suitability for used application. In some applications, for example reading machines for the blind, the speech intelligibility is usually more important feature than the naturalness. On the other hand, prosodic features and naturalness are essential when we are dealing with multimedia applications or email readers. The evaluation can also be made at several levels, such as phoneme, word, or sentence level, depending on what kind of information is needed. The evaluation process is usually done by subjective listening tests. There are also some objective tests developed to evaluate speech quality [1]. According to [21], testing in TTS is not a simple or widely agreed on area.

5.5.1. Goal

The goal of evaluating the speech synthesizer developed is to evaluate the quality of speech in terms of intelligibility and naturalness. As stated in chapter one, intelligibility refers to how much the speech conveys its message. While naturalness refers to how “human” the synthetic speech sounds.

5.5.2. Preparing a Questionnaire

A questionnaire is prepared in order to achieve the aforementioned goal – evaluating intelligibility and naturalness. In doing so, questions that aim at evaluating the intelligibility and the naturalness of the synthetic speech are prepared. Each respondent is asked to give his/her opinion for both intelligibility and naturalness based on the following scales: bad, poor, fair, good, and excellent.

The questions are ordered in such a way that question evaluating intelligibility is made to come first followed by question evaluating naturalness. The reason for this is intelligibility measures how understandable the speech is, without giving due attention to the naturalness. After the speech is understood, the respondent then will be asked to evaluate the naturalness. The questionnaire is attached in Appendix A.

5.5.3. Method

A method called MOS (Mean Opinion Score) is used to measure the overall intelligibility and naturalness of the synthetic speech. Each scale mentioned above is given a value as follows: 1 = bad, 2 = poor, 3 = fair, 4 = good and 5 = excellent. 30 respondents are asked to rate 22 words on a

scale of 1 to 5. The scores are averaged resulting in an overall MOS rating for the synthesizer. The synthesizer is rated as: excellent if it has an MOS value greater than or equal to 4.5, good if it has an MOS value greater than or equal to 3.5 and less than 4.5, fair if it has an MOS value greater than or equal to 2.5 and less than 3.5, poor if it has an MOS value greater than or equal to 1.5 and less than 2.5, or bad otherwise.

5.5.4. Procedure

22 words are selected for testing. The selected words were among the 884 Amharic names that contained all phonemes of the Amharic language and that are not used for training (167 words were used for training). We believed that asking a lot of questions makes the respondents get bored. The 30 volunteer respondents are asked to rate (based on the scale described in the previous section) first the intelligibility and then the naturalness of each of the 22 words. The respondents were not allowed to look at the text because it may affect the evaluation.

5.5.5. Experimental Results and Discussions

The results obtained for intelligibility and naturalness are shown in Table 5.1 and Table 5.2, respectively. The overall MOS value for intelligibility is 4.10 and that of naturalness is 3.69. Based on the values of the MOS for intelligibility and naturalness, our speech synthesizer for mobile phone is good in both intelligibility and naturalness.

Table 5.1: Results for intelligibility by total average

Rank	Excellent	Good	Fair	Poor	Bad
Percentage of words	45.45%	30.45%	15.45%	5.76%	2.88%

Table 5.2: Results for naturalness by total average

Rank	Excellent	Good	Fair	Poor	Bad
Percentage of words	26.67%	34.09%	23.33%	13.79%	2.12%

The reason for having a speech synthesizer of such a good quality, as mentioned in the second chapter, is the use of diphones as a unit for concatenation. Since diphones are taken from relatively stable region of a speech signal, they are able to hold transition information which makes the synthetic voice sound good. Table 5.1 and Table 5.2 show that there are a few words that are categorized as poor and bad. This is because some diphones are hard to segment and also some discontinuities are observed when using diphones extracted from different words.

5.6. Summary

In this chapter, an attempt is made to describe how the speech synthesizer is implemented and experimented. J2ME is used for the implementation. Optimal data selection has been done prior to collecting acoustic data (by recording speech). Diphones are extracted from the recorded speech using the WaveSurfer tool.

After the system is implemented, subjective testing is conducted by asking the respondents to rate the intelligibility and naturalness of selected words. The overall evaluation is done using MOS method. The results show that, the speech synthesizer developed is categorized as good in terms of intelligibility and naturalness.

CHAPTER SIX

CONCLUSION AND RECOMMENDATIONS

6.1. Conclusion

Concatenative speech synthesis technique produces synthetic speech by joining together prerecorded speech segments. There are different speech units that can be used as a unit for concatenation. These include: phonemes (context-independent and context-dependent), diphones, syllables, words, or even sentences. The choice of a unit and also concatenation technique has an impact on the size of the database used to store the speech units, the time required to produce the waveform, and the quality of speech.

In this thesis work two concatenative synthesis techniques: unit selection synthesis and a concatenative synthesis technique that uses only one instance of each unit are compared in terms of storage space and processing time needed for concatenation and also the naturalness and intelligibility of the synthetic speech produced. In addition, the various Amharic units that can be used in a concatenative synthesis technique that stores one instance of a unit are also examined.

In a concatenative synthesis technique that stores only one unit, the most appealing unit in terms of size is context-independent phoneme. For Amharic language, we only need to store one instance of each of the 34 phonemes. Using context-independent phonemes results in audible discontinuities. Hence, including context information in the synthesis improves the quality of speech produced at the expense of storage space and processing time. Diphone is a unit that extends from the middle of the first phoneme to the middle of the second adjacent phoneme.

Since diphones are taken from a relatively stable region of speech signal, they offer a speech with good quality. Another unit to use is syllable. Coarticulation is well-handled within the syllables but not across the syllable. As a result, it is not advisable to use syllables as a unit for concatenation. Unit selection synthesis, even if it produces natural sounding speech, is not selected due to it needs database of larger size and its computational complexity. A concatenative speech synthesis technique that uses only one instance of diphone has been chosen and a speech synthesizer has been developed by concatenating diphones.

The speech synthesizer has two basic modules: the NLP module and the DSP module. The NLP module accepts raw text in a transliterated form and consists of two components: text analysis and phonetic analysis. The text analysis component is responsible for converting the raw text into orthographic representation. The output of the text analysis component is fed into the phonetic analysis component of the NLP module. The phonetic analysis component converts the text to a set of phonemes. The set of phonemes is given to the unit selector component of the DSP module.

The DSP module of a speech synthesizer is responsible for the generation of waveform. It contains two components: the unit selector component and the concatenator component. The unit selector component accepts a set of phonemes as an input and then converts it to a set of diphones. After the conversion has taken place, the unit selector component fetches each diphone (the wave file) from the database and gives the diphone to the concatenator component until all the diphones in the list are processed. After all the diphones have been processed, the concatenator component creates a wave file that can be played on mobile phones.

In preparing the diphone database optimal data selection has been performed first. The data used in this work is a set of Amharic names that can be saved onto mobile phones. To this end, 884 unique Amharic names were used. It is found that all the 420 diphones contained in these names were found in 167 names after performing optimal data selection. Then all of 420 diphones were segmented from the 167 names recorded in a WAVE format.

Evaluation is done to measure how intelligible and how natural the synthetic speech is, based on a method called MOS. According to the MOS results, the synthesizer is categorized as good in terms of both intelligibility and naturalness.

6.2. Recommendation

This research tried to show a speech synthesizer for Amharic language that can be deployed onto mobile phones by first selecting appropriate unit of concatenation, which is the first attempt ever done. Developing a full-fledged speech synthesizer for a mobile phone requires some more efforts to entertain all options in Amharic speech. In light of this we recommend the following:

- One can implement a component that accepts Amharic words, phrases, or sentences from the screen and transliterates it.
- There have been discontinuities observed which can be minimized by using signal smoothing techniques.
- A component that is responsible for generating prosody has not been included in the synthesizer. Incorporating a component for doing prosody generation can also be another task.

- The focus of this research was Amharic names and numbers. One can extend this to any arbitrary text composed of more than one word.

- This application can be combined with the work of [43] for reading aloud SMS messages.

References

[1] Lemmety S., “Review of Speech Synthesis Technology”, Master's Thesis, Helsinki University of Technology, 1999.

[2] Huang X., Acero A. and Hon, H., *Spoken Language Processing*, Prentice Hall, Upper Saddle River, New Jersey, 2001.

[3] Sebsibe H/Mariam, et. al, “Unit Selection Voice for Amharic using Festvox”, 5th ISCA Workshop on Speech Synthesis, Pittsburgh, USA, June 2004.

[4] Quazza S., et al., “ACTOR: a Multilingual Unit-Selection Speech Synthesis System”, 4th ISCA Workshop on Speech Synthesis, Perthshire, Scotland, September 2001.

[5] Black A. and Lenzo K., “Limited Domain Synthesis”, in Proc. ICSLP '00, Beijing, China, vol. 2, pp. 411-414, October 2000.

[6] Laine Berhane, “Text-to-Speech Synthesis of the Amharic Language”, Master’s Thesis, Addis Ababa University, 1998.

[7] Henock Lulseged, “Concatenative Text-To-Speech (TTS) Synthesis for the Amharic Language”, Master’s Thesis, Addis Ababa University, 2003.

[8] Habtamu Taye, “Diphone based Text-To-Speech Synthesis System for Amharic”, Master’s Project, Addis Ababa University, 2007.

[9] Nadew Tademe, “Formant based Speech Synthesis for Amharic Vowels”, Master’s Thesis, Addis Ababa University, 2008.

[10] Ethiopian Reporter, Newspaper, September 22, 2007, www.ethiopianreporter.com, Last Accessed on September 24, 2007.

[11] Kishore S. and Black A., “Unit Size in Unit Selection Speech Synthesis”, European Conference on Speech Communication and Technology, ISCA, 2003.

[12] http://festvox.org/bsv/, Last Accessed on February 28, 2008.

[13] Tucker R. and Shalonova K., “Supporting the Creation of TTS for Local Language Voice Information Systems”, Interspeech, Lisbon, Portugal, 2005.

[14] Deng Zhong-hua, “Review of Technique Body based on J2ME for Programming Mobile Phone Games”, *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on*, vol.2, no., pp. 1383-1386, 23-26 Sept. 2005.

[15] Talafova R., Rozinaj G. and Cepko J., “Speech Synthesis for Mobile Phone”, *ELMAR, 2007*, vol., no., pp.167-170, 12-14 September 2007.

[16] Donovan R., “Trainable Speech Synthesis”, Ph.D Dissertation, Cambridge University, June 1996.

[17] Assaf M., “A Prototype of an Arabic Diphone Speech Synthesizer in Festival”, Master’s Thesis, Uppsala Universitet, 2005.

[18] Sak H, “A Corpus-Based Concatenative Speech Synthesis System for Turkish”, Master’s. Thesis, Bogazici University, 2004.

[19] Klatt D., “Review of Text-to-Speech Conversion for English”, Journal of the Acoustical Society of America, JASA vol. 82, pp 6-15, 1987.

[20] Roach P., “A Little Encyclopaedia of Phonetics”, Available at http://www.linguistics.reading.ac.uk/staff/Peter.Roach/PAPERS/encyc2.pdf, Last Accessed on June 1, 2006.

[21] Taylor P., “Text-to-Speech Synthesis”, Available at http://svr-www.eng.cam.ac.uk/~pat40/ttsbook_draft_2.pdf, Last accessed on June 1, 2008.

[22] Varga I., et al., “ASR in Mobile Phones - An Industrial Approach”, *Speech and Audio Processing, IEEE Transactions on*, vol.10, no.8, pp. 562-569, November 2002.

[23] Clark R., Richmond K. and King S., “Festival 2: Build Your Own General Purpose Unit Selection Speech Synthesizer”, Proceedings of the 5th ISCA Speech Synthesis Workshop. Pittsburgh, PA: 173–178, 2004.

[24] Tesfaye Yihdego, “Diphone based Text-To-Speech Synthesis System for Tigrigna”, Master’s Thesis, Addis Ababa University, 2004.

[25] Morka Mekonnen, “Text-To-Speech System for Afan Orommo,” Master’s Thesis, Addis Ababa University, 2001.

[26] http://ccrma.stanford.edu/courses/422/projects/WaveFormat/, Last Accessed on June 6, 2008.

[27] http://www.codefactory.es/en/, Last Accessed on August 2, 2008.

[28] Rutten P., et al., “Issues in Corpus based Speech Synthesis”, *State of the Art in Speech Synthesis (Ref. No. 2000/058), IEE Seminar on*, vol., no., pp.16/1-16/7, 2000.

[29] http://www.arts.gla.ac.uk/ipa/ipachart.html, Last Accessed on July 25, 2008.

[30] Rouibia S., Rosec O. and Moudenc T., “Unit Selection for Speech Synthesis based on Acoustic Critera”, in Proc. Text Speech and Dialogue, pp. 281-287, Karlovy Vary, Czech Republic, September 2005.

[31] http://www.phon.ucl.ac.uk/home/sampa/, Last Accessed on July 25, 2008.

[32] Hussien Seid and Gamback B., “A Speaker Independent Continuous Speech Recognizer for Amharic”, In: INTERSPEECH 2005, 9th European Conference on Speech Communication and Technology, Lisbon, Portugal, September 2005.

[33] Solomon Tefera, “Automatic Speech Recognition for Amharic”, Ph.D Dissertation, University of Hamburg, 2005.

[34] Worku Alemu, “The Application of OCR Technique to the Amharic Script”, Master’s Thesis, Addis Ababa University, 1997.

[35] **ጌታሁን አማረ ፣ ዘመናዊ የአማርኛ ሰዋሰው በቀላል አቀራረብ ፣ ንግድ ማተሚያ ድርጅት ፡ 1995 እንደገና ታተመ።።**

[36] http://nlp.amharic.org/resources/lexical/word-lists/toponymic/, Last Accessed on April 14, 2008

[37] Hunt A. and Black A., “Unit Selection in a Concatenative Speech Synthesis System using a Large Speech Database”, *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol.1, no., pp.373-376 vol. 1, 7-10 May 1996.

[38] Black A. and Lenzo K., “Optimal Data Selection for Unit Selection Synthesis”, The 4th ISCA Workshop on Speech Synthesis, Perthshire, Scotland, UK, August 29 - September 1, 2001.

[39] Bulut M., Narayanan S. and Syrdal A., “Expressive Speech Synthesis using a Concatenative Synthesizer”, Proc. ICSLP, pp. 1265--1268, Denver, U.S.A., Sep. 2002.

[40] Black A., “Unit Selection and Emotional Speech”, In: Proc. EUROSPEECH-2003, Geneva, Switzerland, pp. 1649-1652, 2003.

[41] Keogh J., J2ME*: The Complete Reference*, Mc-GRAW-HILL, 2003.

[42] Knudsen J., *Wireless Java Developing with J2ME,* Second Edition, 2003.

[43] Shiferaw Abebe, “A Novel Java Wireless Application For Unicode-Based Multi-Script Simple Messaging Service (SMS)”, Master’s Thesis, Addis Ababa University, 2005.

[44] Jannedy S. and Mobius B., “Name Pronunciation in German Text-to-Speech Synthesis”, In Proc. 5th Conference on Applied Natural Language Processing, pp. 49-56, Washington DC, 1997.

[45] Spiegel F., "Proper Name Pronunciations for Speech Technology Applications," International Journal of Speech Technology, vol. 6, no.4, pp. 419-427, 2003.

[46] http://www.euromonitor.com/No_end_in_sight_for_Africas_mobile_telephone_boom, Last Accessed on October 01, 2008.

[47] http://www.cellular-news.com/story/31063.php, Last Accessed on October 01, 2008.

Appendix A: Questionnaire

Addis Ababa University
School of Graduate Studies

Users' Evaluation of Amharic Speech Synthesizer for Mobile Phone

The aim of this questionnaire is to evaluate the intelligibility and naturalness of Amharic Speech Synthesizer for Mobile Phone. All the information that you fill in this form is very critical to the conclusions we make at the end of the research work. So, I request you to answer for the questions freely and honestly.

Listen to the 22 Amharic names and answer the following questions.

1. How do you evaluate the understandability of the synthesized speech?

   - ☐ Excellent
   - ☐ Good
   - ☐ Fair
   - ☐ Poor
   - ☐ Bad

2. How do you evaluate the naturalness of the synthesized speech?

   - ☐ Excellent
   - ☐ Good
   - ☐ Fair
   - ☐ Poor
   - ☐ Bad

Declaration

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

WORKAGEGNEHU PETROS FITAMO

This thesis has been submitted for examination with my approval as an advisor.

SEBSIBE HAILEMARIAM

Addis Ababa, Ethiopia

October, 2008