RESEARCH ARTICLE OPEN ACCESS

Text To Speech System for Telugu Language

M. Siva Kumar[1], E. Prakash Babu[2], Dr. M. V. Subba Reddy[3], M. S. Praveen Kumar[4]

[1](Student, Department of Computer Science and Engineering, Sri Venkatesa Perumal College of Engineering and Technology, Puttur, Andhra Pradesh, India)
[2](Assoc. Prof, Department of Computer Science and Engineering, Sri Venkatesa Perumal College of Engineering and Technology, Puttur, Andhra Pradesh, India)
[3](Professor, Department of Computer Science and Engineering, Sri Venkatesa Perumal College of Engineering and Technology, Puttur, Andhra Pradesh, India)
[4](Student, Department of Computer Science and Engineering, Sri Venkatesa Perumal College of Engineering and Technology, Puttur, Andhra Pradesh, India)

Abstract
Telugu is one of the oldest languages in India. This paper describes the development of Telugu Text-to-Speech System (TTS).In Telugu TTS the input is Telugu text in Unicode. The voices are sampled from real recorded speech. The objective of a text to speech system is to convert an arbitrary text into its corresponding spoken waveform. Speech synthesis is a process of building machinery that can generate human-like speech from any text input to imitate human speakers. Text processing and speech generation are two main components of a text to speech system. To build a natural sounding speech synthesis system, it is essential that text processing component produce an appropriate sequence of phonemic units. Generation of sequence of phonetic units for a given standard word is referred to as letter to phoneme rule or text to phoneme rule. The complexity of these rules and their derivation depends upon the nature of the language. The quality of a speech synthesizer is judged by its closeness to the natural human voice and understandability. In this paper we described an approach to build a Telugu TTS system using concatenative synthesis method with syllable as a basic unit of concatenation.

Keywords **-** Text processing, speech generation, phoneme, grapheme, Speech synthesis.

I. INTRODUCTION

A. Speech Synthesis

The conversion of words in written form into speech is non-trivial. Even if we can store a huge dictionary for most common words; the TTS system still needs to deal with millions of names and acronyms. Moreover, in order to sound natural, the intonation of the sentences must be appropriately generated. Synthesis of speech cannot be accomplished by cutting and pasting smaller units together. Attention has to be paid to smoothing out the discontinuities in such a process so that the resulting signal approximates natural speech. According to the speech generation model used, speech synthesis can be classified into three categories as Articulatory synthesis, Formant synthesis and Concatenative synthesis. Based on the degree of manual intervention in the design, speech synthesis can be classified into two categories Synthesis by rule and Data-driven synthesis.

Prosody and intonation are quite important for natural sounding speech.. There are in existence speech synthesis systems which replicate the prosodic features of human speech. This involves fairly complex parsing of the input sentences and using rather complex rules to determine the intonation patterns.

Generation of sequence of phonetic units for a given standard word is referred to as letter to phoneme rule or text to phoneme rule. Voiced sounds were simulated with a computer model of the vocal fold composed of a single mass vibrating both parallel and perpendicular to the air flow.

The work is divided into 3 main modules.

1. Converting Telugu script to Unicode
2. Differentiating Grapheme
   i. Combination of Consonant-vowel
   ii. Combination of Consonant-consonant
3. Generating voice
   i. Identify the Grapheme recognizer
   ii. Identify the Telugu Audio source

1. Converting Telugu Script to Unicode

In English ASCII characters are used where as In Telugu Unicode characters are used. ASCII takes 8-bits for each character. Unicode takes 16-bits for each character. Unicode provides a unique

number for every character, no matter what the platform, no matter what the program, no matter what the language.

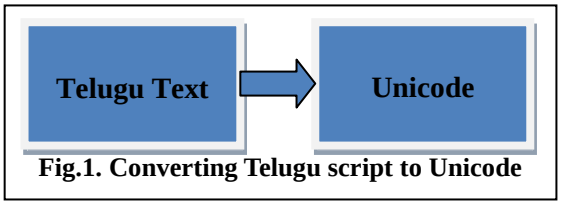

Fig.1. Converting Telugu script to Unicode

2. Differentiating Grapheme

In natural speech, durations of phonetic segments are strongly dependent on contextual factors. For synthetic speech to sound natural, the module for computing segmental duration must mimic these contextual effects as closely as possible.

Grapheme:

Graphemes are "functional spelling units" encompassing one or more letters of the text input, a grapheme in the text input corresponds to a single phoneme.

Phoneme:

Phones characterize any sound that can be produced by a human vocal tract, if a phone is part of a specific language; it becomes a phoneme of the language. Phonemes are the elementary sounds of a language.

In this paper we are going to differentiate A character in Indian language scripts is close to syllable and can be typically of the following form:

Different Combinations:

V-Vowel
C- Consonant
C+V-Consonant+Vowel
C+C-Consonant+Consonant
C+C+V-Consonant+Consonant+Vowel
C+C+C--Consonant+Consonant+Consonant

Example:

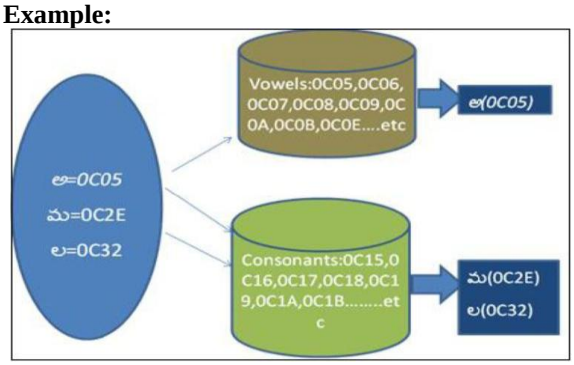

Fig.2. Differentiating grapheme

3. Generating Voice

The function of Text-To-Speech (TTS) system is to convert the given text to a spoken waveform. This conversion involves text processing and speech generation processes. These processes have connections to linguistic theory, models of speech production, and acoustic-phonetic characterization of language. Text processing including end-of-sentence detection, text normalization. Word pronunciation, including the pronunciation of names and the disambiguation of homographs.

In this approach, the pre-recorded speech segments which are to be used in the synthesizer are stored exactly as how it is recorded. Additional information of the speech waveform is attached to the sound to provide proper annotation of the speech waveform.

To synthesize a particular language, required units (diphones) from the database which doesn't contain any language specific information and these selected units were then typically altered by signal processing functions to meet the language specific target specification generated by different modules in the synthesizer. To build a voice/speech for a language text, the steps involved are as follows

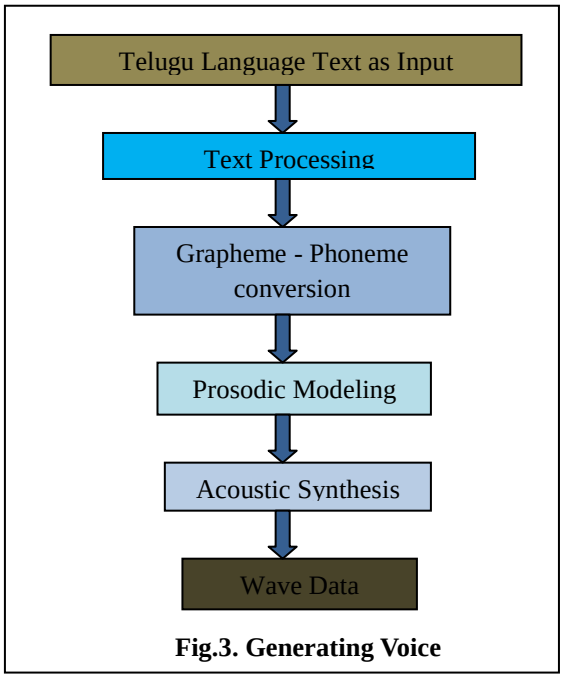

Fig.3. Generating Voice

B. Existing system

Many of the improvements in speech synthesis over the past years have come from creative use of the technologies developed for speech recognition. We can build systems that interact through speech, which the system can listen to what was said, compute or do something, and then speak

back, using spoken language generation. Festivox is speech synthesis system developed by speech group at CMU; it provides a general framework for building speech synthesis systems. It offers full text to speech through a number API's from shell level, though a scheme command interpreter, as a C++ library, from Java, and an Emacs interface. The Festvox project: automating the processes involved in building synthetic voices for new languages. Creating festival TTS for other languages, especially Indian languages is very similar and will mostly require language specific changes to be made to the code in these modules. Festival uses diphone as basic unit.

Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a speech synthesizer, and can be implemented in software or hardware products. A text-to-speech (TTS) system converts normal language text into speech; other systems render symbolic linguistic representations like phonetic transcriptions into speech. The overview of the problems that occur during text-to-speech (TTS) conversion and describe the particular solutions to these problems taken within the AT&T Bell Laboratories TTS system.

II. TELUGU LANGUAGE

The scripts of Indian languages have originated from the ancient Brahmi script. The basic units of writing system are characters which are orthographic representation of speech sounds. A character in Indian language scripts is close to syllable and can be typically of the following form: C, V, CV, CCV and CVC, where C is a consonant and V is a vowel. There are about 35 consonants and about 18 vowels in Indian languages. An important feature of Indian language scripts is their phonetic nature. 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. All Indian language scripts have common phonetic base.

Syllabification Rules

There is almost one to one correspondence between what is written and what is spoken in Indian Languages. Each character in Indian language script has a correspondence to a sound of that language. In Indian languages, a consonant character is inherently bound with the vowel sound /a/, and is almost always pronounced with this vowel. This occurs at both word final and word middle positions. A few heuristic rules to detect IVS of a consonant character are noted below. While letter to phone rules are almost straightforward in Indian languages, the syllabification rules are not trivial. There is need to come up with some rules to break the word into syllables. We have derived certain simplistic rules for syllabification i.e. rules for grouping clusters of C*VC* based on heuristic analysis of several words in Telugu language.

III. SYSTEM ARCHITECTURE

The main purpose of the system is to convert an arbitrary text into its corresponding spoken waveform. Text processing and speech generation are two main components of a text to speech system. To build a natural sounding speech synthesis system, it is essential that text processing component produce an appropriate sequence of phonemic units. Generation of sequence of phonetic units for a given standard word is referred to as letter to phoneme rule or text to phoneme rule. The complexity of these rules and their derivation depends upon the nature of the language. In Telugu TTS the input is Telugu text in Unicode.

Text syllabification will segment the normalized text to syllable unit according to Telugu language rules. Phone set Definition module defines the complete set of phones used in Telugu speech. It also includes feature definitions of these phones. Lexical Analysis module is used to arrive at the phones that make up the pronunciation of a particular word. Since Telugu is phonetic in nature, we do not require a dictionary for lexical analysis. Instead, this module defines letter-to-sound rules (LTS) which are used to arrive at the speech phones based on the spelling of the word

The input text is converted into readable text and the syllabification module will generate the sequence of syllables that should be extracted from the speech corpus to be concatenated and play the sound files.

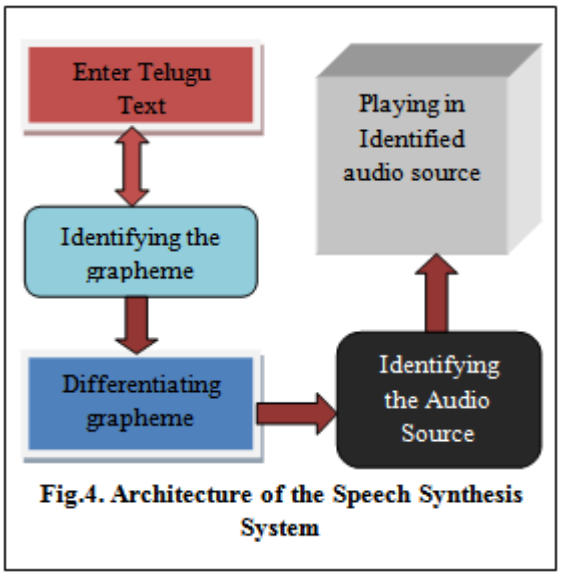

Fig.4. Architecture of the Speech Synthesis System

IV. RESULTS

A Telugu TTS system using syllables as basic unit of concatenation is presented. The quality of the synthesized speech is reasonably natural. The proposed approach minimizes the co articulation effects and prosodic mismatch between two adjacent units. Selection of a unit is based on the presiding and succeeding context of the syllables and the position of the syllable. The system implements a matching function which assigns weight to the syllable based on its nature and the syllable with maximum weight is selected as output speech units. We have observed the efficiency of this approach for Telugu language and found that the performance of this approach is better.

Fig.5. Playing the generated voice

The results showed that there is a strong correlation between the values of the source parameter in the vowel midpoint and the vowel duration. The same parameters tend to decrease on vowel onsets and to increase on vowels offsets. This seems to indicate a prosodic nature of these parameters requiring special treatment in concatenative-based TTS systems that use source modification techniques, such as pitch synchronous overlap add and multipulse.

V. CONCLUSION

Speech synthesis techniques , it is much easier to build a voice in a language with fewer sentences and a smaller Speech]. The proposed approach minimizes the co articulation effects and prosodic mismatch between two adjacent units. Selection of a unit is based on the presiding and succeeding context of the syllables and the position of the syllable. The system implements a matching function which assigns weight to the syllable based on its nature and the syllable with maximum weight is selected as output speech units. We have observed the efficiency of this approach for Telugu language and found that the performance of this approach is better.

Telugu TTS system using syllables as basic unit of concatenation is presented. The quality of the synthesized speech is reasonably natural. The proposed approach minimizes the co articulation effects and prosodic mismatch between two adjacent units. Selection of a unit is based on the presiding and succeeding context of the syllables and the position of the syllable. The system implements a matching function which assigns weight to the syllable based on its nature and the syllable with maximum weight is selected as output speech units. We have observed the efficiency of this approach for Telugu language and found that the performance of this approach is better.

VI. FUTURE ENHANCEMENTS

The generated speech shows distortion at the concatenation point of two syllables. If this distortion is significant then it would loose the naturalness. This distortion can be minimized by adding smoothing module which would modify the speech parameters like pitch formants and intensity at the concatenation point. More attention needs to be paid to the frequency and amplitude range as well as the duration of each pre-recorded segments to produce high quality speech.

A number of other ongoing projects are aimed at developing a POS tag set, POStagger and a tagged corpus for Sinhala. Further work will focus on expanding the pronunciation lexicon. At present, the G2P rules are incapable of providing accurate pronunciation for most compound words. Thus, we are planning to construct a lexicon consisting of compound words along with common high frequency words found in our Sinhala text corpus, which are currently incorrectly phonetized.

REFERENCES

[1] Lakshmi A, Hema A Murthy. A Syllable Based Continuous Speech Recognizer for Tamil. In Proc. of the 2nd Int. Workshop on East-Asian Language Resources and Evaluation,2009.

[2] Sreekanth Majji, Ramakrisshnan A.G "Festival Based maiden TTS system for Tamil Language", 3rd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics October 5-7, 2007, PoznaĔ, Poland.

[3] ZenH.,NoseT.,YamagishiJ.,SakoS.,Masuko T.,Black A.W., andTokudaK.,"The hmm-based speech synthesis system version2.0," in Proc.ofISCASSW6, Bonn, Germany,2007.

[4] BlackA.W. ZenH. and TokudaK. "Statistical parametric speech synthesis," in proceeding sofIEEEInt.Conf. Acoust. Speech, and Signal Processing, Honolulu, USA, 2007.
[5] Rohit Kumar, S. P. Kishore, "Automatic Pruning of Unit Selection Speech Databases for Synthesis without loss of Naturalness", submitted to ICSLP, Jeju Island, Korea,2004
[6] S P Kishore, Rohit Kumar and Rajeev Sangal, "A Data Driven Synthesis Approach For Indian Languages using Syllables as BasicUnit", in Proceedings of Intl. Conf. on NLP (ICON) 2002, pp. 311-316, Mumbai, India, 200
[7] A.M. Zeki and N. Azizah, "A Speech Synthesizer for Malay Language", National Conference on Research and Development in Computer Science, Selangor, Malaysia, October 2001
[8] X. Huang, A.Acero and H.-W. Hon, "Spoken Language Processing A Guide to Theory, Algorithm and System Development", New Jersey: Prentice Hall, 2001
[9] Min Chu, Hu Peng, Hong-yun Yang, Eric Chang, "Selecting Non-Uniform Units from a very large corpus for Concatenative Speech Synthesizer", in Proceedings of ICASSP, Salt Lake City, 2001
[10] Sproat R. (1995): finite-state architecture for tokenization and grapheme-to-phoneme conversion for multilingual text analysis". In From text to tags: Issues in multilingual language analysis. Proc. ACL SIGDAT Workshop (Dublin, Ireland), 65-72
[11] Sproat R., Olive J. (1995): "Text to speech syn-thesis". AT&T T echnical Journal 74(2), 35-44
[12] Sproat R., Olive J. (1996): "A modular architec-ture for multi-lingual text-to-speech". In J. van Santen, R. Sproat, J. Olive and J. Hirschberg (eds.), Progress in speech synthesis (Springer, New York).
[13] T alkin D., Rowley J. (1990): "Pitch-syn-chronous analysis and synthesis for TTS systems".Proc. ESCA Workshop on Speech Synthesis (Autrans, France), 55-58.
[14] A.M. Zeki and N. Azizah, "A Speech Synthesizer for Malay Language", National Conference on Research and Development in Computer Science, Selangor, Malaysia, October 2001.