A Gujarati Phonetic Engine using Microsoft SAPI

Dhaval Shah¹, Rucha Shah², Trupti Manik³

¹,²Department of Information Technology, L.D College of Engineering, G.T.U, Ahmedabad, India
³Asst Prof, Department of Information Technology, L.D College of Engineering, G.T.U, Ahmedabad, India
dhs.delete@gmail.com, rucha_shah17@yahoo.in, trupti_manik@yahoo.co.in

Abstract
Gujarati Phonetic Engine is a Text to Speech converter which converts Gujarati text into corresponding speech. Gujarati TTS deals with reading the simple text provided as an input from the user. After tokenizing the input, the stored sound files are fetched from the repository and synthesized to produce the corresponding pronunciation. TTS system can be proved to be useful for the visually impaired people to listen the content in Gujarati language. This paper is aimed to exploit the speech recognition and speech synthesis functionality provided by Microsoft Speech API (Microsoft SAPI). This paper represents how Microsoft SAPI is customized to support text to speech conversion in native Gujarati language. Gujarati TTS system is a Windows based application implemented on the Microsoft’s Windows operating system.

Keywords: TTS, Microsoft SAPI, Speech Synthesis.

1. Introduction
A TTS system deals with conversion of text provided from the computer based devices either in the form of text or scanning the text using an Optical Character Recognition (OCR) system [1]. The concept of Text to Speech synthesis revolves around two sub-parts, a) Text processing and b) Speech generation [2].

Text processing involves analysis of text and converting them into atomic units which can be synthesized. Speech Generation involves synthesizing the sound available as atomic units into a single unit to produce a smooth acoustical sound as a pronunciation of text provided as an input to the system.

Gujarati being a native language the synthesis system at present are not much supportive as compared to international languages. Present state-of-art TTS systems Gujarati language like all Indian language scripts is phonetic in nature. The major task of text processing is to convert the Gujarati text into Phonetic units to identify the above mentioned possible syllables, using syllabification rules.

2. Gujarati Language and Features
Gujarati script has been descended from Brahmi which is a part of Brahmic family. It is a variant of Devanagiri script which was used to write a Gujarati language. There are about 34 consonants (vyanjana), 2 compound characters that are treated as consonants (not lexically though) and 14 vowels (svara) in a Gujarati language.

In Gujarati, each consonant possess inherent vowel (�性 pronounced as ‘a’) except in following cases When 1) consonant is followed by Virama 2) consonant is followed by a punctuation 3) When consonant is followed by some other vowel [5]. A combination of vowels and consonants are used to represent a syllable in a Gujarati language. The different combinations are: C, V, CV, CCV, and CVC where C is consonant and V is vowel.

2.1 Handling Gujarati Text
Indian languages such as Gujarati turns out to be complex for existing synthesis systems due to its complexity in accumulating the text assemblages into digital and processable format, complexity in the grammatical structure and due to lack of sound repositories for local languages such as Gujarati. using rich set of tools available, processing text using text-to-phonemes rules and generating the sound along with varying stress and intonation levels. Local or native The Windows based Text to Speech application makes use of Microsoft Speech API (SAPI) which is a middleware that lies between the application and the speech engine.

Enter Gujarati Text

Text Analysis

Identification of Syllables

Identifying and Retrieving Syllabic Sounds

Synthesis of Speech

Playing the Audio

Fig. 1. Gujarati TTS System
For handling Gujarati text, system uses Unicode standard instead of ASCII. Unicode provides a unique encoding scheme for each character irrespective of platform, program and language. This standard has been implemented in Microsoft .NET framework. Unicode consists of two encoding schemes UTF-8 and UTF-16. The TTS system uses UTF-8 scheme for handling input text. UTF-8 code has the same value as the ASCII code. The high-order bit of these codes is always 0.UTF-8 is backward-compatible with other UTF standards.

3. Existing System

Many improvements have been made in the field of voice-enabled applications owing to the development of rich set of tools for synthesis of sound in order to perform text to speech conversion. Microsoft Speech platform consists of SDK and runtime languages that can be redistributed in applications. The Speech Platform SDK can significantly reduce the amount of efforts required to implement speech in applications for Windows Server or other platforms that can leverage Microsoft's redistributable speech engines. The Speech Application Programming Interface or SAPI is an API developed by Microsoft that allows the use of speech recognition and speech synthesis within Windows applications. It reduces the overhead in code required to make application for speech synthesis and making speech technology robust and flexible for wide range of applications. This speech API is released in many versions.

C# is a programming language designed to build applications that run on .NET framework. Microsoft Visual Studio supports Visual C# with a code editor, debugger and various rich set of development tools. The .NET Framework class library provides access to many operating system services and, well-designed classes that speed up the development cycle significantly [6].

Microsoft.Speech.Synthesis namespace is used to use the synthesizer provided by Microsoft SAPI. It is used to perform text to speech conversion using the voice shipped with Microsoft Speech Platform named HELEN (US-en). This synthesizer can be used to change the properties of sound such as pitch, tone and speed. The existing system is customized by using C# language to provide support for native Gujarati language.

4. System Architecture

The main aim of the system is to perform text to speech conversion of Gujarati text. The basic steps involve text processing and speech generation. Text processing component must be able to convert the Gujarati text into phonetic units using syllabification rules. Fig.1. shows the Windows based Gujarati TTS system. Once the text is given as input, next two steps are part of text processing module and last three steps are part of Speech Generation module.

5. Working

User interacts with the text to speech converter using the interface created using Microsoft Visual C# Express Edition 2010. GUI is equipped with buttons to play, pause, resume and stop the speech at any moment. User provides a Gujarati text as an input using the interface as shown in Fig.2. Gujarati text which is in Unicode format is processed by the application to tokenize it into the atomic phonetic units as per the Gujarati grammatical rules which act nearly as a syllabic unit for pronunciation.

A repository of sound file is created which includes sounds for most of the syllables. Identification of these sound files is based upon the identifier (name) used to represent the syllable itself. Such sound files are fetched from the database.

Synthesis of sound is a functionality provided by the Microsoft SAPI inherently and it converts these syllabic units into single unit suitable for Microsoft Speech Platform to convert them into speech. The audio obtained as a part of speech can be stored as a .wav file in the system or it can be played to the default audio device present in the system.

5.1 Using Microsoft SAPI to extend support for Gujarati language

Microsoft.Speech.Synthesis namespace provides classes for initialization and configuration of speech synthesis engine. It helps in creating user prompt, generating speech, changing speech properties and responding to events. The custom pronunciation is created by defining a lexicon that consists of a word and its pronunciation. Lexicons (Microsoft. Speech) are created as XML documents that follow the format of Pronunciation Lexicon Specification (PLS) Version 1.0. New pronunciation created using external files (XML files) overrides the internal pronunciation supported by speech synthesizer’s dictionary, which forms the base to support pronunciation for Gujarati Language in TTS application. Lexicon is defined using three elements in the XML document. All three elements are needed together to define a custom pronunciation. <lexeme> is a parent element for defining words and its pronunciation. <grapheme> element contains a word to be pronounced. <phoneme> element contains phones that guide the system in pronunciation of word
mentioned in <grapheme> element. These phones are taken from UPS (Universal Phone Set) which is based on IPA (International Phonetic Alphabet) [7].

The document is saved in the .pls format and it is added in the application to SpeechSynthesizer object using addLexicon( ) method provided by package (Microsoft.Speech.Synthesis).

Fig.3. XML Document Defines a New Lexicon to Support Pronunciation for Consonant ‘S’

As mentioned in Fig.3. The definition of lexicon for each consonant (34), vowel (14) and punctuation is added in the existing system to extend its support in pronunciation of Gujarati Phonetic units (syllables).

6. Results
The application is tested for various combinations of Phonetic units in the form of words as well as sentences. The system processes the Gujarati text and recognizes the syllables to identify the source of corresponding sound from repository and synthesize it to produce an audio speech of the text. The test cases which include various combinations of consonants, vowels and results are shown below:

<table>
<thead>
<tr>
<th>Input</th>
<th>Transliteration</th>
<th>Type</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>अक्षर</td>
<td>Kāraṇ</td>
<td>C</td>
<td>Correct Speech</td>
</tr>
<tr>
<td>ओक्षर</td>
<td>Nīki</td>
<td>CP</td>
<td>Correct Speech</td>
</tr>
<tr>
<td>मुक्षर</td>
<td>Māntran</td>
<td>CNC</td>
<td>Correct Speech</td>
</tr>
<tr>
<td>वितक्षर</td>
<td>Kranti</td>
<td>CPN</td>
<td>Correct Speech</td>
</tr>
<tr>
<td>वृंगक्षर</td>
<td>Engineering</td>
<td>Complex word</td>
<td>Correct Speech</td>
</tr>
<tr>
<td>सांसक्षर</td>
<td>Yuddhīthira</td>
<td>Complex word</td>
<td>Correct Speech</td>
</tr>
</tbody>
</table>

Existing system uses the sound provided by Microsoft Speech platform which is Helen (US-en) and thus articulation of speech in some cases differ from the actual native Gujarati pronunciation. The difference arises because the system is extended to support local Gujarati language and the accent of English pronunciation is different to that of Gujarati language in some cases. After performing all combinations of words it was found that in Gujarati TTS application due to accentual variation in speech, the consonants ‘ē’, ‘ē’, ‘ř’ , ‘ō’ (‘Ta’, ‘Da’, ‘Na’, ‘La’ resp.) is replaced by the equivalent sounds like ‘ē’, ‘S’, ‘ē’, ‘ō’ by the system. This difference neither degrades the quality nor produces an incorrect speech but replaces the soft tone articulation with sharp tone. Testing the application shows that in presence of few exceptions due to the inherent feature of Gujarati language, it is able to perform Gujarati text to speech conversion using the existing resources.

7. Conclusion and Future Work
Through this paper a sincere effort has been made to show how existing TTS system can be extended to support speech synthesis for Gujarati language using Microsoft SAPI and Speech platform. As per the objective, an attempt has been made to show how despite having complexities in handling local language such as Gujarati, Windows based application can perform a text to speech conversion. This application can serve as a potential base to make user’s task easier especially task of visually impaired people and to give Gujarati language a global market. This paper also shows that accent-variation in languages can serve as a limiting factor for articulating the speech exactly. There are improvements in certain areas that this system needs in future. The application should be enhanced to read Gujarati text from the pdf files. It must include the naturalization of voices in the speech like human expressions. The difference in the accent can be minimized by allowing system to replace sharp tone with the soft tone expressions. The existing system can be extended further to support numbers, dates and special characters too.

8. References