HINDI TEXT TO SPEECH SYSTEM

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Shruti Gupta
801032025

Under the supervision of:
Mr. Parteek Bhatia
Assistant Professor, CSED

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2012

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "Hindi Text To Speech System", in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in *Computer Science and Engineering* Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Mr. Parteek Bhatia* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Shruti Gupta

(Shruti Gupta)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Parteek Bhatia

(Mr. Parteek Bhatia)
Computer Science and Engineering Department
Thapar University
Patiala

Countersigned By:

(Dr. Maninder Singh)
Head
6/2/12
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

Acknowledgement

First of all, I am thankful to God for his blessings and showing me the right direction. With His mercy, it has been made possible for me to reach so far.

It is a great privilege to express my gratitude and admiration toward my respected supervisor **Mr. Parteek Kumar**. He has been an esteemed guide and a great support behind achieving the task. Without his able guidance, kind efforts and encouragement, the work would not have been what it is. I am truly grateful to him for extending his total co-operation and understanding whenever I needed help and guidance from him.

I wish to express my heartiest thanks to Dr. Maninder Singh, Head, Computer Science and Engineering Department, Thapar University, Patiala for providing me the opportunity and all necessary facilities to accomplish this thesis successfully.

I am grateful to my parents who soulfully provided me their constant support and encouraging attitude to undertake the challenge of this proportion. They believed in before I believed in myself. To them I owe my wonderful today and dream filled future.

I would also like to say thanks to my friends who were always there at the need of hour and provided help for the completion of my thesis work.

Shruti Gupta

(801032025)

Abstract

A Text to Speech (TTS) Synthesizer is a computer application that is capable of reading out typed text. This generally involves two steps, text processing and speech generation. Speech synthesizers can be characterized by the size of the speech units they concatenate, as well as by the method used to code, store and synthesize the speech. Synthetic speech may be used in several applications like telecommunications services, language education, aid to handicapped persons, fundamental and applied research etc. TTS has to face many challenges during the process of conversion of text to speech. The most important qualities expected from speech synthesis system are naturalness and intelligibility.

In India different languages are spoken, each language being the mother tongue of tens of millions of people. While the languages and scripts are distinct from each other, the grammar and the alphabet are similar to a large extent. One common feature is that all the Indian languages are phonetic in nature.

In this thesis the general architecture of the TTS along with the methods of text to phoneme conversion and various ways of speech synthesis have been discussed. A brief introduction about the various TTS systems that are already developed for Indian languages namely Dhvani-Text-to-speech system, Shruti-an embedded text to speech system, HP Lab System and Vani are given. These systems are also compared on the basis of the methods used by them for speech synthesis.

A graphical user interface has been designed for converting Hindi text to speech in Java Swings. This system works at Linux platform. In this system Dhvani is used as a back end tool. Also detailed analysis of the various intermediate stages has been done which an input text passes through before converting into final speech. After analysing various features of ***Dhvani***, some of its limitations like it works only in Linux environment and has no female voice database has come across. To deal with these limitations a new Hindi text to Speech module has also been proposed.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Speech and spoken words have always played a big role in the individual and collective lives of the people. Wars have been won, peace agreements have been made because of the magical words of a few who knew how to give life to their words. Speech represents the spoken form of a language and is also one of the important means of communication. Over the past few decades, many researches have been done in the field of converting text to speech. This research has resulted in important advances with many systems being able to generate a close to a real natural sound. These advances in speech synthesis also pave the way for many speech related new applications.

1.1 Introduction to Text to Speech System

The function of text-to-speech (TTS) system is to convert an arbitrary text to a spoken waveform. This generally involves two steps, *i.e.*, text processing and speech generation. Text processing is used to convert the given text to a sequence of synthesis units while speech generation is generation of an acoustic wave form corresponding to each of these units in the sequence [17] [15].

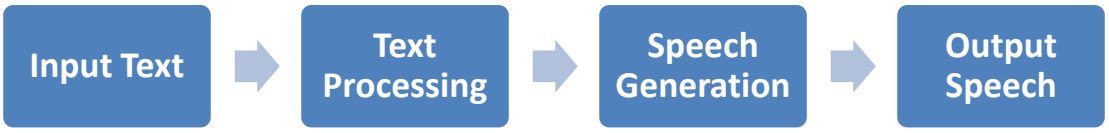


Figure 1.1: Text to Speech System

1.2 Applications of Text-to-Speech System

The application field of TTS is expanding fast whilst the quality of TTS systems is also increasing steadily. Speech synthesis systems are also becoming more affordable for common customers, which makes these systems more suitable for everyday use. Some uses of TTS are described below.

- **Aid to Vocally Handicapped**

  A hand-held, battery-powered synthetic speech aid can be used by vocally handicapped person to express their words. The device will have especially designed keyboard, which accepts the input, and converts into the required speech within blink of eyes.

- **Source of Learning for Visually Impaired**

  Listening is an important skill for people who are blind. Blind individuals rely on their ability to hear or listen to gain information quickly and efficiently. Students use their sense of hearing to gain information from books on tape or CD, but also to assess what is happening around them.

- **Talking Books and Toys**

  Talking book not only teaches how to read but also has more impact on students than text reading. It makes their study more enjoyable and easy. In the same way talking toys are a great source of fun and entertainment for children.

- **Games and Education**.

  Synthesized speech can also be used in many educational institutions in field of study as well as sports. A teacher can be tired at a point of time but a computer with speech synthesizer can teach whole day with same efficiency and accuracy.

- **Telecommunication and Multimedia.**

  TTS systems make it possible to access textual information over the telephone. Texts can be large databases which can hardly be read and stored as digitized speech. Queries to such information retrieval systems could be put through the user's voice (with the help of a speech recognizer), or through the telephone keyboard. Synthesized speech may also be used to speak out short text messages in mobile phones.

- **Man-Machine Communication.**

  Speech synthesis may be used in several kinds of human-machine interactions. For example, in warning, alarm systems, clocks and washing machines synthesized

speech may be used to give more accurate information of the current situation. Speech signals are far better than that of warning lights or buzzers as it enables to react to the signal more fast if the person is unable to get light due some obstacles.

- **Voice Enabled E-mail.**

  Voice-enabled e-mail uses voice recognition and speech synthesis technologies to enable users to access their e-mail from any telephone. The subscriber dials a phone number to access a voice portal, then, to collect their e-mail messages, they press a couple of keys and, perhaps, say a phrase like "Get my e-mail." Speech synthesis software converts e-mail text to a voice message, which is played back over the phone. Voice-enabled e-mail is especially useful for mobile workers, because it makes it possible for them to access their messages easily from virtually anywhere (as long as they can get to a phone), without having to invest in expensive equipment such as laptop computers or personal digital assistants (PDAs) [6].

1.3 Challenges in Text-to-Speech System

Speech synthesis has been developed steadily over the recent decades and it has been integrated into several new applications. Developing speech synthesis system is a complicated process and, it includes the following important challenges.

- Development of TTS systems require knowledge about human speech production and about languages being developed.
- The actual implementation of a fully functional system requires good software skills.
- Most TTS systems do not generate semantic representations of their input text, as a result, various heuristic techniques are used to guess the proper way to disambiguate homographs [18], like examining neighbouring words and using statistics about frequency of occurrence.
- The most important qualities of a speech synthesis system are naturalness and intelligibility [25]. Naturalness describes how closely the output sounds like human speech, while intelligibility is the ease with which the output is understood. The ideal speech synthesizer should be both natural and intelligible [26].

1.4 Overview of Work

Speech synthesis is a process of converting an input text into speech waveforms. This technology is a great help to people with physical impairments like visually handicapped and vocally disabled. The ability to convert text to voice reduces the dependency, frustration, and sense of helplessness of these people.
In this thesis four Indian languages text-to-speech systems, namely ***Dhvani***, ***Shruti, HP Lab System*** based on Festival framework and ***Vani*** system are discussed. These systems are compared on the basis of languages it support, size of the speech units, methodologies used to code, store and synthesize the speech and prosody.

Many Indian languages TTS has been developed, but they don't have a good user friendly interface. This thesis discusses about the development of a user friendly Interface for converting Hindi text to speech using ***Dhvani*** as a back end tool which runs on Linux. The intermediate stages in converting the text to speech are also analysed. A new Hindi text to speech module is also proposed which covers some of the shortcomings of ***Dhvani***.

Chapter 2

Literature Survey

2.1 Indian Languages

Language is not only a rule-governed system of communication but also a phenomenon that to a great extent structures our thought and defines our social relationships in terms of both power and equality. India is rich in languages. There are a quite a number of languages spoken in India. Some of these languages are accepted nationally while others are accepted as dialects of that particular region.

The Indian languages belong to four language families namely *Indo-European, Dravidian, Austro-Asiatic (Austric) and Sino-Tibetan.* Majority of India's population are using Indo-European and Dravidian languages. The former are spoken mainly in northern and central regions and the latter in southern India. Some ethnic groups in Assam and other parts of eastern India speak Austric languages. People in the northern Himalayan region and near the Burmese border speak Sino-Tibetan languages. The written forms of language or scripts come from an ancient Indian script called Brahmi [20].

India has 22 officially recognised languages. But around, 33 different languages and 2000 dialects have been identified in India. Hindi, in the Devanagari script is the official language of the Federal government of India. English is an associate official language. Sanskrit, the classical language of India, represents the highest achievement of the Indo-Aryan Languages. Some features of Indian Languages are given below.

- According to Eighth Schedule, twenty two Scheduled Languages are - Assamese, Bengali, Gujarati, Hindi, Kannada, Kashmiri, Konkani, Malayalam, Manipuri, Marathi, Nepali, Oriya, Punjabi, Sanskrit, Sindhi, Tamil, Telugu, Urdu, Bodo, Dogri, Maithili and Santhali [1].
- Hindi is spoken by largest population of India followed by Bengali, Telugu, Marathi and others.
- Scripts used in Indian languages are phonetic in nature.

- All languages have systematic manner of production.

Table 2.1 represents some of the commonly spoken languages in various parts of India along with the percentage of people who use them and its state name where it is official language.

Table 2.1 Commonly Spoken Languages in India

Language	Percentage of People (In India)	State (Official Language)
Assamese	1.28%	Assam
Bengali	8.11%	Tripura, West Bengal
Dogri	0.22%	Jammu and Kashmir
Gujarati	4.48%	Dadra and Nagar Haveli, Daman and Diu, Gujrat
Hindi	41.03%	Arunachal Pradesh, Andaman and Nicobar Islands, Bihar, Chandigarh, Chhattisgarh, Delhi, Haryana, Himachal Pradesh, Jharkhand, Madhya Pradesh, Rajasthan, Uttar Pradesh and Uttaranchal
Kannada	3.69%	Karanataka
Kashmiri	0.54%	Jammu and Kashmir
Konkani	0.24%	Goa
Maithli	1.18%	Bihar
Malayalam	3.21%	Kerala and Lakshadweep
Manipuri	0.14%	Manipur
Marathi	6.99%	Maharashtra
Oriya	3.21%	Orissa
Punjabi	2.83%	Punjab
Tamil	5.91%	Tamil Nadu , Pondicherry
Telgu	7.19%	Andhra Pradesh
Urdu	5.01%	Jammu and Kashmir

By referring Table 2.1, we concluded that Hindi is the most popular language spoken in India and hence a user-friendly interface for converting Hindi text to speech is of great use in India.

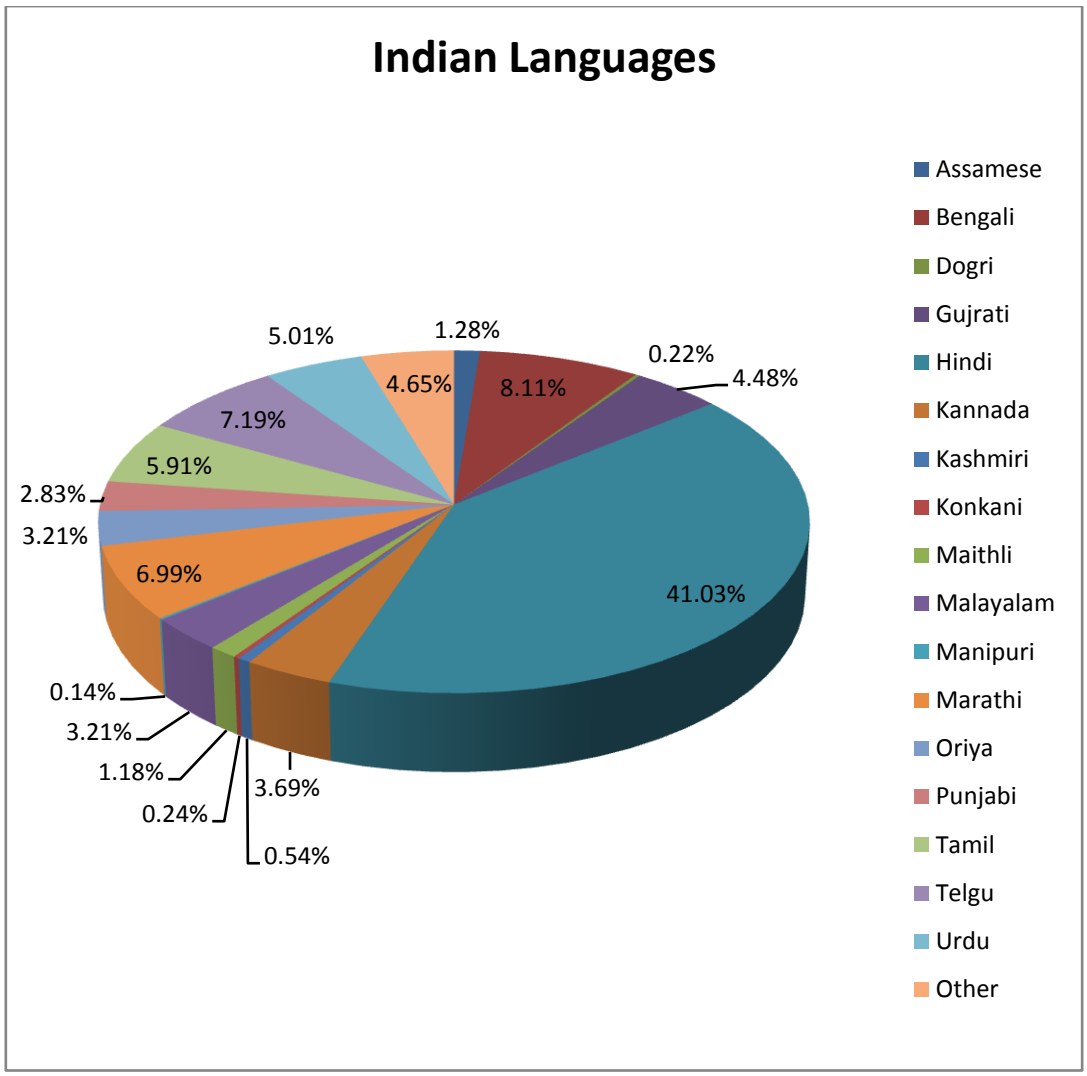


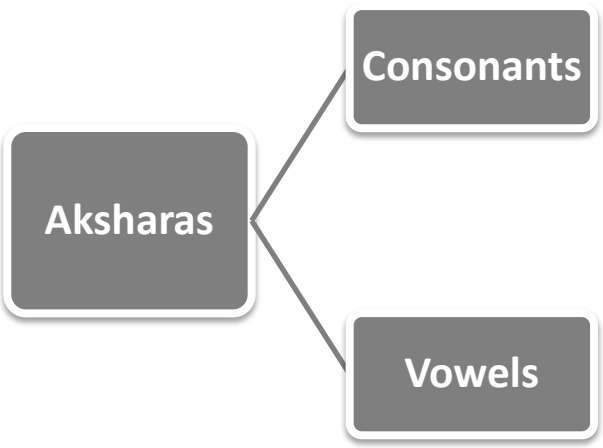
Figure 2.1: Percentage of Various Indian Languages Uses in India

2.1.1 Brahmi Script

Brahmi script is an ancient script from which Indian Languages have been originated. The basic units of the writing system are referred to as *Aksharas*. Aksharas are divided into ***Consonants(C)*** and ***vowels (V)*** as shown in figure 2.2.

The properties of Aksharas are as follows:

- An Akshara is an orthographic representation of a speech sound in an Indian language.
- Aksharas are syllabic in nature.
- The typical forms of Akshara are **V, CV, CCV** and **CCCV**, thus have a generalized form of **C*V**.



*A **consonant** is a sound in spoken language that is characterized by a constriction or closure at one or more points along the vocal tract such as the lips, tongue and teeth.*

*A **vowel** is a sound in spoken language, pronounced with an open vocal tract so that there is no build-up of air pressure at*

Figure 2.2: Components of Akshara

Consonants

Eden Golshani [20] has divided the consonants into Gutturals, Palatals, Cerebrals, Dentals, Labials, Semi-Vowels, Sibliants/Aspirants, Miscallaneous and Compound as shown in figure 2.3.

Gutturals	Hindi	Punjabi	Bengali	Gujarati
K	क	ਕ	ক	ક
Kh	ख	ਖ	খ	ખ
G	ग	ਗ	গ	ગ
Gh	घ	ਘ	ঘ	ઘ
N'	ङ	ਙ	ঙ(ng)	ઙ

(a) Gutturals

Palatals	Hindi	Punjabi	Bengali	Gujarati
Ch	च	ਚ	চ	ચ
Chh	छ	ਛ	ছ	છ
J	ज	ਜ	জ	જ
Jh	झ/झ	ਝ	ঝ	ઝ
N'	ञ	ਞ	ঞ	ઞ

(b) Palatals

Figure 2.3: Consonants Division

Cerebrals	Hindi	Punjabi	Bengali	Gujarati
T	ट	ਟ	ট	ટ
Th	ठ	ਠ	ঠ	ઠ
D	ड	ਡ	ড	ડ
Dh	ढ	ਢ	ঢ	ઢ
N'	ण	ਣ	ণ	ણ

(c) Cerebrals

Dentals	Hindi	Punjabi	Bengali	Gujarati
T'	त	ਤ	ত	ત
T'h	थ	ਥ	থ	થ
Th	द	ਦ	দ	દ
Thh	ध	ਧ	ধ	ધ
N	न	ਨ	ন	ન

(d) Dentals

Semivowels	Hindi	Punjabi	Bengali	Gujarati
Y	य	ਯ	য়	ય
R	र	ਰ	র	ર
L	ल	ਲ	ল	લ
V	व	ਵ	–	વ
Z	ज़	ਜ਼	য	–

(e) Semi-Vowels

Sibliants/ Aspirant	Hindi	Punjabi	Bengali	Gujarati
Sh	श	ਸ਼	শ	શ
Shh	ष	–	ষ	ષ
S	स	ਸ	স	સ
H	ह	ਹ	হ	હ

(f) Sibliants/Aspirant

Misc.	Hindi	Punjabi	Bengali	Gujarati
ughDh	ड़	ੜ	ড়	–
ughDhh	ढ़	–	ঢ়	–
Ru'	–	–	–	ૃ
L'	–	–	–	ૢ

(g) Miscallaneous

Compound	Hindi	Punjabi	Bengali	Gujarati
Ksh	क्ष	–	–	ક્ષ
T'r	त्र	–	–	ત્ર
Gy	ज्ञ	–	–	જ્ઞ
Shr	श्र	–	–	શ્ર
Ri	ऋ मृ	–	–	ઋ મૃ

(h) Compound

Figure 2.3: Consonants Division

Vowels

Vowels are most interesting class of sound in any language as shown in figure 2.4. Their duration in any word is also of most significance. They play a major role in the pronunciation of any word. Vowels are always voiced sounds and they are produced

with the vocal cords in vibration, while consonants may be either voiced or unvoiced.

.

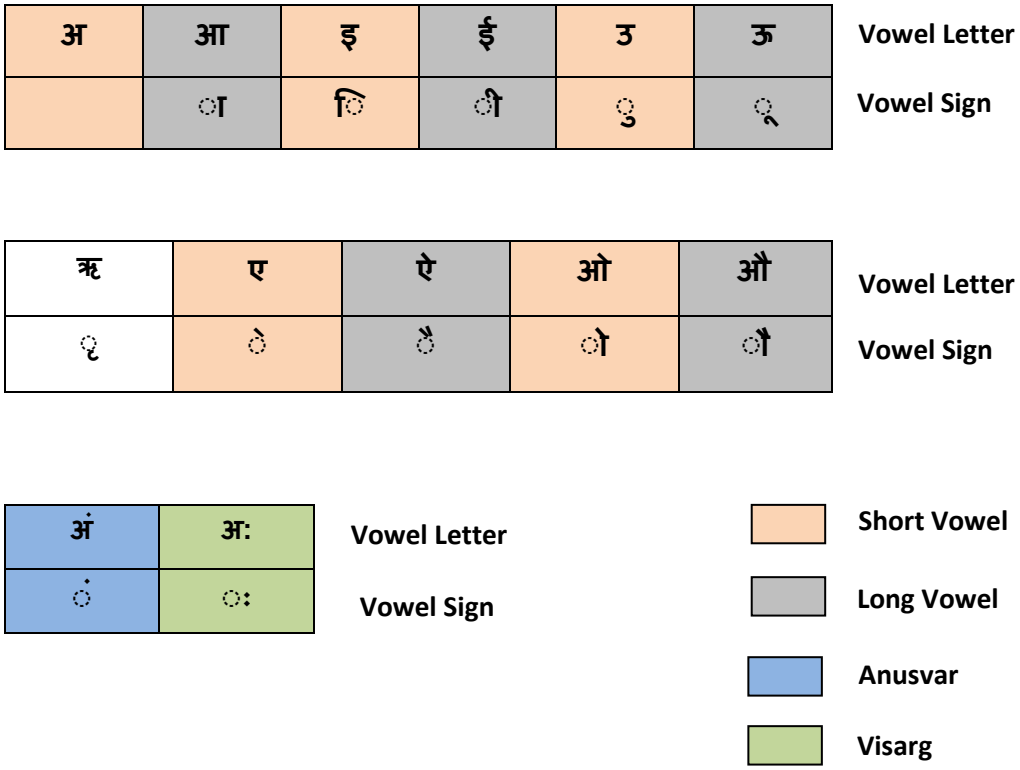


Figure 2.4: Types of Vowels

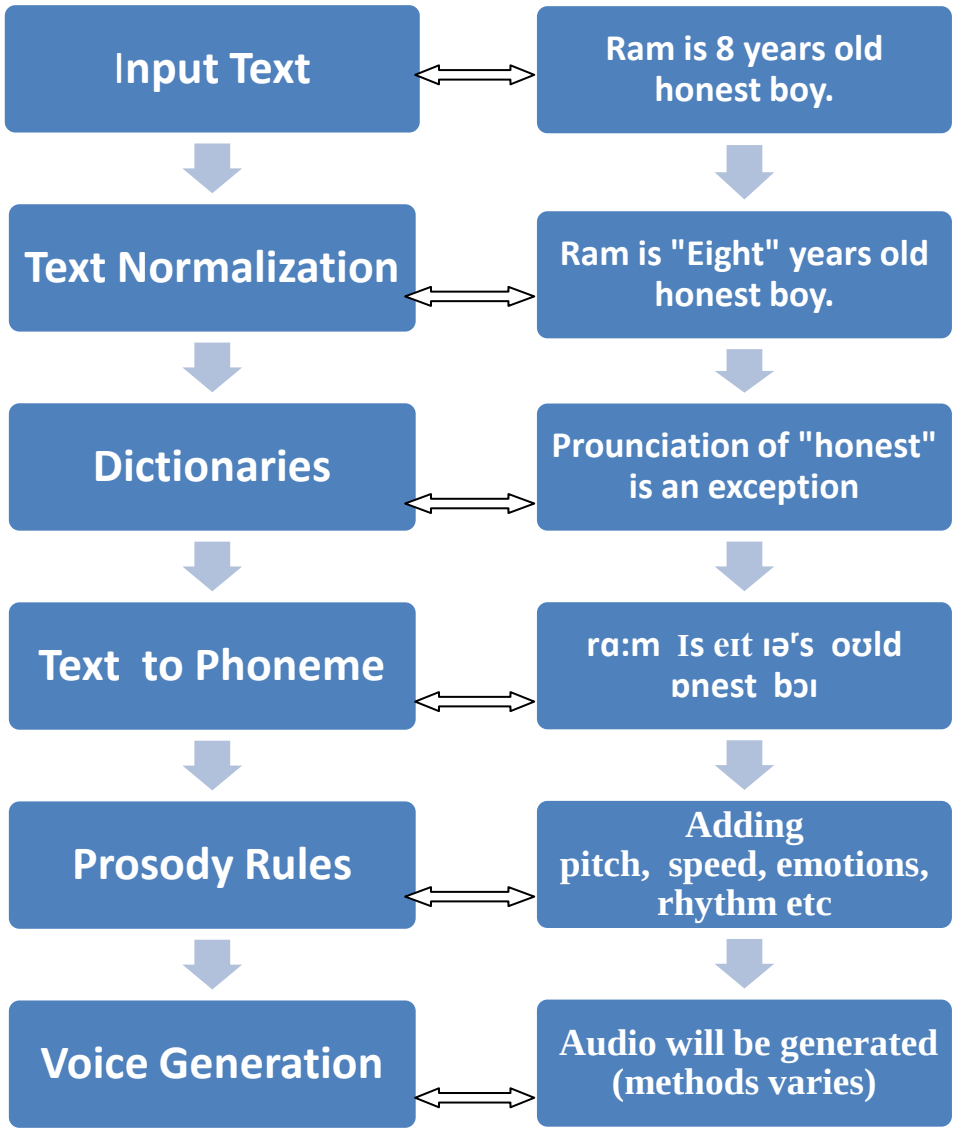
Vowels when used with the consonants in various languages like Hindi, Punjabi, Bengali and Gujarati is shown in figure 2.5.

Vowels	Hindi	Punjabi	Bengali	Gujarati
A	अ म	ਅ ਮ	অ ম	અ મ
Aa	आ मा	ਆ ਮਾ	আ মা	આ મા
I	इ मि	ਇ ਮਿ	ই মি	ઇ મિ
Ee	ई मी	ਈ ਮੀ	ঈ মী	ઈ મી
U	उ मु	ਉ ਮੁ	উ মু	ઉ મુ
Oo	ऊ मू	ਊ ਮੂ	ঊ মূ	ઊ મૂ
E	ए मे	ਏ ਮੇ	এ মে	એ મે
Ey	ऐ मै	ਐ ਮੈ	ঐ মৈ	ઐ મૈ
O	ओ मो	ਓ ਮੋ	ও মো	ઓ મો
Ou	औ मौ	ਔ ਮੌ	ঔ মৌ	ઔ મૌ
Un'	अं मं	ਅਂ ਮਂ	—	અં મં
An'	अँ मँ	ਅੰ ਮੰ	অঁ মঁ	અઁ મઁ

Figure 2.5: Vowels with Consonants

2.2 General Architecture of Text to Speech System

The conversion of text to speech is not a single step process. Many things need to be considered while conversion of text to speech. The process of converting text into speech breaks down into a number of stages. The general architecture of converting text to speech is shown in figure 2.6.



2.6: Architecture of Text-to-Speech System

O'Malley [23] in 1990 has described the process of converting text into speech in a number of stages.

- Text normalization
- Dictionaries
- Text to phoneme
- Voice generation
- Prosody rules

These stages are discussed in detail in the further sections.

2.2.1 Text Normalization

Text normalization converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. It isolates words in the text. Text normalization then searches for numbers, times, dates, and other symbolic representations. These are analysed and converted to words.

2.2.2 Dictionaries

Some words in languages are not pronounced in accordance with the basic rules of pronunciation. To say these words correctly, the system stores a phonemic transcription of their exact pronunciation in an exception dictionary. The dictionary is also needed to store grammatical information about particular words.

2.2.3 Text to Phoneme

Text to phoneme conversion is to derive the pronunciation for each word based on its spelling. Word pronunciations may be conveniently described as sequences of phonemes, which are units of sound in a language that serve to distinguish one word from another. There are mainly two approaches for text to phoneme conversion.

- Dictionary based approach
- Rule based approach

These approaches are discussed below.

Dictionary-based Approach

In this approach, a large dictionary containing all the words of a language and their correct pronunciations is stored in the database. To determine the correct

pronunciation of each word, it is looked up in the dictionary and the spelling is replaced with the pronunciation specified in the dictionary. This approach is quick and accurate, but it fails if the looked up word is not present in the dictionary. Also, the size of dictionary should be considered carefully, as the memory space requirements of the synthesis system depends on the size.

The advantage of dictionary-based approach is that it is quick and accurate. On the other hand its disadvantages are that this approach completely fails if it is given a word which is not in its dictionary and as its dictionary size grows, so too does the memory space requirements of the synthesis system.

Rule-based Approach

In this approach pronunciation rules are applied to words to determine their pronunciations based on their spellings.The advantage of rule-based approach is that is good for languages with a phonemic orthography having a very regular writing system, and the prediction of the pronunciation of words based on their spellings is quite successful. In contrast, this approach works on any input, but the complexity of the rules grows substantially as the system takes into account irregular spellings or pronunciations.

2.2.4 Prosody Rules

To find correct intonation, stress, and duration from written text is probably the most challenging problem. The intonation implies how the pitch pattern or fundamental frequency changes during speech. These features together are called prosodic features and considered as the melody, rhythm, and emphasis of the speech at the perceptual level [30]. There are many separate aspects on which the prosody of continuous speech depends, like meaning of the sentence and the speaker characteristics and emotions.

2.2.5 Voice Generation

At this stage, speech synthesis has already been performed. Now the voice is generated on the basis of list of phonemes and their duration, pitch, and volume. Methods for generating the digital audio will vary. In a summarized form, the engine receives the phoneme to speak, loads the digital audio from a database, does some pitch, time, and volume changes, and sends it out to the sound card.

2.3 Methods of Speech Synthesis

Synthesized speech can be produced by several different methods. All of these have some benefits and deficiencies. The methods are usually classified into three groups [21].

- Articulatory synthesis, which attempts to model the human speech production system directly.
- Formant synthesis, which is done by exciting a set of resonators by a voicing source or noise generator to achieve the desired speech spectrum.
- Concatenative synthesis, which uses different length pre-recorded samples derived from natural speech.

The formant and concatenative methods are the most commonly used in present synthesis systems. The articulatory method is still too complicated for high quality implementations, but may arise as a potential method in the future [28]. These three methods namely articulatory synthesis, formant synthesis and concatenative synthesis are discussed in further sections.

2.3.1 Articulatory Synthesis

Articulatory synthesis is a method in which the human vocal organs are modeled. The name of the method is inspired by the term 'Articulators' which implies speech organs like jaw, tongue, lips etc. It is potentially the most gratifying method to produce high-quality synthetic speech. The computational load of this method is considerably higher as compared to other common methods. It is an attempt to follow actual speech production mechanism.

This method is also one of the most difficult methods to implement and hence has not yet achieved the same level of success as other synthesis method. But due to the rapid increase in analysis method and computational resources, it might be a potential synthesis method in the future [21].

2.3.2 Formant Synthesis

Probably the most widely used synthesis method during last decades has been formant synthesis which is based on the source-filter-model of speech. It consists of artificial reconstruction of the formant characteristics to be produced. This is done by exciting a set of resonators by a voicing source or noise generator to achieve the desired

speech spectrum. Parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech [4]. This method is sometimes called rules-based synthesis.

2.3.3 Concatenative Synthesis

Concatenative synthesis is based on the concatenation of segments of recorded speech. In this method waveform segments are stored in a database. For a given text, these segments are joined based on some joining rules. Connecting pre-recorded natural utterances is probably the easiest way to produce intelligible and natural sounding synthetic speech. One of the most important aspects in concatenative 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 labelling procedures become more difficult and complex [3]. There are three main sub-types of concatenative synthesis.

- Unit selection synthesis
- Diphone Synthesis
- Domain-specific synthesis

These three methods of concatenative synthesis are discussed below.

Unit Selection Synthesis

In unit selection synthesis large databases of recorded speech are used [16]. The primary motivation for the use of large databases is that with a large number of units available with varied prosodic and spectral characteristics it should be possible to synthesize more natural-sounding speech than that can be produced with a small set of controlled units [11]. During database creation, each recorded utterance is segmented into some or all of the following: individual phones, diphones, half-phones, syllables, morphemes, words, phrases, and sentences [19].

Diphone Synthesis

Diphone synthesis uses a minimal speech database containing all the diphones occurring in a language. In diphone synthesis, only one example of each diphone is

contained in the speech database. The quality of the resulting speech is generally worse than that of unit-selection systems, but more natural-sounding than the output of formant synthesizers [4].

Domain-specific synthesis

Domain-specific synthesis concatenates pre-recorded words and phrases to create complete utterances. It is used in applications like transit schedule announcements or weather reports, where the variety of texts the system will output is limited to a particular domain [4]. Because the variety of sentence types is limited, and they closely match the prosody and intonation of the original recordings, the level of naturalness of these systems can be very high.

2.4 Text to Speech Systems for Indian Language

Speech synthesis involves algorithmically converting an input text into speech waveforms and some previously coded speech data. Speech synthesizers can be characterized by the size of the speech units they concatenate, as well as by the method used to code, store and synthesize the speech. The choice of synthesis method is influenced by the size of the vocabulary because they must model all possible utterances. Unlimited-text systems are generally more complex, and yield lower quality speech than limited-text systems. Thus it is seen that various different techniques exist to convert a given text to speech and that each such system has its own advantages and disadvantages[2].

In India, to help the visually impaired, vocally disabled and day to day increasing applications of speech synthesis has necessitated the development of more and more innovative text-to-speech (TTS) system. Some of the already developed TTS are described below.

2.4.1 Dhvani- Indian Language Text To Speech System

Dhvani [29] [7] system has won FOSS India award in 2008. The main characteristics of this system are as follows.

- Dhvani is a Text To Speech System specially designed for Indian languages.
- This system has been developed by Simputer trust headed by Dr. Ramesh Hariharan at Indian Institute of Science Bangalore in year 2000.

- It uses diphone concatenation algorithm.
- Currently this system has Hindi, Malayalam, Kannada, Bengali, Oriya, Punjabi, Gujarati, Telugu and Marathi modules.
- All sound files stored in the database are *'gsm'* compressed files.
- It has different modules for every language.
- It is based on the observation that a direct grapheme to phoneme mapping exits for all Indian languages in general.
- It is an attempt in India to cover all Indian languages under a single framework.
- In this system each language requires a Unicode parser.

2.4.1.1 Architecture of *Dhvani* System

The architecture of ***Dhvani*** system is shown in figure 2.7. It includes the following components.

Text Parser: Each language requires a unicode parser. It simply parses the input text.

Text to Dhvani Phonetic Script: This makes ***Dhvani***, language independent. In this phase any Unicode text is converted to a common script. This script is further act as the input to the speech synthesizer.

Grapheme to Phoneme Conversion: This converts the grapheme to phoneme. The phonetic description is syllable based. In this eight kinds of sounds are allowed.

Sound Database: All sound files stored in the database are *'gsm'* compressed files.

Speech Synthesizer: Speech synthesizer takes phonetic script and with the help of CV (consonant-vowel) pair algorithm and sound concatenation component concatenates the sound files to produce speech.

Limitations of *Dhvani*

This speech engine has not made any attempt to do prosody on the output. It simply concatenates basic sound units at pitch periods and plays them out. Adding prosody is a task for the future. Also, the total size of the database is currently around 1MB,

though work is going on to get it down to about half the size by storing only parts of vowels and extending them on the fly.

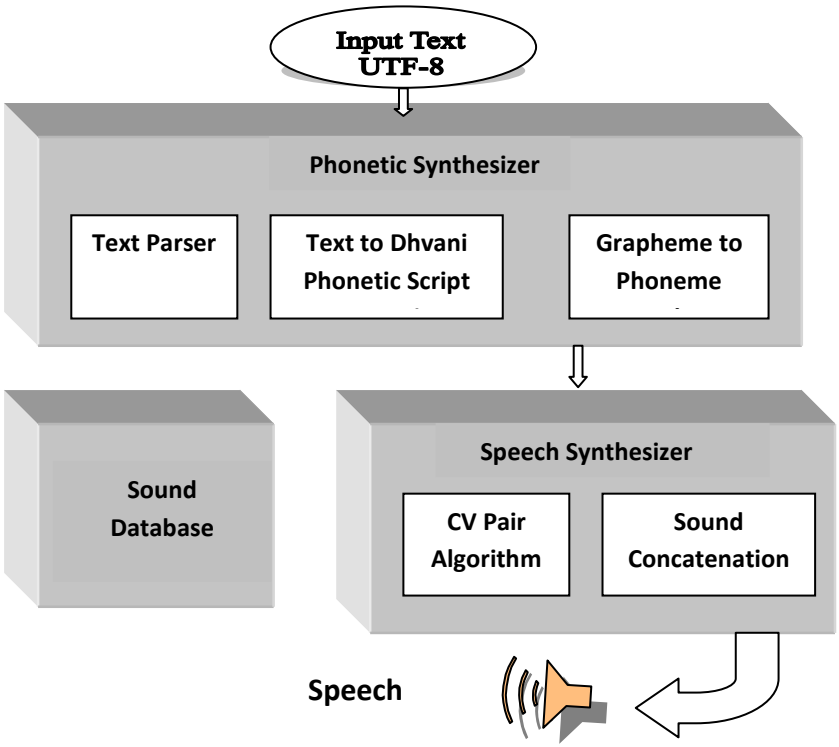


Figure 2.7: Dhvani- Indian language Text to Speech System [29]

2.4.2 *Shruti:* An Embedded Text To Speech System for Indian Languages

Mukhopadhyay *et al.* [22] have developed ***Shruti*** system in year 2006 at Indian Institute of Technology Kharagpur. The main characteristics of this system are as follows.

- It is a text-to-speech system, which has been developed using a concatenative speech synthesis technique.
- This is the first text-to-speech system built specifically for two of the Indian languages, namely Bengali and Hindi.
- The system, however, has been designed in such a manner that it can be very easily extended to other Indian languages as well.
- For ease of use and portability, the system has been ported to two existing handheld platforms running two processor families, namely, the Compaq iPaq and the Casio Cassiopeia.

2.4.2.1 Architecture of *Shruti* system

A schematic diagram of the speech synthesis system is shown in figure 2.8. The system consists of two main blocks, block-A and block-B, where block-A is the language-dependent block and block-B is the Indian language phonetic synthesizer (ILPS).

The block-A consists of an input device, a natural language processing (NLP) unit and an intonational and prosodic rule base. The system accepts input text in iTrans, a notation for expressing Indian language scripts through Roman characters. A language-dependent prosodic and intonation rules are included in this block primarily as a knowledge base. The NLP in block-A comprises a phoneme parser, which uses either a phonological dictionary, syllable and word marker or a linguistic rule base to produce an output phonemic string. The NLP simply takes the text and analyses it to perform grapheme to phoneme conversion. The grapheme is the actual text, whereas the phoneme is a token that directly maps to a signal unit in the voice database or in the partneme dictionary.

The block-B is the synthesizer. The output speech is produced by taking the output string of phonemes (in ILPS symbols) and information for intonation and prosody from the basic rule base as input. The concatenation unit reads the phonemes one by one from the output phoneme string. For each read phoneme, the unit fetches the corresponding sound unit that the phoneme maps to from the voice database. To make the speech output more natural, the speech units are algorithmically modified before being concatenated. Finally, intonational and prosodic rules are applied to produce the final output speech [22] [2].

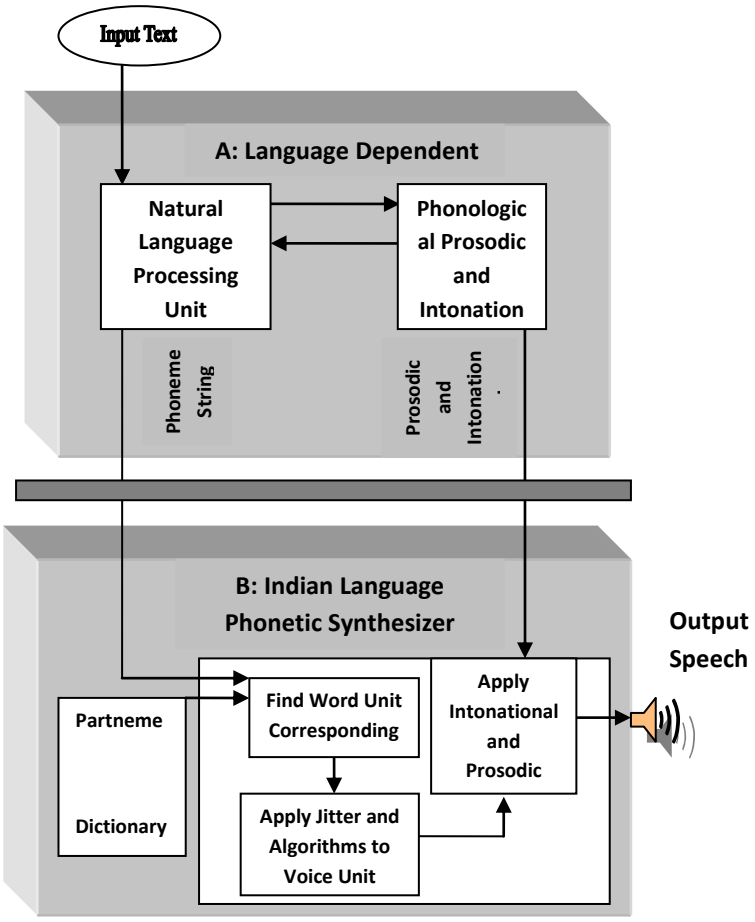


Figure 2.8: Schematic Diagram of ***Shruti*** System [22]

2.4.3 HP Labs India TTS System

HP Labs India developed a Hindi TTS system based on the open source TTS framework, Festival. This effort is a part of the Local Language Speech Technology Initiative (LLSTI), which facilitates collaboration between motivated groups around the world, by enabling sharing of tools, expertise, support and training for TTS development in local languages [24].

2.4.3.1 Architecture of HP Lab India System

As part of the LLSTI initiative, a generic grapheme [27] into phoneme conversion system has been developed at HP Labs India. The architecture of the grapheme to phoneme (G2P) is shown in Figure 2.9. This G2P framework has then been customized for Hindi. It is composed of following units.

Grapheme to Phoneme Conversion Engine: It is a language independent engine, which requires language dependent information in the form of lexicon, rules and mapping.

G2P Rule Base: It gives grapheme to phoneme conversion rule.

Character Phone Mapping*:* It provides the character phone mapping.

Exception lexicon: It contains the phonetic transcription of the exceptions.

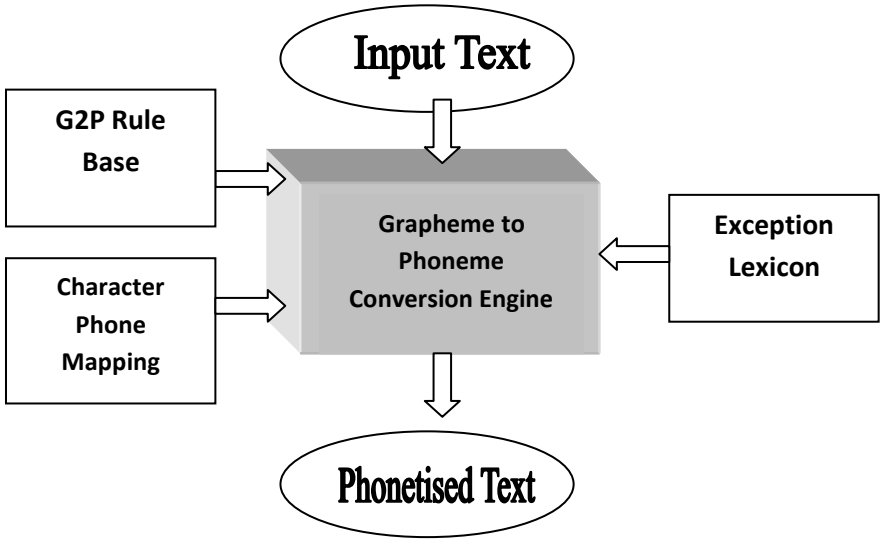


Figure 2.9: HP Lab India TTS System Architecture [24]

2.4.4 *Vani*: An Indian Language Text to Speech Synthesizer

Vani [12] [13] is an Indian Language text to speech synthesizer developed at IIT Bombay, India. Generally, all existing TTS system allows user to specify what is to be spoken but does not give any control on how it has to be spoken. In ***Vani***, a new encoding scheme has been introduced called *vTrans*. A *vTrans* file makes a person to encode what text he wants to be spoken and also the way that text to be spoken. A signal processing module is then used to bring out this speech by making appropriate variations to the sound database.

vTrans is an XML document that contains a head and a body. In the head part of XML document the parameters and styles are defined. These parameters are pitch, volume and duration. The body section contains several tags. They may be nested, but

text is put only in the innermost of the tags. The text within the tags is iTrans encoded. The tags may be any of the parameters defined in the head section. The attributes assigned to these tags determines which style to use and allow the user to scale and translate the function to be used as required.

2.4.4.1 Architecture of *Vani* System

The architecture of ***Vani*** system is given in Figure 2.10. It is based on fact that phonemes can be decomposed into fract-phonemes (a very small segment whose continuous repetition forms a vowel) for all vowels. These fract-phonemes are very small in size, and hence a good choice for a base unit.

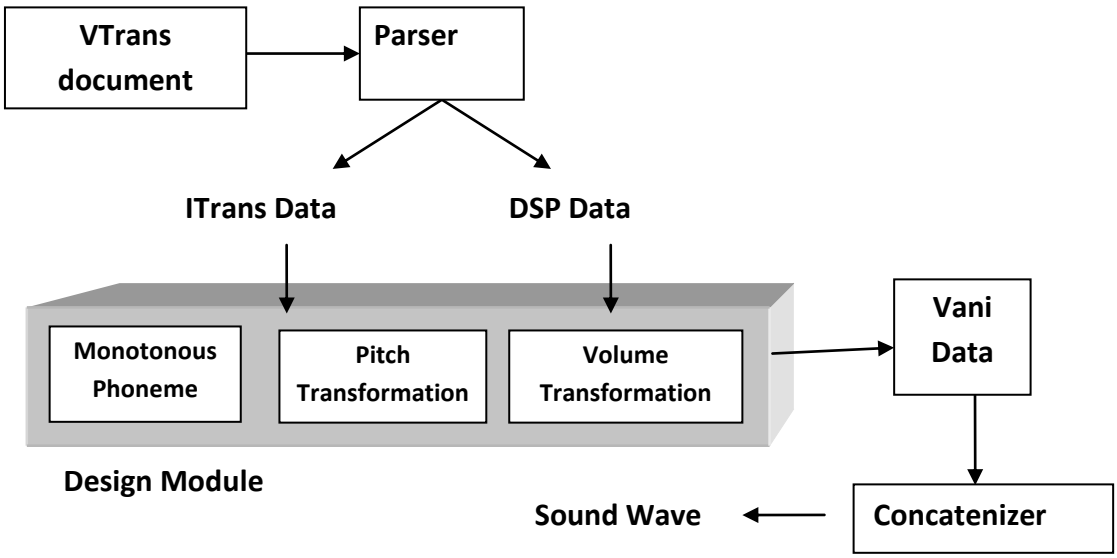


Figure 2.10: Vani Architecture [13]

The duration a phoneme can be generated by taking into account the volume and frequency curves using fract-phoneme as per the following procedure.

- Virtual duration is calculated.
- Monotonous phoneme of virtual duration is generated.
- Pitch transformation is applied.
- Volume transformation is applied.

2.5 Comparison of Indian Languages TTS System

In the above sections four TTS systems have been discussed, namely **Dhvani**, ***Shruti***, ***HP Lab system*** and ***Vani*** system. These systems are compared in Table 2.2 on the basis of languages it support, size of the speech units, methodologies used to code, store and synthesize the speech and prosody.

Table 2.2 Comparison of Indian Languages TTS system

SNO	NAME OF SYSTEM	LANGUAGE SUPPORTED	SYNTHESIS STRATEGY	UNIT/DATA BASE	TEXT PROCESSING /TOOLS	PROSODY
1.	Dhvani-Indian Language Text To Speech System.	Hindi, Malayalam, Kannada, Bengali, Oriya, Punjabi, Gujarati, Telugu, Marathi	Diphone Concatenation	Syllable database	Parsing rules for phonemization (separate Unicode parser for each language)	Still a research area
2.	Shruti: an embedded text-to-speech system for Indian languages.	Bengali and Hindi	Concatenation Synthesis	Partneme database	Parsing rules for phonemization	Prosodic and intonational rules are applied.
3.	Text Analysis Module Created by HP Lab India.	Hindi	Concatenation Synthesis	Phoneme or syllable database	Festival based-Fest VOX tools	No prosody for specific language (research area)
4.	Vani: An Indian Language Text to Speech Synthesizer.	Hindi	Concatenation Synthesis	Fract-phoneme database	Encoding scheme called vTrans is used, phoneme is generated from fract-phoneme	Parameters *to* control speech like pitch, volume and duration is given by user.

By referring Table 2.1 in which ***Dhvani***, ***Shruti***, ***HP Lab System*** and ***Vani*** system are compared on the basis of languages it support, size of the speech units, methodologies used to code, store and synthesize the speech and prosody, we conclude that Dhvani is the tool which has maximum languages modules and has the better quality result as compared to all other systems. Hence Dhvani is used as the back end tool in the user friendly Hindi text to speech system designed in this thesis, which is discussed in further chapters.

Chapter 3

Problem Statement

Many text to speech systems for Hindi language exist but they are not so user friendly. Hence, to full fill this requirement a user friendly text to speech interface for Hindi language has been developed. The problem statement in the next section discusses about it.

3.1 Problem Statement.

Many Indian languages TTS has been developed, but they don't have a good user friendly interface. The aim of this thesis is to develop a user friendly interface for converting Hindi text to speech form. The intermediate stages in converting the text to speech are also analysed in this work.

3.2 Objective.

The main objectives of the user friendly Hindi text to speech system are:

- To design a user interface in which the end user can write editable Hindi text to be converted into speech in the text box.
- To design a user interface having keyboard of Hindi characters so that the end user can easily type the text in Hindi.
- To make a system in which the end user can listen the text by click of a button.
- To make a system in which the end user can listen full text, only the selected text and also has an option to listen selected text character wise.
- To make a system in which the end user can convert the text from a saved Hindi text file to speech.
- To make a system in which the end user can increase or decrease the speed of the speech of text.
- To make a system which overcome the limitation of ***Dhvani*** TTS of platform dependency, *i.e*, it can run on any platform like WindowsXP, Window7, Linux etc.

- To make a system having female voice database for converting Hindi text to speech.

3.3 Methodology Used.

Following methodology is used to achieve the above mentioned objectives.

- Study the various existing text to speech conversion methodologies.
- Study various text to speech conversion systems developed for Indian languages.
- Study the architecture of *Dhvani*- Indian language text-to-speech system and study the intermediate stages which the input text go through to produce the final output.
- Install *Dhvani*- Indian language text-to-speech system in Linux.
- Design the interface in the Java netbeans 6.8, adding all the required buttons and text areas.
- Map Dhvani as a back-end tool for generating speech at the interface.
- Design the interface to show the various intermediate stages like phonetic conversion, classification of characters, sound units formed and correct phonetic transcription for converting Hindi text to speech.
- Design a new Hindi TTS module in java netbeans 6.8 which overcome the limitations of Dhvani TTS of working at only Linux platform and having no female voice database.
- For new Hindi TTS module a new database is designed in female voice and proper mapping between the sound and characters is done.

Chapter 4

Design and Implementation

This chapter discusses about the user friendly interface for Hindi text to speech conversion which is developed in java netbeans 6.8 platform at Linux. It frees the user from typing commands at Linux command prompt and allows the text to speech conversion by clicks of buttons. The tool which works behind this interface is ***Dhvani***, whose speech synthesis process is described in the coming sections. After analysing various features of ***Dhvani***, some of its limitations like it works only in Linux environment and has no female voice database has come across. To deal with these limitations a new Hindi text to Speech module has also been proposed.

4.1 User Friendly Hindi Text to Speech System based on Dhvani

A user friendly Interface for Hindi text to speech conversion has been developed which has the following features.

- It has the features to read full text, only the selected text and selected text character-wise also.
- It consists of keyboard of Hindi characters so that the user can easily type the text in Hindi in the text box.
- It also provides the facility to save the speech file of the input text and can also play any of the previously saved audio file.
- Various intermediate stages of converting text to speech is also analyzed and shown in the interface.

Dhvani has been implemented in C at Linux platform. The speech synthesis process of Dhvani is mainly divided into two modules as given below.

- Letter to Sound (LTS) module
- Phonetic to Speech (PTS) module

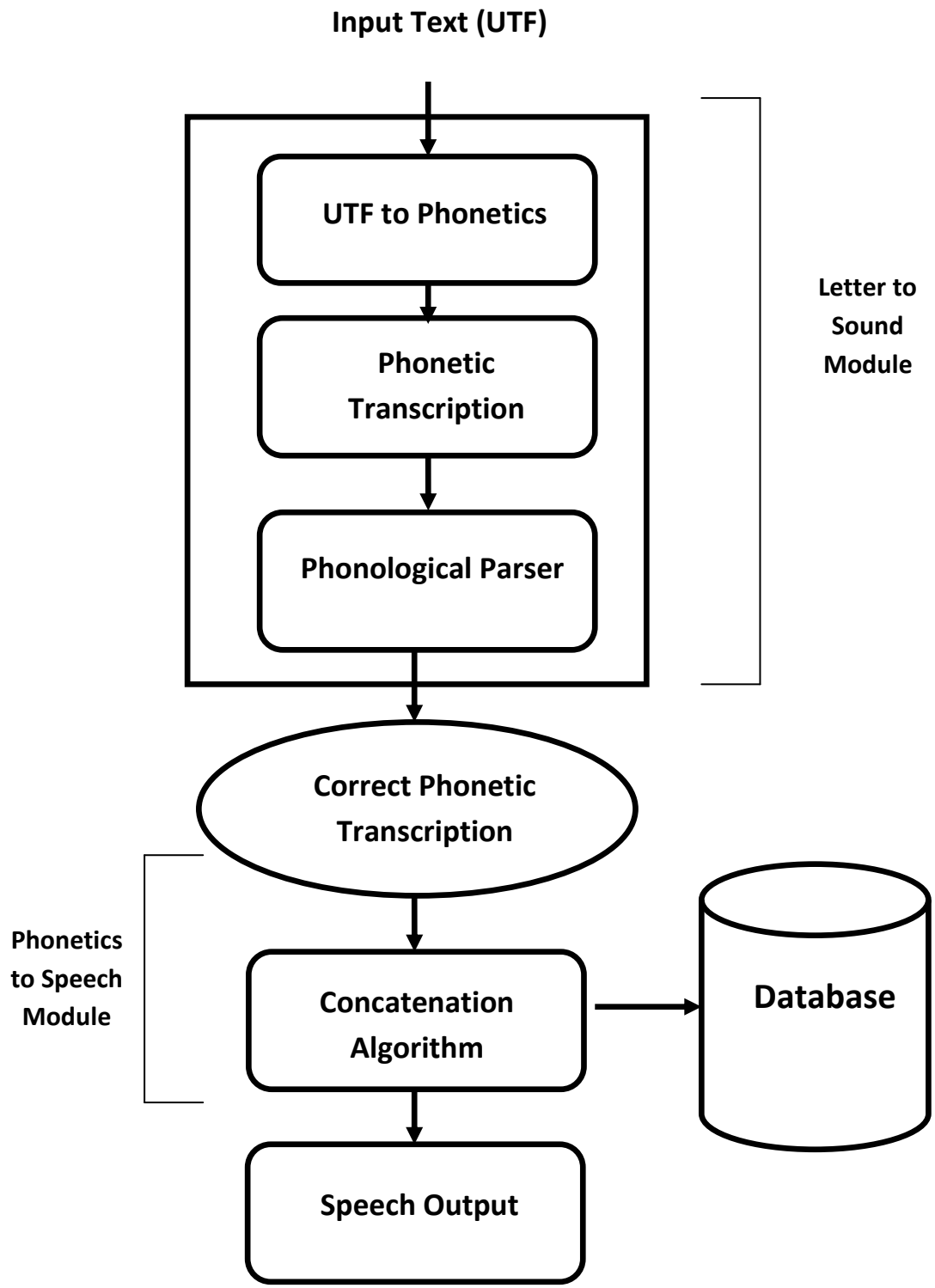


Figure 4.1: Dhvani- Speech Synthesis Process[9]

The Letter to Sound (LTS) module converts the input text into its equivalent phonetic transcription. This string is parsed through a set of language specific rules to get the correct phonetic representation. This string is then fed into the Phonetic to Speech (PTS) module. The concatenation algorithm reads the phonetic description and selects the appropriate diphones from the database and then joins them at pitch-point. These two modules are discussed in detail in further sections. The speech synthesis process of ***Dhvani*** is shown in figure 4.1.

4.1.1 Letter to Sound Module

The Letter to Sound module for Hindi reads a text in UTF-8 format and converts this text into phonetic description[8]. This is achieved by a careful mapping of the UTF symbols for Hindi into the phonetic symbols in the database. Simultaneously, each symbol is tagged as a *Consonant* (***C***), *Vowel* (***V***), or *Halant* (***H***).

A. Conversion to Phonetics

As Hindi language scripts is syllabic in nature and have a one to one correspondence between the letters and the sounds, this is a simple process of mapping the letters to a phonetic alphabet.

Vowels

A vowel is a sound in spoken language, pronounced with an open vocal tract so that there is no build-up of air pressure at any point above the glottis**.** In ***Dhvani*** there are 15 vowels allowed which is shown in Table 4.1.

Table 4.1 Vowels in ***Dhvani***

SNo	Vowel Notation	Example
1	a	Pun
2	aa	Saal
3	i	Pin
4	ii	Keen
5	u	Pull
6	uu	Pool
7	e	Met
8	ee	Meet
9	ae	Mate

10	ai	Height
11	o	Gold
12	oo	Court
13	au	Call
14	ow	Cow
15	tamil-u	tamil aanddu (as in tamil)

In phonetic description, the numbers 1-15 are used as symbols to represent vowels. In Hindi language script all of the above mentioned vowels are not used. The vowels used in Devnagri script are given in Table 4.2.

Table 4.2 Phonetic Symbols for Vowels

Vowel Notation	**Vowels in Hindi**	**Phonetic Symbol**
a	‘अ‘	1
aa	’आ’ /’ा’	2
i	’इ’/’ि’	3
ii	’ई’ /’ी’	4
u	’उ’/’ु’	5
uu	’ऊ’ /’ू’	6
e	’ए’/’े’	7
ai	’ऐ’/’ै’	10
o	’ओ’/’ो’	11
au	’औ’/’ौ’	13

Consonants

Consonants is a sound in spoken language, which is pronounced with a constriction or closure at some point along the vocal tract. In Dhvani there are 34 consonants allowed which is shown in Table 4.3.

Table 4.3 Consonants in ***Dhvani***

k	kh	g	gh	
ch	chh	j	jh	
t	th	d	dh	n
tt	tth	dd	ddh	nna
p	f	b	bh	m
y	r	l	ll	v
sh	s	h		
zh	z	an		

Not all above mentioned consonants came across in Hindi language. The consonants which are used in Hindi module are given in Table 4.4.

Table 4.4 Phonetic Symbols of Consonants

Hindi Character	**Phonetic Symbol**
‘क’	’k’
’ख’	’kh’
’ग’	’g’
’घ’	’gh’
‘ङ’	‘nna’
’च’	’ch’
’छ’	’chh’
’ज’	’j’
’झ’	’jh’
‘ञ’	‘nna’
’ट’	’tt’
’ठ’	’tth’
’ड’	’dd’
’ढ’	’ddh’
‘ण'	‘nna’
‘त’	‘t’

Hindi character	**Phonetic Symbol**
’थ’	’th’
’द’	’d’
’ध’	’dh’
'न'	‘n’
’प’	’p’
’फ’	’ph’
’ब’	’b’
’भ’	’bh’
’म’	’m’
’य’	’y’
’र’	’r’
’ल’	‘l’
’व’	’v’
’श’/’ष’	’sh’
’स’	’s’
’ह’	’h’

Half Consonants

H is a *Halant* used for denoting consonants without the inherent vowel or half consonants. The concept of half *Consonant* is based on the orthographical representations of Indian languages which are essentially syllabic in nature. Each letter representing a consonant has an inherent vowel associated with it, usually, *schwa* or 'a'. A true consonant, *i.e.*, without any vowel associated with it is marked by a special ligature known as *Halant*. Thus, a structure of the form HCV is essentially a consonant followed by a CV, or, in other words, a consonant cluster followed by a vowel. Dhvani allows the half sounds which are shown in Table 4.5.

Table 4.5 Half Sounds

ky	kr	kl	kll	kv	ksh
khy	khr	khl	khv		
gy	gr	gl	gv	gn	
ghy	ghr	ghv	ghn		
chy	chr	chv			
jy	jv				
ty	tr	tv			
thy	thr				
dy	dr	dv			
dhy	dhr	dhv			
ny	nr	nr			
tty	ttr	ttv			
ddy	ddr	ddu			
py	pr	pl	pll		
fr	fl				
by	br	bl			
bhy	bhr	bhl			
my	mr				
vy	vr	vl			

The half sounds which are not present in the above mentioned list are denoted by ***0C***. ***0C*** is a sound of consonant without the inherent vowel *schwa* or 'a'. The half sounds used in Hindi module are shown in Table 4.6.

Table 4.6 Half Sounds for Hindi Module

क्य	क्र	क्ल	क्व	क्श
ख्य	ख्र	ख्ल	ख्व	
ग्य	ग्र	ग्ल	ग्व	ग्न
घ्य	घ्र	घ्व	घ्न	
च्य	च्र	च्व		
ज्य	ज्व			
त्य	त्र	त्व		
थ्य	थ्र			
ड्य	ड्र	ड्व		
ध्य	ध्र	ध्व		
न्य	न्र	न्व		
ट्य	ट्र	ट्व		
ड्य	ड्र	ड्व		
प्य	प्र	प्ल		
फ्र	फ्ल			
ब्य	ब्र	ब्ल		
भ्य	भ्र	भ्ल		
म्य	म्र			
व्य	व्र	व्ल		

By following above standards, the phonetic transcription of the input text can be easily done. For example let the word is "मुबारक", its phonetic transcription is shown in Table 4.7.

Table 4.7 Phonetic Transcription of Word "मुबारक"

Hindi Character	Phonetic Symbol
म	m
ु	5
ब	b
ा	2
र	r
क	k

Hence, phonetic transcription of the word "मुबारक" is "m5b2rk". Similarly, phonetic transcription of the sample 50 Hindi words are shown in Table 4.8.

Table 4.8 Phonetic Transcription of Hindi words

SNo	Word	Phonetic Transcription
1.	समारोह	sm2r11h
2.	खरगोश	khrg11sh
3.	गपशप	gpshp
4.	किताब	k3t2b
5.	यातायात	y2t2y2t
6.	नौजवानौ	n13jv2n13
7.	गुलाब	g5l2b
8.	कलम	klm
9.	अजायब	1j2yb
10.	कारीगर	k2r4gr
11.	सरगम	srgm
12.	सरल	srl

13.	लड़की	lddk4
14.	लेखक	l7khk
15.	पानी	p2n2
16.	कबूतर	kb6tr
17.	अतिथि	1t3th3
18.	रविवार	rv3v2r
19.	सोमवार	s11mv2r
20.	सफलता	sphlt2
21.	अभितेश	1bh3t7sh
22.	पूजन	p6jn
23.	शनिवार	shn3v2r
24.	छिपकली	chh3pkl4
25.	आकाश	2k2sh
26.	विमान	v3m2n
27.	कन्या	knHy2
28.	सम्मान	s0mm2n
29.	नमस्कार	nm0sk2r
30.	मुस्कान	m50sk2n
31.	पुरस्कार	p5r0sk2r
32.	सम्मेलन	s0mm8ln
33.	पुस्तक	p50stk
34.	स्कूल	0sk6l
35.	पटियाला	pt3y2l2
36.	अनार	1n2r
37.	दिव्या	d3vHy2
38.	भोपाल	bh11p2l
39.	गुमराह	g5mr2h
40.	आदत	2dt
41.	होशियारी	h11sh3y2r4
42.	चिडिया	ch3dd3y2
43.	रुड़की	r6ddk4
44.	खतरनाक	khtrn2k
45.	जलन्धर	Jl0ndhr

46.	मसूरी	ms6r4
47.	देहरादून	d7hr2d6n
48.	खुशमिजाज	kh5shm3j2j
49.	देखभाल	d7khbh2l
50.	थापर	th2pr

B. Classification of Characters

The characters in the Indian languages are basically of three types namely, *Consonants (***C***), Vowels (**V**)* and *Halant* (***H***). In order to find the correct sound units of the words, there is a need to do their character classification. Algorithm 4.1 is followed to perform the character classification which is given below.

Algorithm 4.1

1. Read the token and convert it into an array of characters.
2. Traverse each character of the token and check-
   (a) If the Unicode of character matches with any of the Unicode of symbol present in Table 4.10, then it is classified as a consonant 'C'.
   (b) If the Unicode of character matches with any of the Unicode of symbol present in Table 4.11, then it is classified as a vowel 'V'.
   (c) If the Unicode of character matches with any of the symbol present in table 4.6, followed by a Halant, then it is classified as a half-consonant 'Ch' or 'H'.

By following above algorithm, the character classification of the input text can be easily done. For example let the word is "मुबारक", its character classification is shown in Table 4.9.

Table 4.9 Character Classification of word “मुबारक”

Hindi Character	Character Classification
म	C
ु	V
ब	C
ा	V
र	C
क	C

Hence, character classification of “मुबारक” is “CVCVCC”.

Table 4.10 Consonants

क	ख	ग	घ	ङ
च	छ	ज	झ	ञ
ट	ठ	ड	ढ	ण
त	थ	द	ध	न
य	र	ल	व	
स	श	ष	ह	

Table 4.11 Vowels

अ	आ	इ	ई	उ	ऊ	ए	ऐ	ओ	औ
	ा	ि	ी	ु	ू	े	ै	ो	ौ

Similarly, character classification of the sample 50 Hindi words are shown in Table 4.12.

Table 4.12 Character Classification of Hindi words

SNo	Word	Phonetic Transcription	Classification of Characters
1.	समारोह	sm2r11h	CCVCVC
2.	खरगोश	khrg11sh	CCCVC
3.	गपशप	gpshp	CCCC
4.	किताब	k3t2b	CVCVC
5.	यातायात	y2t2y2t	CVCVCVC
6.	नौजवानौ	n13jv2n13	CVCCVCV
7.	गुलाब	g5l2b	CVCVC
8.	कलम	Klm	CCC
9.	अजायब	1j2yb	VCVCC
10.	कारीगर	k2r4gr	CVCVCC
11.	सरगम	srgm	CCCC
12.	सरल	srl	CCC
13.	लड़की	lddk4	CCCV
14.	लेखक	l7khk	CVCC
15.	पानी	p2n2	CVCV
16.	कबूतर	kb6tr	CCVCC
17.	अतिथि	1t3th3	VCVCV
18.	रविवार	rv3v2r	CCVCVC
19.	सोमवार	s11mv2r	CVCCVC
20.	सफलता	sphlt2	CCCCV
21.	अभितेश	1bh3t7sh	CCVCVC
22.	पूजन	p6jn	CVCC
23.	शनिवार	shn3v2r	CCVCVC
24.	छिपकली	chh3pkl4	CVCCCV
25.	आकाश	2k2sh	VCVC
26.	विमान	v3m2n	CVCVC
27.	कन्या	knHy2	CHCV
28.	सम्मान	s0mm2n	CChCVC
29.	नमस्कार	nm0sk2r	CCChCVC
30.	मुस्कान	m50sk2n	CVChCVC

31.	पुरस्कार	p5r0sk2r	CVCChCVC
32.	सम्मेलन	s0mm8ln	CChCVCC
33.	पुस्तक	p50stk	CVChCV
34.	स्कूल	0sk6l	ChCVC
35.	पटियाला	pt3y2l2	CCVCVCV
36.	अनार	1n2r	VCVC
37.	दिव्या	d3vHy2	CVChCV
38.	भोपाल	bh11p2l	CVCVC
39.	गुमराह	g5mr2h	CVCCVC
40.	आदत	2dt	VCC
41.	होशियारी	h11sh3y2r4	CVCVCVCV
42.	चिडिया	ch3dd3y2	CVCVCV
43.	रुड़की	r6ddk4	CVCCV
44.	खतरनाक	khtrn2k	CCCCVC
45.	जलन्धर	Jl0ndhr	CCChCC
46.	मसूरी	ms6r4	CCVCV
47.	देहरादून	d7hr2d6n	CVCCVCVC
48.	खुशमिजाज	kh5shm3j2j	CVCCVCVC
49.	देखभाल	d7khbh2l	CVCCVC
50.	थापर	th2pr	CVCC

C. Phonological Parser

Most Indian languages have a simple mapping between orthographic units and sound units. Sanskrit from which Hindi has been originated, is phonetically almost perfect. So for a TTS in Sanskrit, no language processing is required. Hindi on the other hand, does require some language processing. In Hindi the inherent schwa associated with a consonant is suppressed depending upon context in which it is used.

The vowel "a" which inherently occurs in all consonants is called as **schwa**. There are some cases in Hindi where schwa after certain characters are not pronounced. For example consider “sa-ra-la” it would be pronounced as “sa-ra-l”. The "a" following the "l" is deleted. A number of facts are responsible for the issues of schwa pronunciation in Hindi. It is clear that not every schwa is pronounced. Particularly, the

word final schwa is invariably deleted. Within a word also, the schwa can be deleted in certain positions, and in compound words, the presence of a morpheme boundary blocks schwa deletion where it might otherwise occur. Hence, a general rule is applied to insert schwa in the phonetic string.

In Dhvani the text to be spoken out must be expressed in terms of these eight types of sound units as shown in Table 4.13.

Table 4.13 Sound Units in Dhvani

Sound Unit	Description
V	Plain vowel
CV	A consonant followed by a vowel
VC	A vowel followed by a consonant
CVC	A consonant followed by a vowel followed by a consonant
HCV	A half consonant followed by a CV
HCVC	A half consonant followed by a CVC
0C	A consonant alone
G[0-9]	A silence of specified length

In Dhvani the Hindi parser uses an algorithm that parses recursively, from right to left. The algorithm 4.2 is followed by the phonological parser in which it takes the character classified string as input and produce the correct phonetic transcription of the words.

Algorithm 4.2

1. Start parsing the symbol from the right side i.e. from the rightmost to the leftmost.
2. If the leftmost symbol is ‘C’ then replace it by ‘Ch’ i.e. it will become a consonant having Halant.
3. Follow the steps 4-6 till each token of the input is traversed.
4. If the symbol traverse is ‘C’, then perform the below mentioned cases-

(a) If the leftmost symbol of symbol being traversed is ‘C’, replace it by “C1C”, *i.e.*, form a class of “CVC” and shifts two places towards leftmost.

(b) If the leftmost symbol of symbol being traversed is ‘V’, then

(i) Check if left of - leftmost symbol unit of symbol being traversed is ‘C’, then form a class of “CVC” and shifts three places towards leftmost.

(ii) Else form a class of “VC” and shifts two places towards leftmost.

(c) If the leftmost symbol of symbol being traversed is ‘Ch’, replace it by “C1C”, *i.e.*, form a class of “CVC” and shifts two places towards leftmost.

(d) If the leftmost symbol of symbol being traversed is ‘ ’, replace it by “C1” i.e. form a class of “CV”.

5. If the symbol traverse is ‘V’, then perform the below mentioned cases-

(a) If the leftmost symbol of symbol being traversed is ‘C’, form a class of “CV” and shifts two places towards leftmost.

(b) If the leftmost symbol of symbol being traversed is ‘V’, form a class of “V” and shifts one places towards leftmost.

(c) If the leftmost symbol of symbol being traversed is ‘Ch’, form a class of “CV” and shifts two places towards leftmost.

(d) If the leftmost symbol of symbol being traversed is ‘ ’, form a class of “V” and shifts one places towards leftmost.

6. If the symbol traverse is ‘Ch’, then perform the below mentioned cases-

(a) If the leftmost symbol of symbol being traversed is ‘C’, replace it by “C1C”, *i.e.*, form a class of “CVC” and shifts two places towards leftmost.

(b) If the leftmost symbol of symbol being traversed is ’V’, then

(i) Check if left of - leftmost symbol of symbol being traversed is ’C’, then form a class of “CVC” and shifts three places towards leftmost.

(ii) Else form a class of “VC” and shifts two places towards leftmost.

(c) If the leftmost symbol of symbol being traversed is ’Ch’, then

(i). Check if rightmost symbol being traversed is ’V’, *i.e.* they form a CV pair and the corresponding 'half' sound is also available then form a class of “ChCV” and shifts two places towards leftmost.

(ii). Else form a class of “0C” and shifts one place towards leftmost.

(d) If the leftmost symbol of symbol being traversed is ’ ’ , form a class of “C”.

7. End.

By following above algorithm, the sound classes of the input phonetic string can be made. For example let the word is “मुबारक”, its sound classes formed is shown in Table 4.14.

Table 4.14 Sound Classes of word “मुबारक

Input String	**Classes Formed**	**Rules Applied**
↓ CVCVCC		Since leftmost is a consonant ‘C’ replaces it by ‘Ch’(From Step 2)
↓ CVCVCCh	CVC	Now the symbol traversed is ‘Ch’, and its leftmost sound unit is ‘C’, so it will be replaced by “C1C” and a class of “CVC” will be formed. Pointer will be shifted two places left(From Step 6(a))
↓ CVCVCCh	CV	The symbol traversed is ‘V’ and its leftmost symbol is ‘C’, hence a class of “CV” will be formed. Pointer will be shifted two places

		left (From 5(a))
↓ CVCVCCh	CV	The symbol traversed is again 'V' and its leftmost symbol is 'C', hence again a class of "CV" will be formed. Pointer will be shifted two places left.(From 5(a))

Hence sound classes formed of the word "मुबारक" are "CV" "CV" "CVC"

Similarly, sound classes of the sample 50 Hindi words are shown in Table 4.15.

Table 4.15 Sound Units

SNo	Word	Phonetic Transcription	Classification of Characters	Sound Classes
1.	समारोह	sm2r11h	CCVCVC	CV CV CVC
2.	खरगोश	khrg11sh	CCCVC	CVC CVC
3.	गपशप	gpshp	CCCC	CVC CVC
4.	किताब	k3t2b	CVCVC	CV CVC
5.	यातायात	y2t2y2t	CVCVCVC	CV CV CVC
6.	नौजवानौ	n13jv2n13	CVCCVCV	CVC CV CV
7.	गुलाब	g5l2b	CVCVC	CV CVC
8.	कलम	klm	CCC	CV CVC
9.	अजायब	1j2yb	VCVCC	V CV CVC
10.	कारीगर	k2r4gr	CVCVCC	CV CV CVC
11.	सरगम	srgm	CCCC	CVC CVC
12.	सरल	srl	CCC	CV CVC
13.	लड़की	lddk4	CCCV	CVC CV
14.	लेखक	l7khk	CVCC	CV CVC
15.	पानी	p2n2	CVCV	CV CV
16.	कबूतर	kb6tr	CCVCC	CV CV CVC
17.	अतिथि	1t3th3	VCVCV	V CV CV
18.	रविवार	rv3v2r	CCVCVC	CV CV CVC
19.	सोमवार	s11mv2r	CVCCVC	CVC CVC
20.	सफलता	sphlt2	CCCCV	CV CVC CV

21.	अभितेश	1bh3t7sh	CCVCVC	V CV CVC
22.	पूजन	p6jn	CVCC	CV CVC
23.	शनिवार	shn3v2r	CCVCVC	CV CV CVC
24.	छिपकली	chh3pkl4	CVCCCV	CV CVC CV
25.	आकाश	2k2sh	VCVC	V CVC
26.	विमान	v3m2n	CVCVC	CV CVC
27.	कन्या	knHy2	CChCV	CVC CV
28.	सम्मान	s0mm2n	CChCVC	CVC CVC
29.	नमस्कार	nm0sk2r	CCChCVC	CV CVC CVC
30.	मुस्कान	m50sk2n	CVChCVC	CVC CVC
31.	पुरस्कार	p5r0sk2r	CVCChCVC	CV CVC CVC
32.	सम्मेलन	s0mm8ln	CChCVCC	CVC CV CVC
33.	पुस्तक	p50stk	CVChCV	CVC CVC
34.	स्कूल	0sk6l	ChCVC	C CVC
35.	पटियाला	pt3y2l2	CCVCVCV	CV CV CV CV
36.	अनार	1n2r	VCVC	V CVC
37.	दिव्या	d3v0y2	CVChCV	CVC CV
38.	भोपाल	bh11p2l	CVCVC	CV CVC
39.	गुमराह	g5mr2h	CVCCVC	CVC CVC
40.	आदत	2dt	VCC	V CVC
41.	होशियारी	h11sh3y2r4	CVCVCVCV	CV CV CV CV
42.	चिडिया	ch3dd3y2	CVCVCV	CV CV CV
43.	रुड़की	r6ddk4	CVCCV	CVC CV
44.	खतरनाक	khtrn2k	CCCCVC	CV CVC CVC
45.	जलन्धर	Jl0ndhr	CCChCC	CV CVC CVC
46.	मसूरी	ms6r4	CCVCV	CV CV CV
47.	देहरादून	d7hr2d6n	CVCCVCVC	CVC CV CVC
48.	खुशमिजाज	kh5shm3j2j	CVCCVCVC	CVC CV CVC
49.	देखभाल	d7khbh2l	CVCCVC	CVC CVC
50.	थापर	th2pr	CVCC	CV CVC

The Sound classes are formed, *i.e*, the schwa is inserted in the phonetic string. Table 4.16 shows the correct phonetic transcription of the 50 Hindi words.

Table 4.16 Correct Phonetic Transcription

SNo	Word	Phonetic Transcription	Classification of Characters	Sound Classes	Correct Phonetic Transcription
1.	समारोह	sm2r11h	CCVCVC	CV CV CVC	s1 m2 r11 h
2.	खरगोश	khrg11sh	CCCVC	CVC CVC	kh1r g12sh
3.	गपशप	Gpshp	CCCC	CVC CVC	g1p sh1p
4.	किताब	k3t2b	CVCVC	CV CVC	k3 t2b
5.	यातायात	y2t2y2t	CVCVCVC	CV CV CVC	y2 t2 y2t
6.	नौजवानौ	n13jv2n13	CVCCVCV	CVC CV CV	n13j v2 n13
7.	गुलाब	g5l2b	CVCVC	CV CVC	g5 l2b
8.	कलम	Klm	CCC	CV CVC	k1 l1m
9.	अजायब	1j2yb	VCVCC	V CV CVC	1 j2 y1b
10.	कारीगर	k2r4gr	CVCVCC	CV CV CVC	k2 r4 g1r
11.	सरगम	Srgm	CCCC	CVC CVC	s1r g1m
12.	सरल	Srl	CCC	CV CVC	s1 r1l
13.	लड़की	lddk4	CCCV	CVC CV	l1dd k4
14.	लेखक	l7khk	CVCC	CV CVC	l7 kh1k
15.	पानी	p2n2	CVCV	CV CV	p2 n2
16.	कबूतर	kb6tr	CCVCC	CV CV CVC	k1 b6 t1r
17.	अतिथि	1t3th3	VCVCV	V CV CV	1 t3 th3
18.	रविवार	rv3v2r	CCVCVC	CV CV CVC	r1 v3 v2r
19.	सोमवार	s11mv2r	CVCCVC	CVC CVC	s12m v2r
20.	सफलता	sphlt2	CCCCV	CV CVC CV	s ph1l t2
21.	अभितेश	1bh3t7sh	CCVCVC	V CV CVC	1 bh3 t7sh
22.	पूजन	p6jn	CVCC	CV CVC	p6 j1n
23.	शनिवार	shn3v2r	CCVCVC	CV CV CVC	sh1 n3 v2r
24.	छिपकली	chh3pkl4	CVCCCV	CV CVC CV	chh3 p1k l4
25.	आकाश	2k2sh	VCVC	V CVC	2 k2sh
26.	विमान	v3m2n	CVCVC	CV CVC	v3 m2n
27.	कन्या	knHy2	CChCV	CVC CV	k1n y2
28.	सम्मान	s0mm2n	CChCVC	CVC CVC	s1m m2n

29.	नमस्कार	nm0sk2r	CCChCVC	CV CVC CVC	n m1s k2r
30.	मुस्कान	m50sk2n	CVChCVC	CVC CVC	m5s k2n
31.	पुरस्कार	p5r0sk2r	CVCChCVC	CV CVC CVC	p5 r1s k2r
32.	सम्मेलन	s0mm8ln	CChCVCC	CVC CV CVC	s1m m7 l1n
33.	पुस्तक	p50stk	CVChCV	CVC CVC	p5s t1k
34.	स्कूल	0sk6l	ChCVC	C CVC	s k6l
35.	पटियाला	pt3y2l2	CCVCVCV	CV CV CV CV	p1 t3 y2 l2
36.	अनार	1n2r	VCVC	V CVC	1 n2r
37.	दिव्या	d3v0y2	CVChCV	CVC CV	d3v y2
38.	भोपाल	bh11p2l	CVCVC	CV CVC	bh11 p2l
39.	गुमराह	g5mr2h	CVCCVC	CVC CVC	g5mr2h
40.	आदत	2dt	VCC	V CVC	2 d1t
41.	होशियारी	h11sh3y2r4	CVCVCVCV	CV CV CV CV	h11 sh3 y2 r4
42.	चिडिया	ch3dd3y2	CVCVCV	CV CV CV	ch3 dd3 y2
43.	रुड़की	r6ddk4	CVCCV	CVC CV	r6dd k4
44.	खतरनाक	khtrn2k	CCCCVC	CV CVC CVC	kh1 t1r n2k
45.	जलन्धर	Jl0ndhr	CCChCC	CV CVC CVC	j1 l1n dh1r
46.	मसूरी	ms6r4	CCVCV	CV CV CV	m1 s6 r4
47.	देहरादून	d7hr2d6n	CVCCVCVC	CVC CV CVC	d7h r2 d6n
48.	खुशमिजाज	kh5shm3j2j	CVCCVCVC	CVC CV CVC	kh5sh m3 j2j
49.	देखभाल	d7khbh2l	CVCCVC	CVC CVC	d7kh bh2l
50.	थापर	th2pr	CVCC	CV CVC	th2 p1r

4.1.2 Phonetic to Speech Module

The Phonetic-to-Speech module consists of a diphone concatenation algorithm and a database which contains all the sound files. The work of this module is to read the phonetic description, identify the appropriate diphones, concatenate them at pitch-marks, and play out the resulting signal.

A. Sound Database

It has a database of about 800 basic sounds, which are pitch-marked. All sound files stored in the database are gsm compressed .gsm files which are recorded at 16KHz as 16bit signed linear samples [10]. The database classifies and stores sound files in CV,

VC, V, Half and 0C files. The total size of the database is currently around 1MB. The sound database has the following architecture:
The following sound units are stored in the database (the number notation for vowels is same as in Table 4.2 and consonants are numbered from 1-34 sequentially).

CV pairs- 1..33 * 2 4 6 8 9 10 12 13 14 15
VC pairs- 2 4 6 8 9 10 12 13 14 15 * 1..34
V -1..14
33 ***0C*** sounds, all consonants except 'an'.

- CV files are named x.y.gsm where x is the consonant number and y is the vowel number.
- VC files are named x.y.gsm where x is the vowel number and y is the consonant number.
- V files are named x.gsm where x is the vowel number.
- Halfs files are named x.y.gsm where x,y are the two consonants involved.
- 0C files are named x.gsm where x is the consonant number.

All files other than the **0C** files have been pitch marked and the marks appear in the corresponding *.marks* files, one mark per byte as an unsigned char. Other than the the sound files, there are three additional files in *database/*, namely *cvoffsets, vcoffsets* and *voffsets*, which store various attributes of the sound files as shown in Table 4.17.

Table 4.17 Files in Dhvani Database

File Name	Fields Description
cvoffsets	CV fields: • start(start of the cv) • diphst(diphone start position: default halfway to ctov from start) • ctov(cons to vowel change position) • longvowlen(length of long vowel, currently not really used) • shortvowlen(length of short vowel) • diphend(end of diphone for long vowel, short will be obtained from long)

	• diphshortfactor(factor for getting short diphone from long) • halfst(place where this cv is cut to connect to previous half)
vcoffsets	VC fields: • end(end of vc) • diphend(diphone end position: default halfway from ctov to end) • vtoc(vowel to cons change position) longvowlen(length of long vowel, currently not really used) • shortvowlen(length of short vowel) • diphst(start of diphone for long vowel, short will be obtained from long
voffsets	V fields: • length (length to be played starting from 0)
hoffsets	Halfs fields: • start (start of half) end (place where this half is cut and appended to the next

B. Concatenation Algorithm

The concatenation algorithm of Dhvani with the help of pitch marks and the durational information selects the appropriate diphones and concatenate them for the final output. The algorithm also takes into account the characteristics of certain consonantal sound to induce a delay or a gap in certain contexts and inter-word silences.

For example, after obtaining the correct phonetic transcription of the word "मुबारक", the concatenation algorithm selects the associated diphones files as shown in Table 4.18 and concatenates them.

Table 4.18 Diphone Concatenation in "मबारक" Pronunciation

Sound Class	Phonetic Transcription	Concatenation Algorithm	Word Pronounced
CV	m5	It concatenates 0C file 23.gsm and V file 5.gsm	मु
CV	b2	It directly selects the CV pair 21.2.gsm	बा
CVC	r1k	It concatenates the 0C file 25.gsm, V file 1.gsm and 0C file 1.gsm	रक्

4.2 Features of New Proposed Hindi TTS

The new proposed TTS can convert text type in Hindi to speech. It reads the text in orthographic form ('Akshara'). It has been developed in java netbeans 6.8. This system can run on OS like WindowXp, Windows7, Linux etc.

4.2.1 Architecture of New TTS Module

The architecture of the new text to speech module is shown in figure 4.5. It follows the following steps.

- First the input text is converted into tokens.
- Each token's characters are classified as consonants, vowels and Halant.
- Each token is then converted into its phonetic transcription.
- After this the sound classes and correct phonetic transcription are produced using the algorithm discussed in previous sections.
- At this step the appropriate diphones are identified, concatenated and played out.

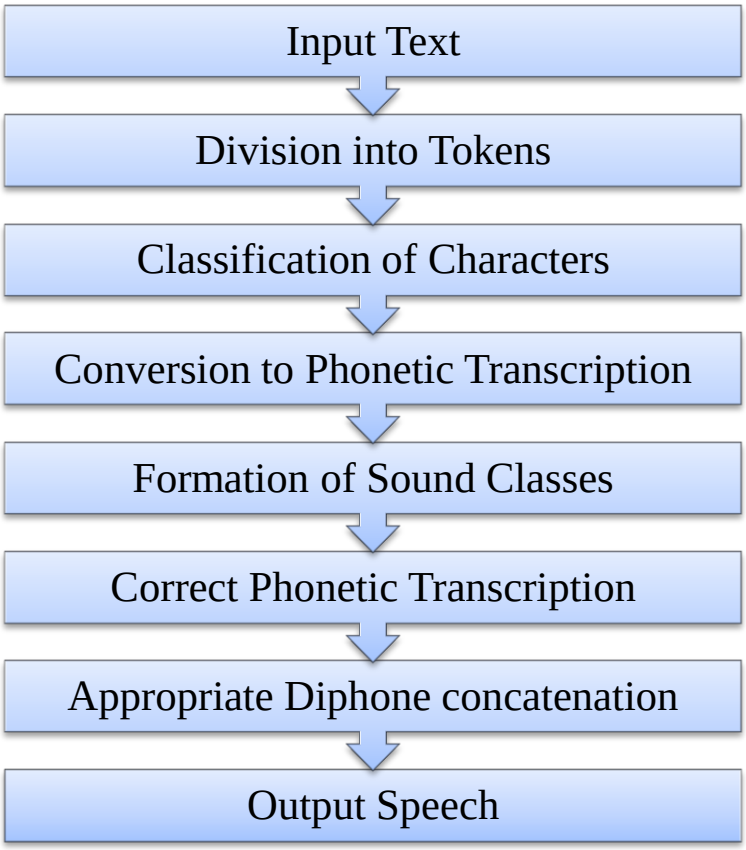


Figure 4.5 Architecture of New Text to Speech Module

4.2.2 Voice Database of New TTS Module

In this system female voice is recorded in the form of diphones using the Audacity tool. In the database voices are recorded as 'CV','V','VC' and '0C' (half-consonant). The 30 consonants and 10 vowels (refer table 4.4) are stored as 'C' and 'V' form files. There are 30*10 'CV' pairs, 10*30 'VC' pairs, 30 half sounds and 10 vowels are stored. Table 4.19 shows the names of CV, V and 0C files stored in *.wav* format.

Audacity Tool

Audacity is a free, easy-to-use and multilingual audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems. The snapshot of Audacity tool is shown in figure 4.1.

Table 4.19 ‘CV’, ‘0C’ and ‘V’ Pairs Files in Database

V / C	Half-consonant (0C)	अ	ा	ि	ी	ु	ू	े	ै	ो	ौ
		1	2	3	4	5	6	7	8	9	10
क	k	k1	k2	k3	k4	k5	k6	k7	k8	k9	k10
ख	kh	kh1	kh2	kh3	kh4	kh5	kh6	kh7	kh8	kh9	kh10
ग	g	g1	g2	g3	g4	g5	g6	g7	g8	g9	g10
घ	gh	gh1	gh2	gh3	gh4	gh5	gh6	gh7	gh8	gh9	gh10
च	ch	ch1	ch2	ch3	ch4	ch5	ch6	ch7	ch8	ch9	ch10
छ	chh	chh1	chh2	chh3	chh4	chh5	chh6	chh7	chh8	chh9	chh10
ज	j	j1	j2	j3	j4	j5	j6	j7	j8	j9	j10
झ	jh	jh1	jh2	jh3	jh4	jh5	jh6	jh7	jh8	jh9	jh10
त	t	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
थ	th	th1	th2	th3	th4	th5	th6	th7	th8	th9	th10
द	d	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
ध	dh	dh1	dh2	dh3	dh4	dh5	dh6	dh7	dh8	dh9	dh10
न	n	n1	n2	n3	n4	n5	n6	n7	n8	n9	n10
ट	tt	tt1	tt2	tt3	tt4	tt5	tt6	tt7	tt8	tt9	tt10
ठ	tth	tth1	tth2	tth3	tth4	tth5	tth6	tth7	tth8	tth9	tth10
ड	dd	dd1	dd2	dd3	dd4	dd5	dd6	dd7	dd8	dd9	dd10
ढ	ddh	ddh1	ddh2	ddh3	ddh4	ddh5	ddh6	ddh7	ddh8	ddh9	ddh10
ण	nna	nna1	nna2	nna3	nna4	nna5	nna6	nna7	nna8	nna9	nna10
प	p	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10
फ	ph	ph1	ph2	ph3	ph4	ph5	ph6	ph7	ph8	ph9	ph10
ब	b	b1	b2	b3	b4	b5	b6	b7	b8	b9	b10
भ	bh	bh1	bh2	bh3	bh4	bh5	bh6	bh7	bh8	bh9	bh10
म	m	m1	m2	m3	m4	m5	m6	m7	m8	m9	m10
य	y	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10
र	r	r1	r2	r3	r4	r5	r6	r7	r8	r9	r10
ल	l	l1	l2	l3	l4	l5	l6	l7	l8	l9	l10
व	v	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10
श	sh	sh1	sh2	sh3	sh4	sh5	sh6	sh7	sh8	sh9	sh10
स	s	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
ह	h	h1	h2	h3	h4	h5	h6	h7	h8	h9	h10

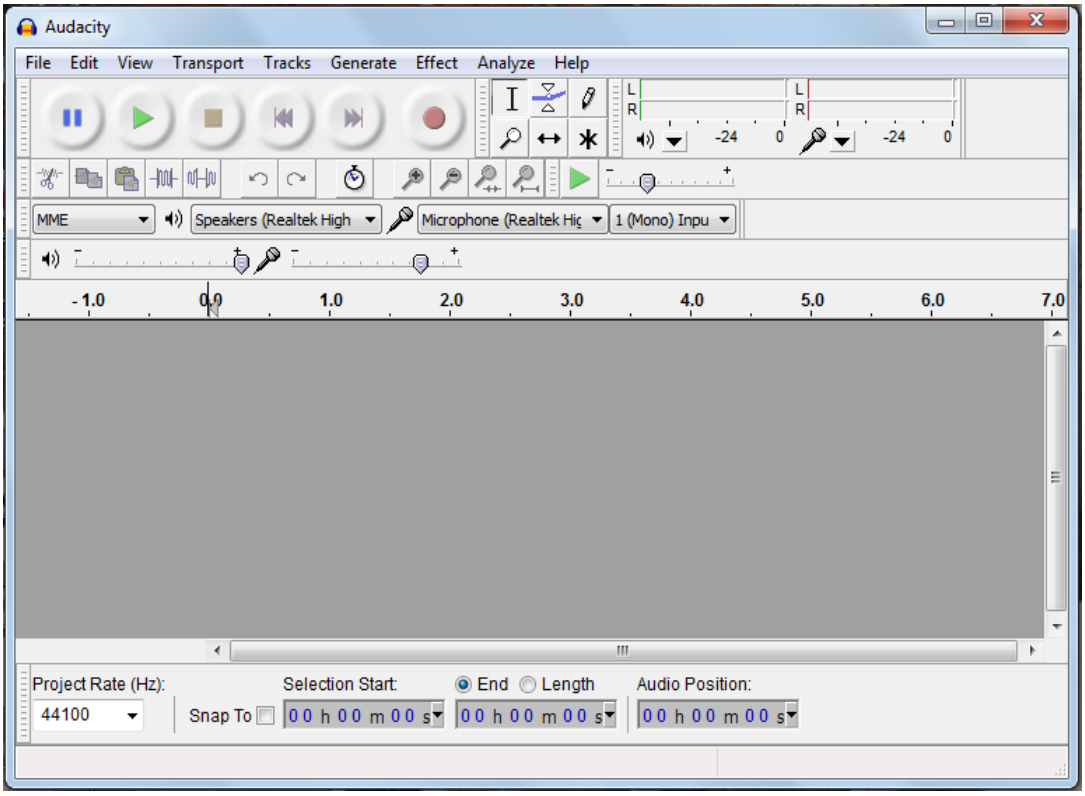


Figure 4.2 Snapshot of Audacity Tool

The sound is recorded in the form of diphones as wave files. The snapshots of wave files of 'क', 'का', 'कि', 'की' is shown below.

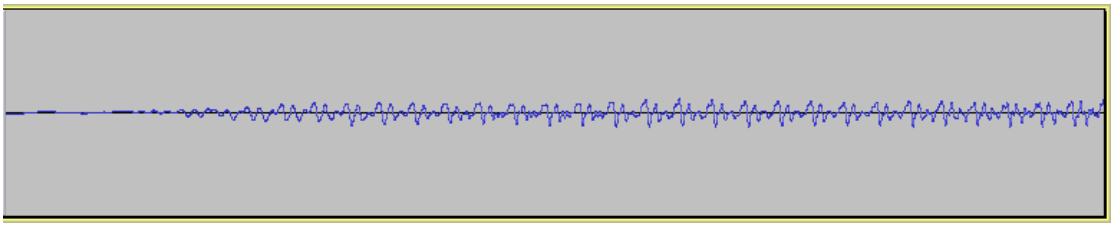


Figure 4.3 Wave File 'k1'

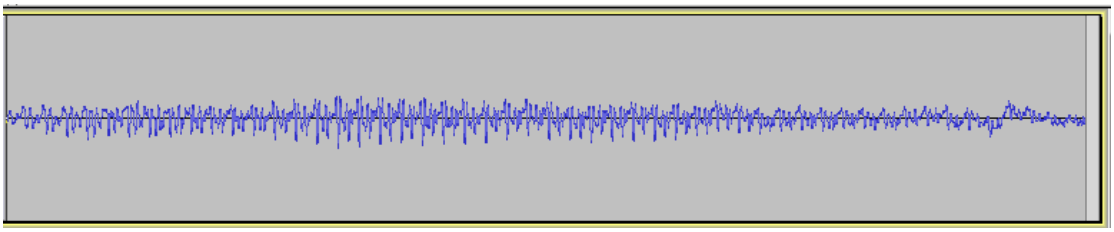


Figure 4.4 Wave File 'k2'

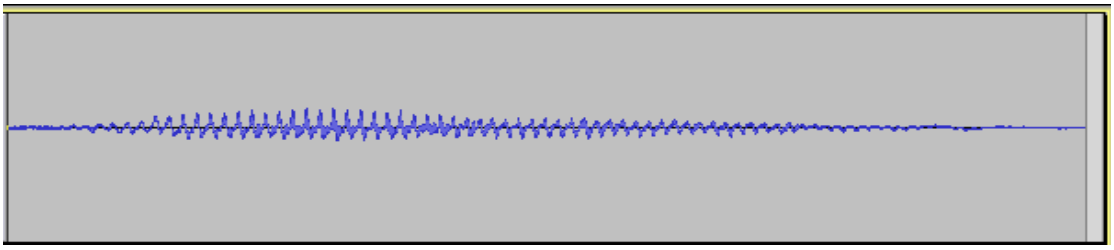


Figure 4.5 Wave File ‘k3’

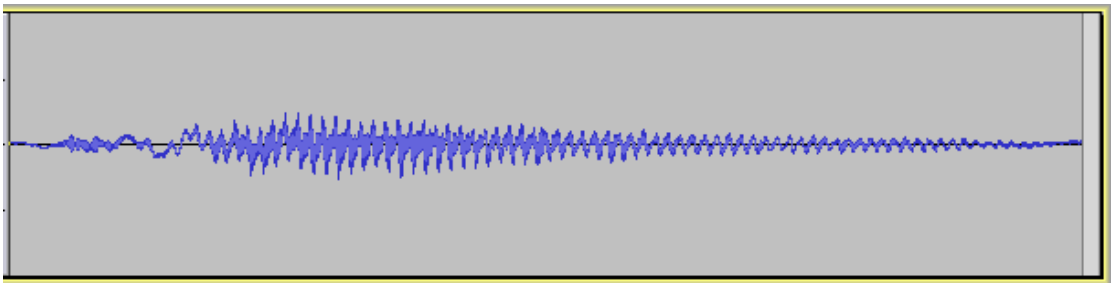


Figure 4.6 Wave File for ‘k4’

Chapter 5

Testing and Results

5.1 Testing of User Friendly Hindi Text-to-Speech System

The User-Friendly Interface for Hindi text to speech conversion has been tested for its various functionalities. The testing results for the various tests performed on it are shown in further sections of this chapter. The first screen shot at the start of the system is shown in figure 5.1. The screenshot shows the textbox to enter the text, keyboard and various other buttons.

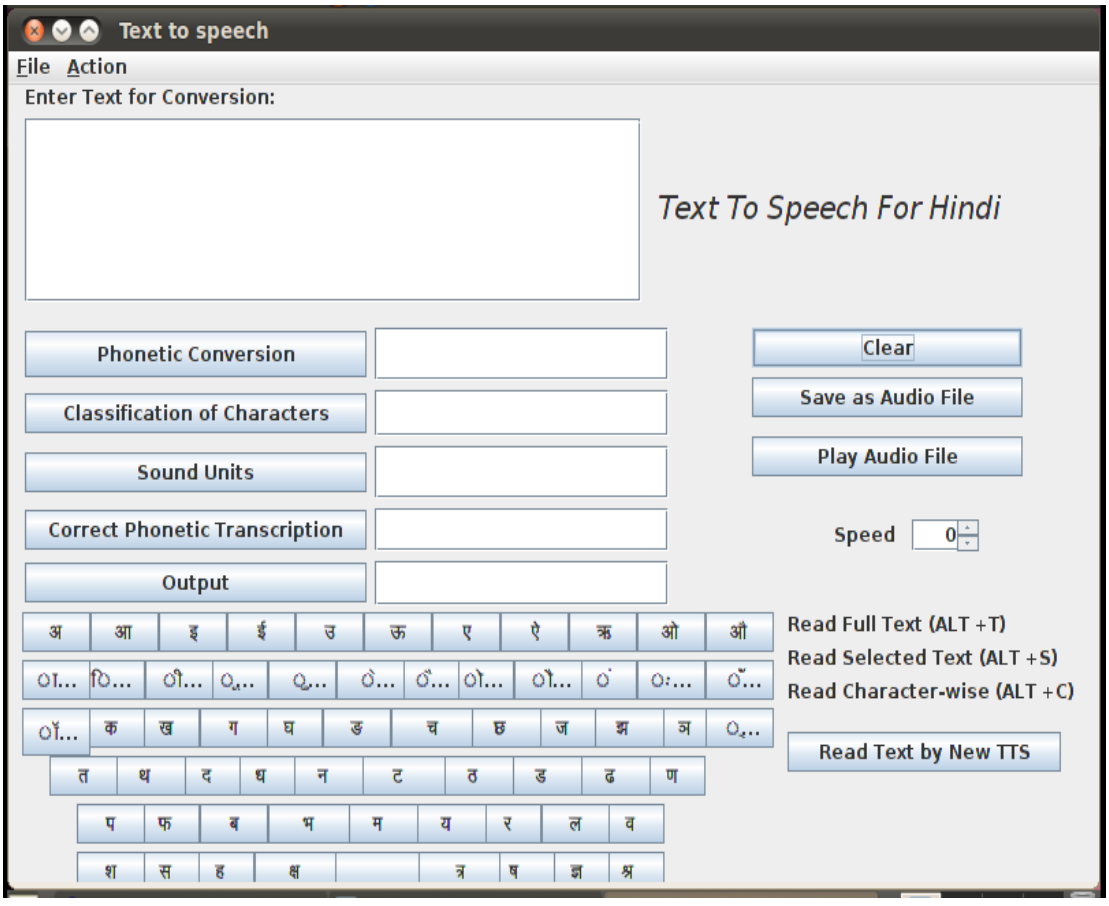


Figure 5.1: Screenshot of User Friendly Hindi Text to Speech System

A. Entering Hindi Text into the System

There are two ways through which text can be entered into the system.

Through Keyboard:

To test the working of the keyboard, text is entered at the textbox through the keyboard. The screenshot of entered Hindi text is shown in figure 5.2.

Through File:

The other method of entering a text is through file. You can select and open the file whose text you want to read. This can be done by selecting the **Open** menu from the **File** menubar. The whole text of that file will be displayed at the textbox. The screenshot of opening a file is shown in figure 5.3.

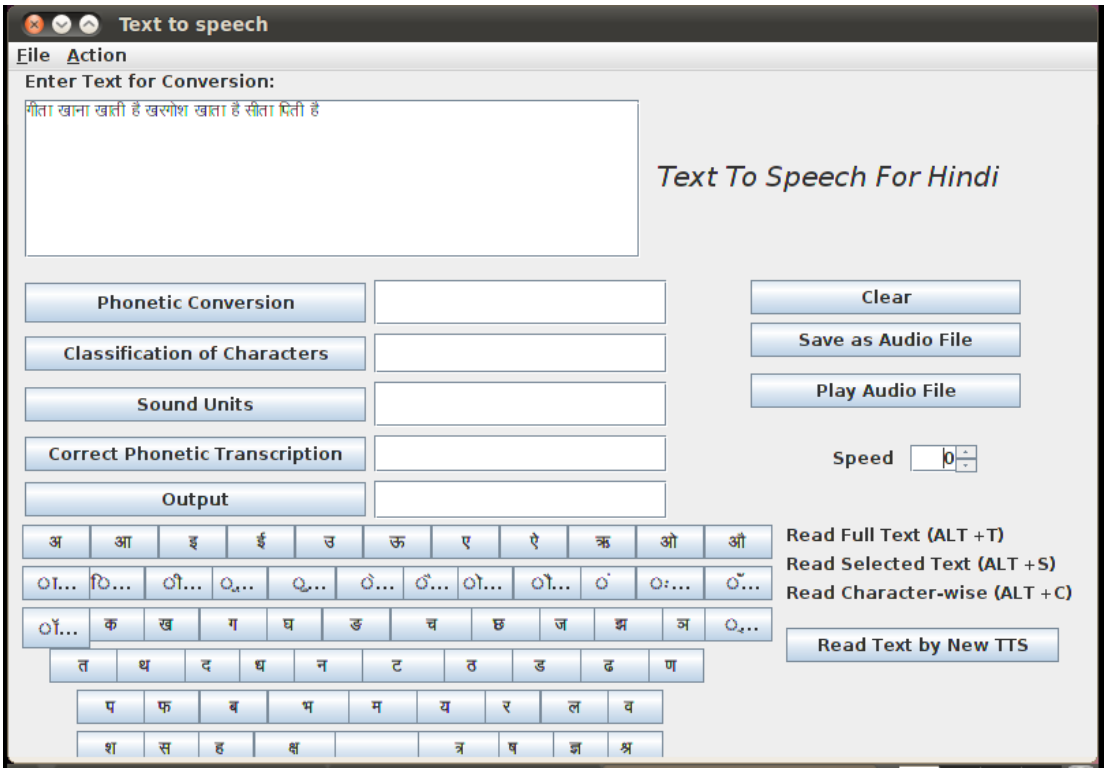


Figure 5.2: Screenshot of Entering Text through Keyboard

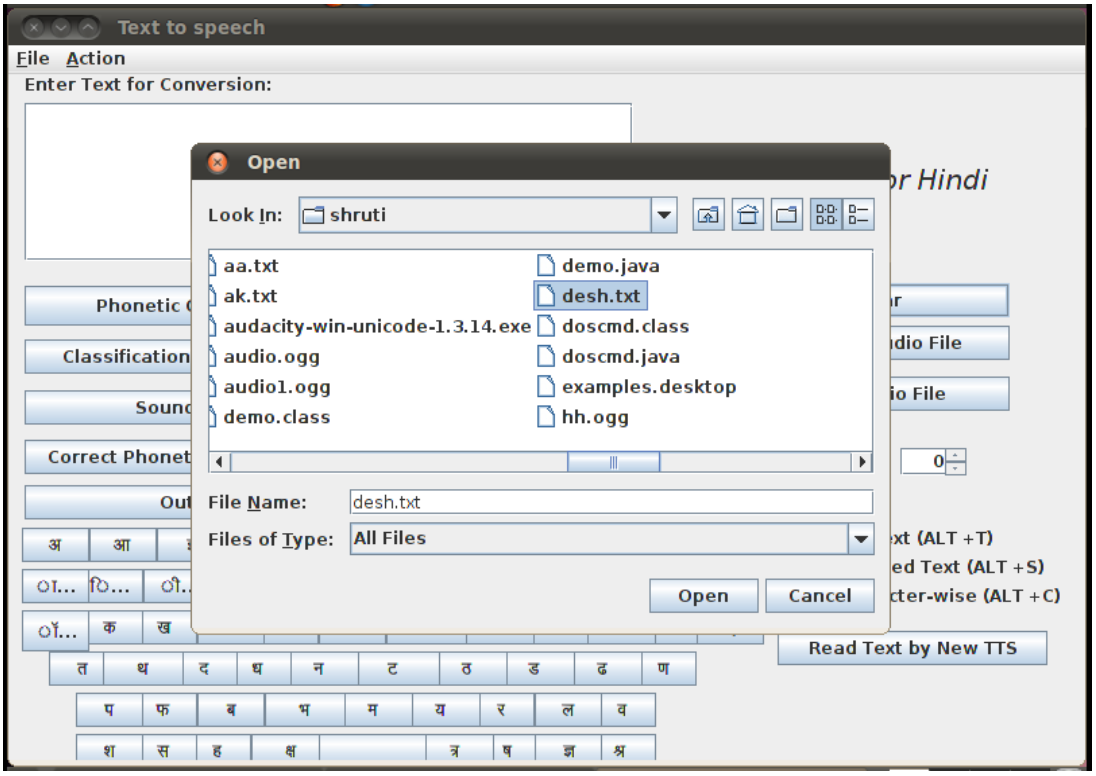


Figure 5.3: Screenshot of Opening a File

B. Conversion of Hindi Text to Speech.

There are three options present to read out the entered text. These options are given in the **Action** menu bar.

Read Full Text:

This option reads the full text written in the textbox. The shortcut for this option is ALT + T. Its screen shot is shown in figure 5.4.

Read Selected Text:

This option reads only the selected text by the user in the textbox. The shortcut for this option is ALT + S. Its screen shot is shown in figure 5.5.

Read Selected Text Characterwise:

This option reads the selected text by the user character wise. The shortcut for this option is ALT + C. Its screen shot is shown in figure 5.6.

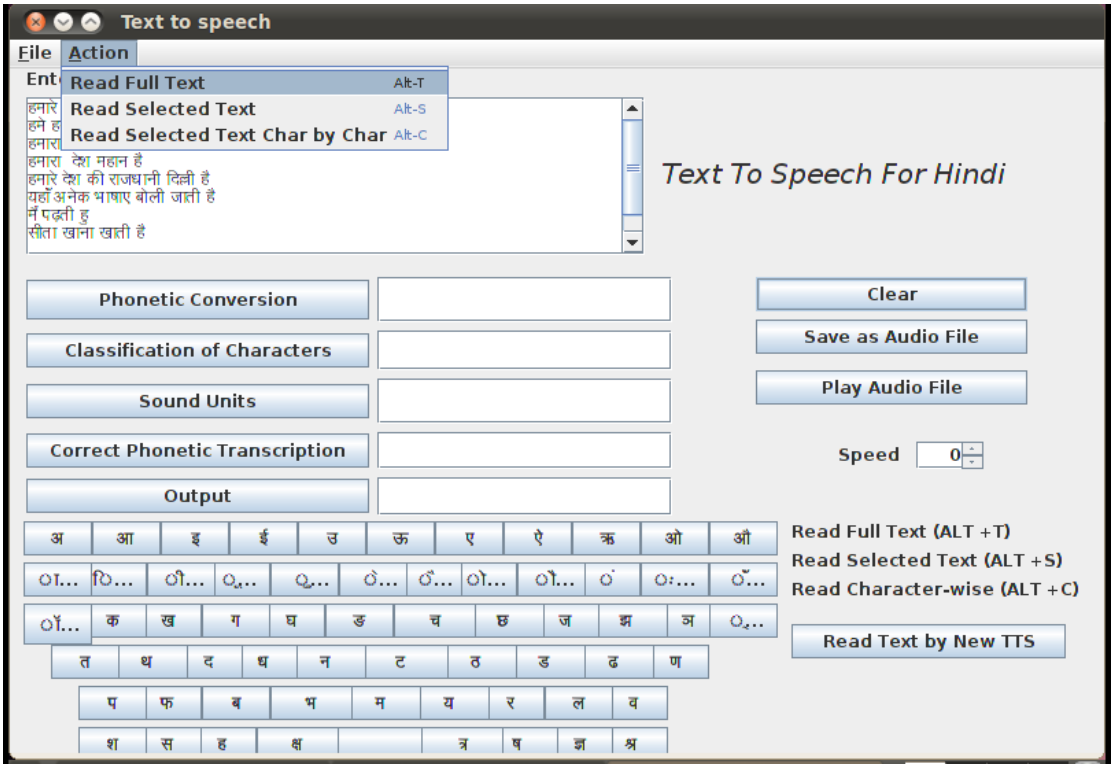


Figure 5.4: Screenshot of Read Full Text

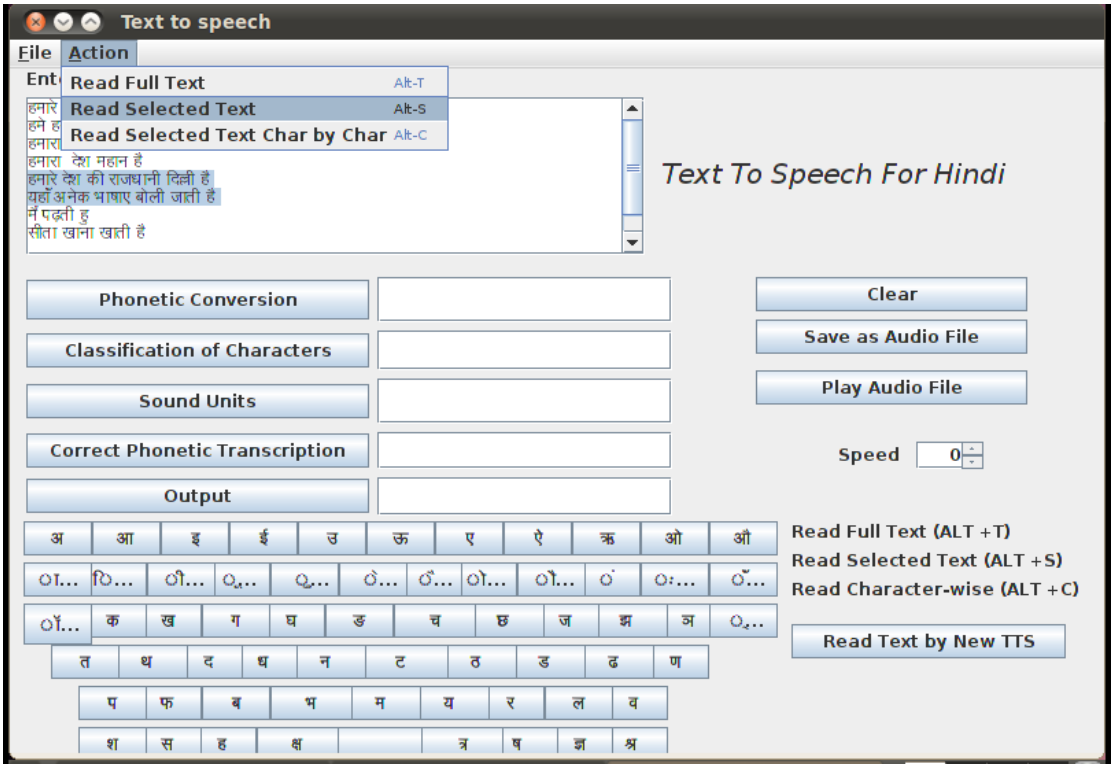


Figure 5.5: Screenshot of Read Selected Text

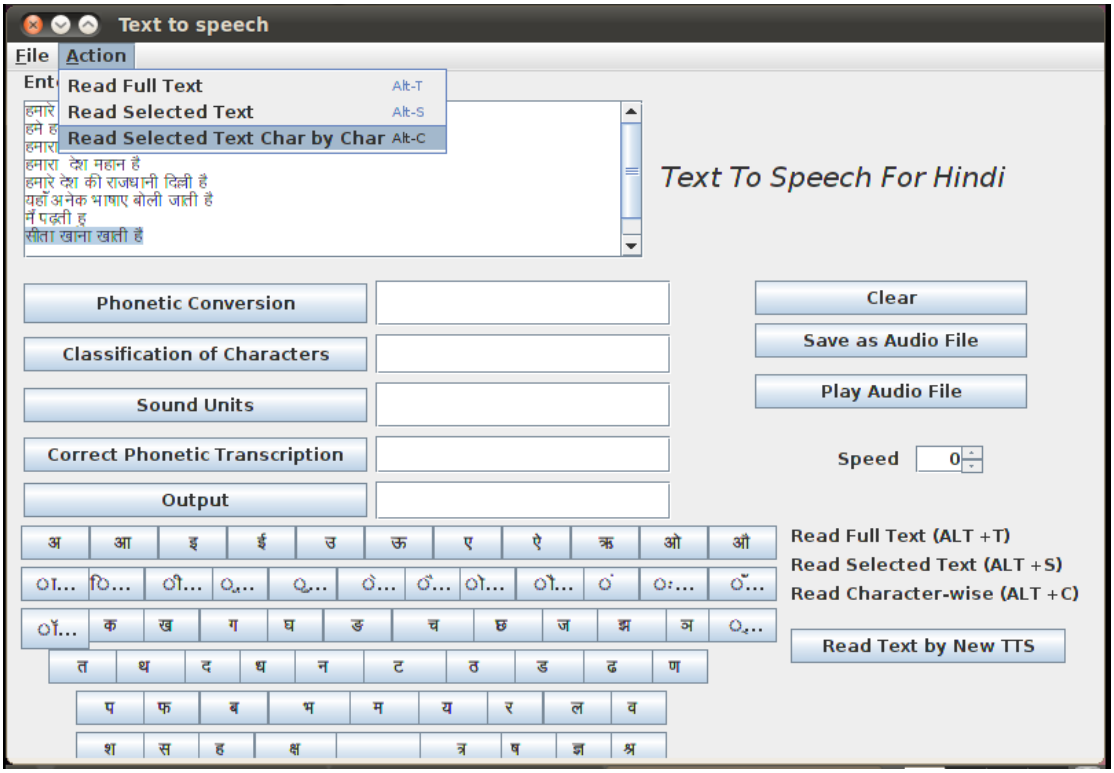


Figure 5.6: Screenshot of Read Selected Text Characterwise

C. Making of Audio File

This system also provides the facility to make the audio file of the speech of the entered Hindi text.

Save as Audio File:

This button is used to make an audio file of the Hindi text in present in the textbox. The click on this button will display a small window asking the name from which the audio file should be saved. The audio file must be of .ogg extension. Its screen shot is shown in figure 5.7.

Play Audio File:

This button is used to play an already made audio file. The click on this button will allow you to browse the audio file which you want to play. Its screen shot is shown in figure 5.8.

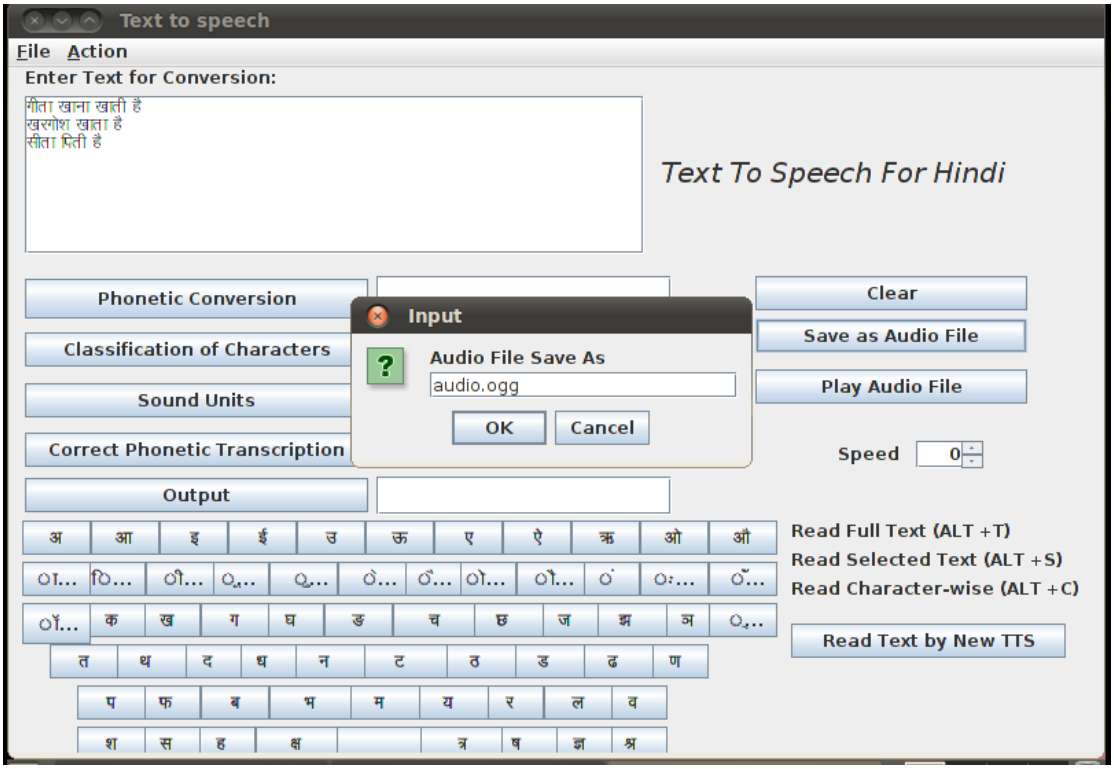


Figure 5.7: Screenshot of Saving the Audio File

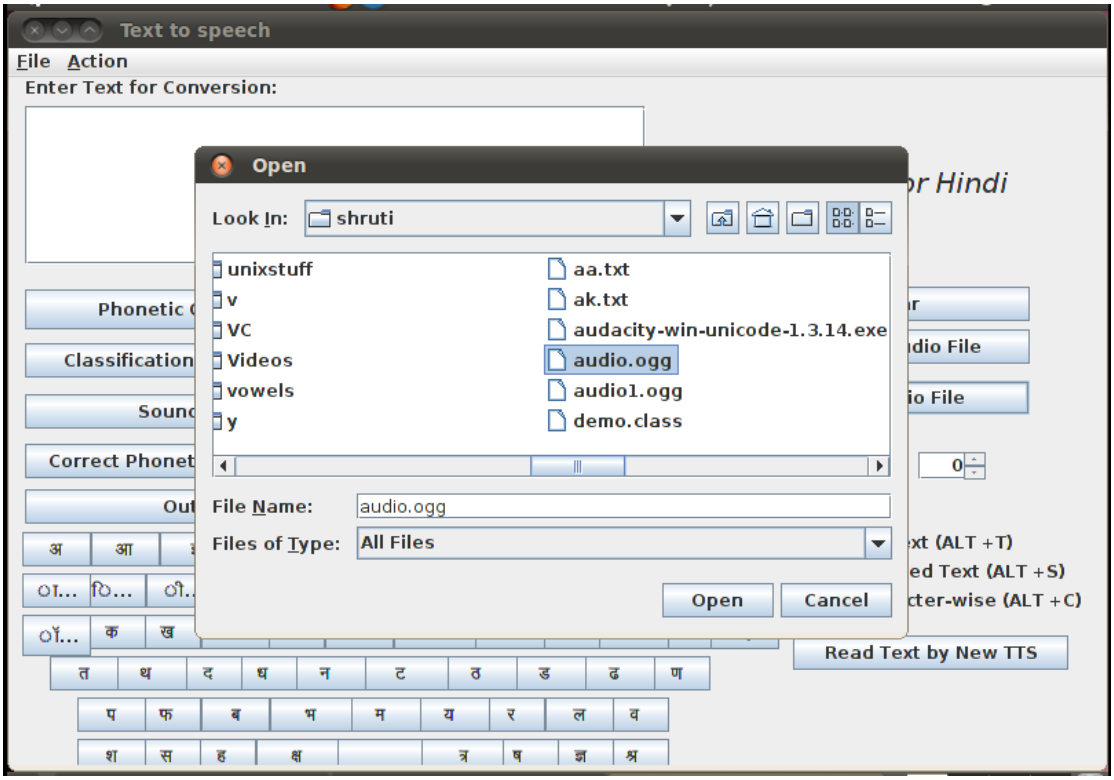


Figure 5.8: Screenshot of Opening the Audio File for Playing

D. Display of Intermediate Stages

There are four buttons present to show the intermediate stages which an input text token passes through before converting into the speech.

Phonetic Conversion:

This button shows the phonetic conversion of the input text. For example if the word is '**अनुशासन**', then its phonetic conversion will be "1n5sh2sn". Its screen shot is shown in figure 5.9.

Classification of Characters:

This button shows the character classification as Consonants (C), Vowels (V) and Half-Consonants (H or Ch) of the input text. For example if the word is '**अनुशासन**', then its characters will be classified as "VCVCVCC". Its screen shot is shown in figure 5.10.

Sound Units:

This button shows correct sound units of the input text. For example if the word is '**अनुशासन**', then its sound units will be "V CV CV CVC". Its screen shot is shown in figure 5.11.

Correct Phonetic Transcription:

This button shows correct phonetic transcription of the input text. For example if the word is '**अनुशासन**', then its correct phonetic transcription will be "1 n5 sh2 s1n". Its screen shot is shown in figure 5.12.

Output:

This button shows the final output. The output of the word '**अनुशासन**' will be '**अ नु शा सन्**'. Its screen shot is shown in figure 5.13.

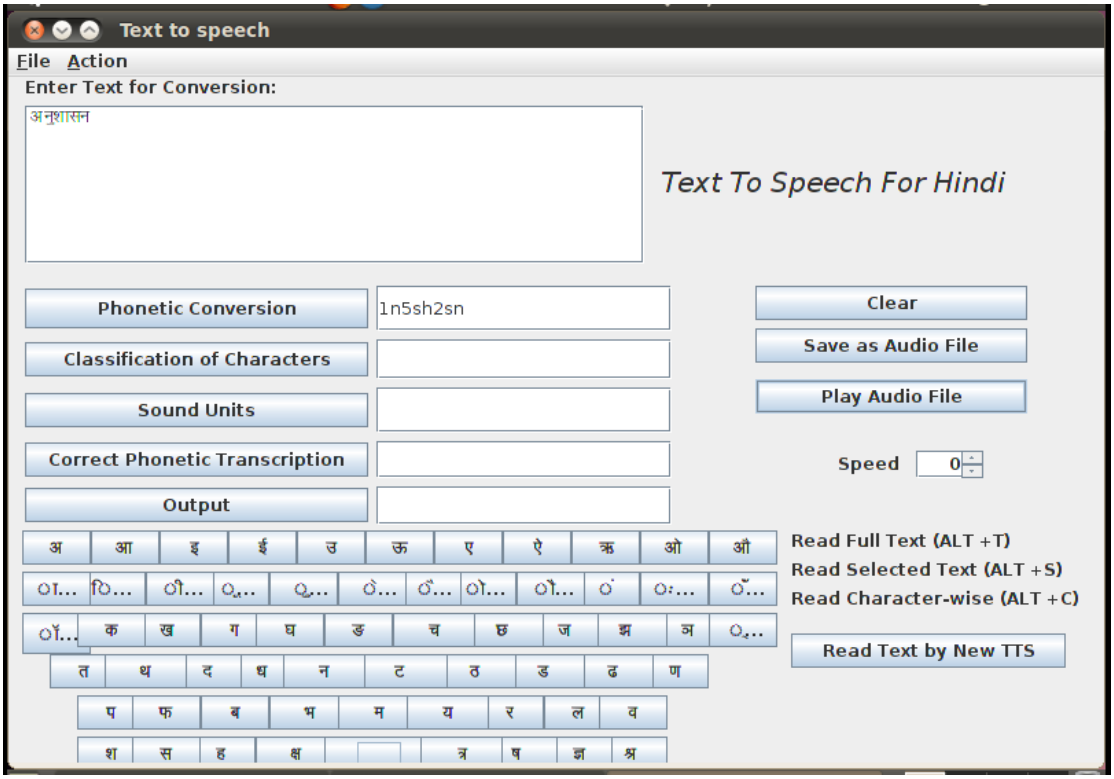


Figure 5.9: Screenshot Displaying Phonetic Conversion

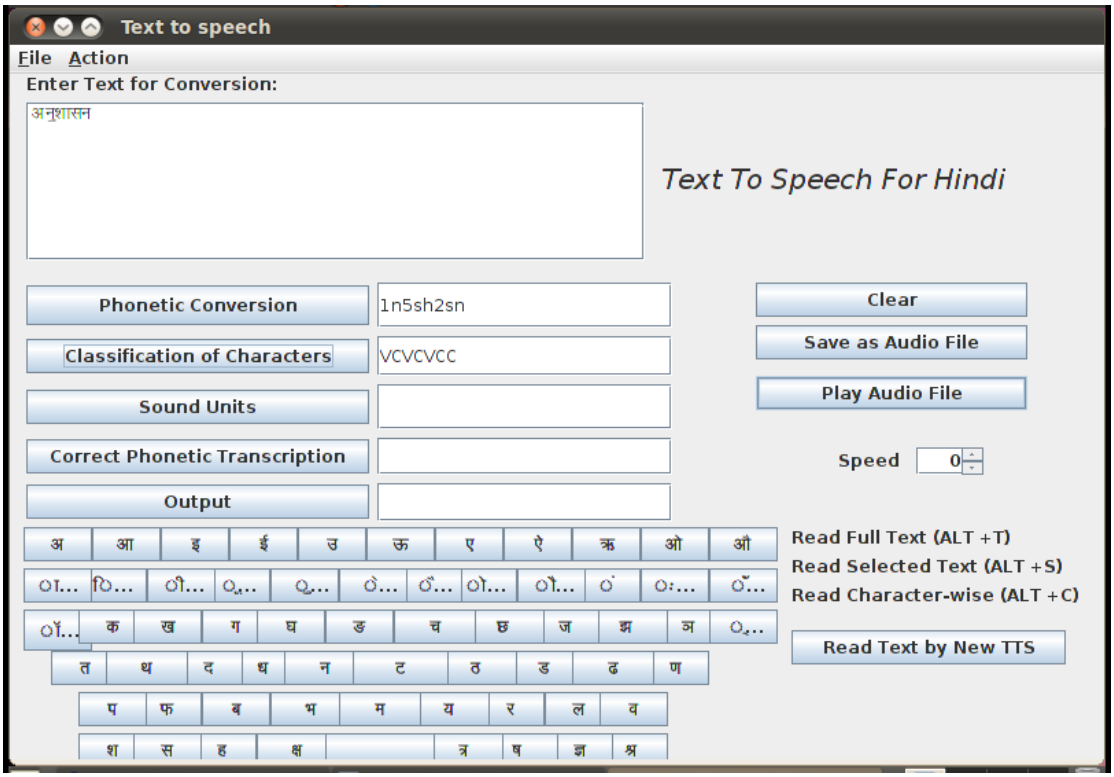


Figure 5.10: Screenshot Displaying Classification of Characters

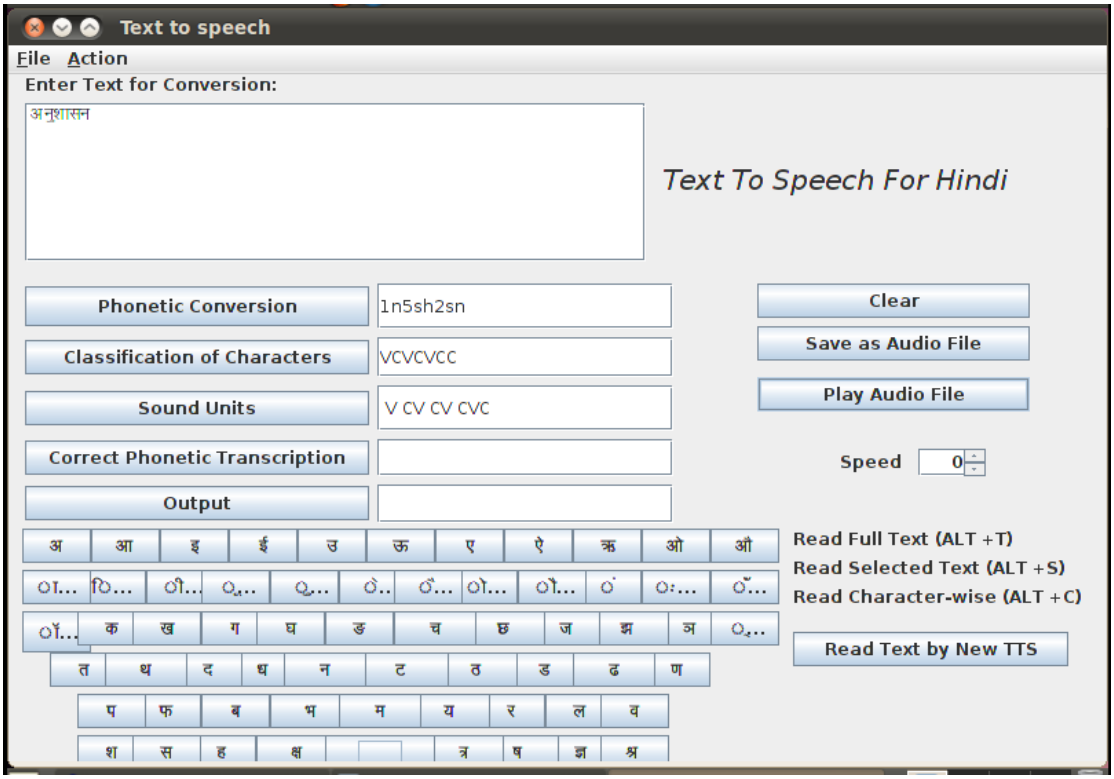


Figure 5.11: Screenshot Displaying Sound Units

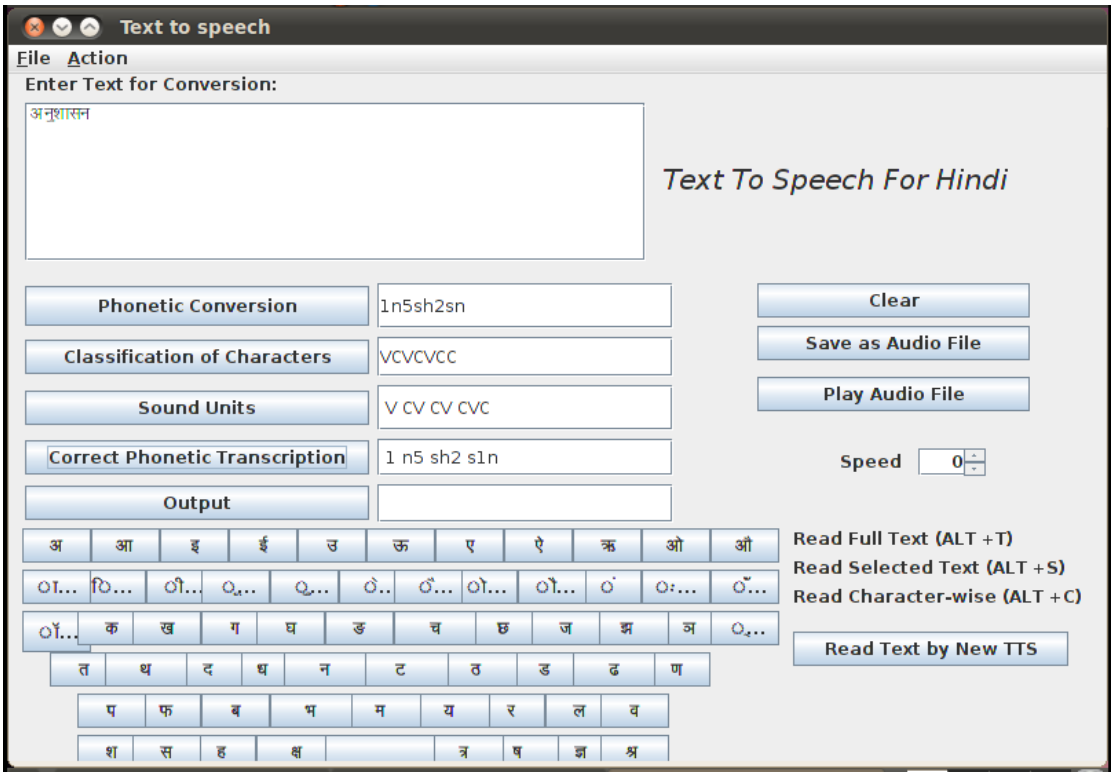


Figure 5.12: Screenshot Displaying Correct Phonetic Transcription

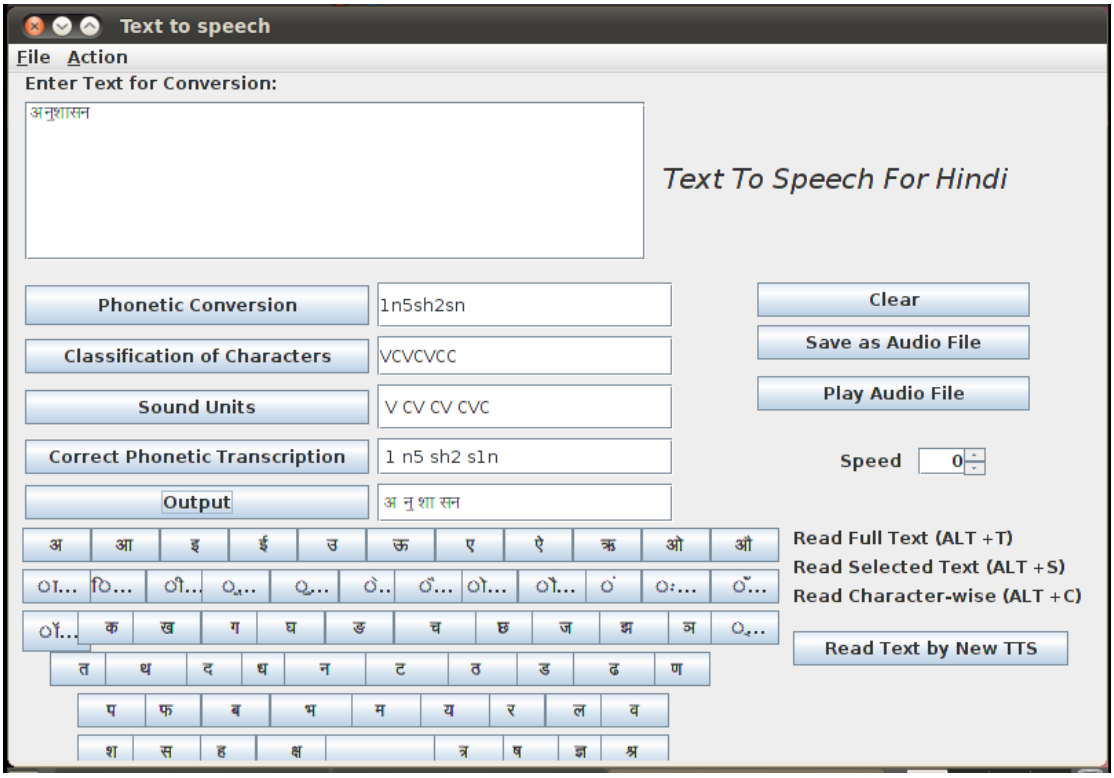


Figure 5.13: Screenshot Displaying the Final Output

Hence for verifying the system correctness the functionalities of various other buttons including Clear, Exit and Speed is also tested. The new proposed system is also tested by giving text into the system.To analyse the intermediate stages, 50 Hindi words are tested at the User-Friendly interface and the results are shown in Table 5.1.

.

Table 5.1 Testing of Sample 50 Hindi Words in Designed System

SNo	Word	Phonetic Transcription	Classification of Characters	Sound Classes	Correct Phonetic Transcription	Output
1.	समारोह	sm2r11h	CCVCVC	CV CV CVC	s1 m2 r11 h	स् मा रोह्
2.	खरगोश	khrg11sh	CCCVC	CVC CVC	kh1r g11sh	खर् गोश्
3.	गपशप	gpshp	CCCC	CVC CVC	g1p sh1p	गप् शप्
4.	किताब	k3t2b	CVCVC	CV CVC	k3 t2b	कि ताब्
5.	यातायात	y2t2y2t	CVCVCVC	CV CV CVC	y2 t2 y2t	या ता यात्
6.	नौजवानौ	n13jv2n13	CVCCVCV	CVC CV CV	n13j v2 n13	नौज् वा नौ
7.	गुलाब	g5l2b	CVCVC	CV CVC	g5 l2b	गु लाब्
8.	कलम	klm	CCC	CV CVC	k1 l1m	क लम्
9.	अजायब	1j2yb	VCVCC	V CV CVC	1 j2 y1b	अ जा यब्
10.	कारीगर	k2r4gr	CVCVCC	CV CV CVC	k2 r4 g1r	का री गर्
11.	सरगम	srgm	CCCC	CVC CVC	s1r g1m	सर् गम्
12.	सरल	srl	CCC	CV CVC	s1 r1l	स रल्
13.	लड़की	lddk4	CCCV	CVC CV	l1dd k4	लड़ की
14.	लेखक	l7khk	CVCC	CV CVC	l7 kh1k	ले खक्
15.	पानी	p2n2	CVCV	CV CV	p2 n2	पा नी
16.	कबूतर	kb6tr	CCVCC	CV CV CVC	k1 b6 t1r	क बू तर्
17.	अतिथि	1t3th3	VCVCV	V CV CV	1 t3 th3	अ ति थि
18.	रविवार	rv3v2r	CCVCVC	CV CV CVC	r1 v3 v2r	र वि वार्

19.	सोमवार	s11mv2r	CVCCVC	CVC CVC	s11m v2r	सोम् वार्
20.	सफलता	sphlt2	CCCCV	CV CVC CV	s1 ph1l t2	स फल् ता
21.	अभितेश	1bh3t7sh	CCVCVC	V CV CVC	1 bh3 t7sh	अ भि तेश्
22.	पूजन	p6jn	CVCC	CV CVC	p6 j1n	पू जन्
23.	शनिवार	shn3v2r	CCVCVC	CV CV CVC	sh1 n3 v2r	श नि वार्
24.	छिपकली	chh3pkl4	CVCCCV	CV CVC CV	chh3 p1k l4	छि पक् ली
25.	आकाश	2k2sh	VCVC	V CVC	2 k2sh	आ काश्
26.	विमान	v3m2n	CVCVC	CV CVC	v3 m2n	वि मान्
27.	कन्या	knHy2	CChCV	CVC CV	k1n y2	कन् या
28.	सम्मान	s0mm2n	CChCVC	CVC CVC	s1m m2n	सम् मान्
29.	नमस्कार	nm0sk2r	CCChCVC	CV CVC CVC	n1 m1s k2r	न मस् कार्
30.	मुस्कान	m50sk2n	CVChCVC	CVC CVC	m5s k2n	मुस् कान्
31.	पुरस्कार	p5r0sk2r	CVCChCVC	CV CVC CVC	p5 r1s k2r	पु रस् कार्
32.	सम्मेलन	s0mm7ln	CChCVCC	CVC CV CVC	s1m m7 l1n	सम् मे लन्
33.	पुस्तक	p50stk	CVChCV	CVC CVC	p5s t1k	पुस् तक
34.	स्कूल	0sk6l	ChCVC	C CVC	0s k6l	स् कूल्
35.	पटियाला	pt3y2l2	CCVCVCV	CV CV CV CV	p1 t3 y2 l2	प टि या ला
36.	अनार	1n2r	VCVC	V CVC	1 n2r	अ नार्
37.	दिव्या	d3v0y2	CVChCV	CVC CV	d3v y2	दिव् या
38.	भोपाल	bh11p2l	CVCVC	CV CVC	bh11 p2l	भो पाल्
39.	गुमराह	g5mr2h	CVCCVC	CVC CVC	g5m r2h	गुम राह्

40.	आदत	2dt	VCC	V CVC	2 d1t	आ दत्
41.	होशियारी	h11sh3y2r4	CVCVCVCV	CV CV CV CV	h11 sh3 y2 r4	हो शि या री
42.	चिडिया	ch3dd3y2	CVCVCV	CV CV CV	ch3 dd3 y2	चि डि या
43.	रुड़की	r6ddk4	CVCCV	CVC CV	r6dd k4	रुड़ की
44.	खतरनाक	khtrn2k	CCCCVC	CV CVC CVC	kh1 t1r n2k	ख तर् नाक्
45.	जलन्धर	jl0ndhr	CCChCC	CV CVC CVC	j1 l1n dh1r	ज लन् धर्
46.	मसूरी	ms6r4	CCVCV	CV CV CV	m1 s6 r4	म सू री
47.	देहरादून	d7hr2d6n	CVCCVCVC	CVC CV CVC	d7h r2 d6n	देह् रा दून्
48.	खुशमिजाज	kh5shm3j2j	CVCCVCVC	CVC CV CVC	kh5sh m3 j2j	खुश् मि जाज्
49.	देखभाल	d7khbh2l	CVCCVC	CVC CVC	d7kh bh2l	देख् भाल्
50.	थापर	th2pr	CVCC	CV CVC	th2 p1r	था पर्

Chapter 6

Conclusion and Future scope

6.1 Conclusion

This thesis shows the study about the Devnagri Script and the symbols used in it. The full process of converting text to speech is analyzed and various methods used for storing sound and generating voice is studied. The survey on various text to speech systems available for different languages has been performed and these systems are compared on the basis of languages it support, size of the speech units, methodologies used to code, store and synthesize the speech and prosody.

A User-Friendly Interface for Hindi Text to Speech conversion has been developed in which ***Dhvani*** is used as a back end tool. It consists of keyboard of Hindi characters so that the user can easily type the text in Hindi in the text box and a speed modifier to modify the speed of the speech. It also provides the facility to save the speech file of the input text and can also play any of the previously saved audio file. Various intermediate stages namely, phonetic conversion, classification of characters, sound units formed and correct phonetic transcription for converting Hindi text to speech is analyzed and shown in the interface. After analysing various features of ***Dhvani***, some of its limitations like it works only in Linux environment and has no female voice database has come across.

To overcome the limitations a new Hindi text to speech module is proposed which can run on OS like WindowXp, Windows7, Linux *etc*. It can convert the text type in Hindi to speech. It reads the text in orthographic form ('Akshara'). It follows the method of diphone concatenation and has a female voice database with diphone as the storage unit. The diphones sounds are recorded using the Audacity tool.

6.2 Future Scope

All Indian language scripts have common phonetic base. There is more or less one to one correspondence between what is written and what is spoken. The rules required to

map the letters to sounds of Indian languages are almost straight forward. Hence, a common user friendly TTS can be supposed for all Indian languages which can run on various operating systems. At present, with inadequate prosodic models in place, the quality of synthetic speech generated by the synthesizers is poor. So efforts can be done for the development of prosodic models. The further work can be done to improve the naturalness and intelligibility of TTS. A Web based application can also be designed which can convert text in any Indian languages into speech.

References

[1] Agarwal S. S, "Development of Resources and Techniques for Processing of some Indian Languages" (February 2012) [Online]. Available: http://projects.ldc.upenn.edu/LDC_Institute/Visitors/Agrawal.ppt

[2] Basu A, Sen D , Sen S and Chakraborty S , "An Indian Language Speech Synthesizer – Techniques and Applications", Proceedings of national systems conference, NSC 2003, Indian Institute of Technology, Kharagpur, India, 2003, pp. 217-223.

[3] Chaudhury P, Rao M, Kumar K. V, "Symbol based concatenation approach for Text to Speech System for Hindi using vowel classification technique", in World Congress on Nature and biologically Inspired computing ,2009, pp.1082 – 1087.

[4]Chauhan A, Chauhan V, Singh S. P, Tomar A. K, Chauhan H, "A Text to Speech System for Hindi using English Language" ,in International Journal of Computer Science and Technology, Vol. 2, Issue 3, September 2011, pp. 322-326.

[5] Choudhury M, "Rule Based Grapheme to Phoneme Mapping for Hindi Speech Synthesis", in 90th Indian Science Congress of the International Speech Communication Association (ISCA), Bangalore, India, 2003.

[6] Duggan B, Deegan M, "Considerations In The Usage Of Text To Speech (TTS) In The Creation Of Natural Sounding Voice Enabled Web Systems", ISICT '03 Proceedings of the 1st international symposium on Information and communication technologies, 2003, pp 433 - 438.

[7] Hariharan R (February 2012) [Online]. Available: http://dhvani.sourceforge.net/

[8] Hariharan R , "Simputer- Radical simplicity for Universal Access" (February 2012) [Online]. Available: http://www.simputer.org/simputer/downloads/software/dhvani/tech.php

[9] Hariharan R, Bali K, Manohar S, Vinay V, Vivek K. S, “Language Technology Solutions in Simputer: an Overview” in-49-Proceedings of the Language Engineering Conference (LEC’02), IEEE Computer Society, pp 189-196.

[10] Hariharan R, Thottingal S, “Dhvani Foss Community India” (February 2012) [Online]. Available: http://fci.wikia.com/wiki/Dhvani

[11] Hunt A. J, Black A. W, “Unit selection in a concatenative speech synthesis system using a large speech database”, in IEEE international Conference on Accoustics, Speech and Signal Processing Atlanta, GA , USA , vol 1, 1996, pp. 373 – 376.

[12] Jain H (February2012) [Online]. Available: http://www.cse.iitb.ac.in/vani/

[13] Jain H, Kanade V, Sivakumar G, "Design of a Text to Speech Synthesizer to Generate Arbitrary Speech", in International Conference on Speech and Language Technology, Noida, 2004.

[14] Kishore S. P, Black A. W, Kumar R, Sangal R, “Experiments with Unit Selection Databases for Indian Languages”, Seminar on Language, 2003.

[15] Kishore S. P, Kumar R, Sangal R, “A data-driven synthesis approach for Indian languages using syllable as basic unit,” International. Conf. on Natural Language Processing (ICON), 2002, pp. 311–316.

[16] Kishore S. P, Black A W, 2003, “Unit Size in Unit Selection Speech Synthesis”, EUROSPEECH 2003, Geneva, Italy.

[17] Klatt D, “Review of Text-to-Speech Conversion for English,” Journal of the Acoustical Society of America, JASA vol. 82 (3), pp.737-793, 1987.

[18] Klatt D. H, “The Klattalk text—to—speech conversion system,” in Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82, 1982, pp 1589 – 1592.

[19] Krishna N. S, Murthy H. A, Gonsalves T. A, " Text-to-Speech in Indian Languages", in the proceedings of International Conference on Natural Language Processing, ICON-2002, Mumbai, 2002, pp. 317-326.

[20] "Languages of India" (February 2012) [Online]. Available: http://www.webindia123.com/india/people/language.htm

[21] Lemmetty S, "Review of Speech Synthesis Technology", Master's thesis, Helsinki University of Technology.

[22] Mukhopadhyay A, Chakraborty S, Choudhury M, Lahiri A, Dey S, Basu A, "Shruti- an Embedded Text-to-speech System for Indian Languages", IEEE Proceedings on Software Engineering,153, Issue 2, 2006, pp. 75–79.

[23] O'Malley M. H, "Text to Speech conversion Technology", in Computer vol 23, pp 17-23, 1990.

[24] Ramakrishnan A. G, Bali K, Talukdar P. P, Krishna N. S, "Tools for the Development of a Hindi Speech Synthesis System", in 5th ISCA Speech Synthesis Workshop, Pittsburgh, 2004, pp. 109-114.

[25] Rao M. N, Thomas S, Nagarajan T, Murthy HA, "Text-to-speech synthesis using syllable- like units," in National Conference on Communication, IIT Kharagpur, India, 2005, pp. 227–280.

[26] Saraswathi S, Vishalakshy R, "Design of Multilingual Speech Synthesis System" in Information Management, 2010, vol. 2, pp. 58-64.

[27] Singh A. K, "A computational phonetic model for indian language scripts," in Constraints on Spelling Changes: Fifth International Workshop on Writing Systems. Nijmegen, The Netherlands, 2006.

[28] Thomas S, “Natural sounding text-to-speech synthesis based on Syllable-like units,” MS thesis, Indian Institute of Madras, 2007.

[29] Thottingal S, “Dhvani Indian Language Text to Speech System,” (February 2012) [Online]. Available: http://foss.in/2007/register/slides/Dhvani__Indian_Langauge_Text_to_Speech_System-_Demo,_Adding__Language_support,_Usage_370.pdf.

[30] Yegnanarayana B, Rajendran S, Ramachandran V.R, Madhukumar A.S, “Signifcance of knowledge sources for a text-to-speech system for Indian languages”, Sadhana, pages 147-169, 1994.

Research Publications

Research Paper Published

- Shruti Gupta, Parteek Kumar, "Comparative Study of Text to Speech Systems for Indian Languages", in International Journal of Advances in Computing and Information Technology, April 2012,vol. 1, issue 2,pp. 182-192, Available: http://www.ijacit.com/articles/twelve/vol1issue2/vol1issue2/EIJACIT120026.pdf

Research Paper Communicated

- Shruti Gupta, Parteek Kumar, "User Friendly Hindi Text to Speech System based on Dhvani", communicated to CSI Journal on Computer Science and Engineering.