Text to Speech Conversion of Documents Using Concatenative Speech Synthesis

Radha N[1], Neharika.S[2]
Assistant Professor, Department of Information Technology, SSN College of Engineering, Kalavakkam, Chennai, India[1]
U.G. Student, Department of Information Technology, SSN College of Engineering, Kalavakkam, Chennai, India[3]

Abstract: Text-to-speech conversion is used for the transformation of semantic information in the form of text into speech. This system is predominantly used in reading devices for visually challenged and illiterate people to help them overcome numerous problems they face in their life owing to their inability to perceive the script around them. However, lately the use of this system has extended from more than just being an aid to people to multiple functionalities like voice mail and response systems, and also digitization of text files to audio files. The text to speech conversion is facilitated by extracting text from a PDF file, followed by normalizing the text to be on par with the speech conversion conditions and converting it to speech using dictionary approach.The TTS system generates synthetic speech by concatenating segments of natural speech. The architecture of the system is designed as a modular pipeline where each module handles one particular step in the process of converting text into speech.

***Keywords:**Text-to-speech, digitization of text files, Hidden Markov Model, Unidirectional modular system, concatenative synthesis*

*****

I. INTRODUCTION

The aim of the text-to-speech conversion system is to convert textual information into an audio format, primarily to aid the visually challenged people. A sharp hike in the sales of smart phones creates the necessity for the service providers to provide smart phones which cater the needs of all cadres of the society. Hence a system to provide non-visual feedback i.e. an output in the form of auditory signals is required. Text-to speech systems provide this functionality and enables smart phones to convey messages in the form of speech.

In the recent past, AT&T Bell laboratories made the most significant advances in multilingual text-to-speech conversion. Their system generates speech by concatenating segments of natural speech. Its architecture is in the form of a unidirectional modular system with each module handling a particular functional step in the conversion process. This modular structure has been significant in out TTS system for multiple languages. A single set of modules with all language-specific information in tables is incorporated in the system.

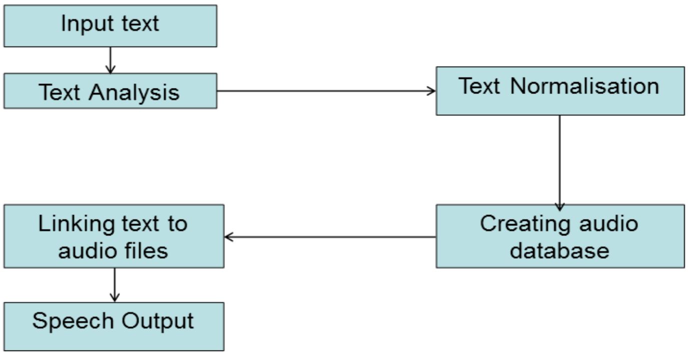
The key features of the modular architecture incorporated from the AT&T Bell system are:

- Modules with well-defined sub tasks
- Division of labour in the software team promoting faster development of the system
- Enable interruption and re-initiation anywhere i.e. Insertion of tools or programs to modify the TTS parameters at any point of time

The three module system facilitates unidirectional information flow with uniform set of data structures performing inter-module communication. A stream of synthesis parameters is the final output and is used by the waveform synthesizer for speech conversion. This system is currently designed for 9 languages namely Mandarin Chinese, Taiwanese, Japanese, Mexican Spanish, Russian, Romanian, Italian, French, and German. As the primary stages of working on a new language imposed the problem of unavailability of much information, we start with a phonetic representation of a language to build the acoustic module, while the text analysis and prosodic parts are added later.

In the second section, the system architecture is explained which involves three modules: text to pdf conversion, normalization of text and text to speech conversion. The third section describes the how the system is implemented using matlab and what kind of testing is performed to make the system efficient.

II. SYSTEM ARCHITECTURE



The proposed system has the three following modules:

1. **Extraction of text from a PDF file into a text file:**

During text analysis, the input text is organized into a list of words consisting of numbers, abbreviations, acronyms and idioms. A problem encountered in this module is the inability to determine the sentence termination and punctuation ambiguity. But this can be eliminated to a certain extent by the usage of elementary grammar.

2. Text Normalization

Text normalization is the conversion of text to a standard form. It is performed to bring about a consistency in the text before any form of text processing is performed. Identifying punctuations and pauses in sentences are one of the primary functions of text normalization.

The main 4 phases of Text Normalization are:

a) **Number converter:**
Numbers have contextual pronunciations. Their diction varies depending on whether the number represents a date, a phone number, a quantifier or decimal/fractional numbers.

b) **Abbreviation converter**:
Abbreviations are converted to their expanded forms. Eg: Mr to Mister.

c) **Acronym converter**:
Acronyms are converted to individual letter components.

d) **Word segmentation:**
Sentences are a group of word segments. Special delimiter to separate segments are identified. (i.e. „||‟).Segments can be an acronym, a single word or a numeral.Punctuation marks are also identified.

One of the major problems faced in TTS conversion is the contextual pronunciation of words, abbreviations and acronyms. Abbreviations are pronounced differently in different situations, for example St is pronounced as saint or street depending on the context. Similar complications arises with numbers and sometimes even words. These problems are overcome by using various heuristic techniques to disambiguate the words by the statistical analysis of the neighbouring words and the frequencies of the occurrences. The Hidden Markov Model (HMM) is used for this purpose.

3. Conversion of normalized text into speech

Pronunciation of words in speech has two approaches:

a) **Dictionary based approach**
All words with the right pronunciation are stored in a dictionary. This approach is very quick and accurate. Quality of the speech generated is better. The need for a large database to store all the words is one major drawback of this approach. Also, the system fails to work if a word is not found in the dictionary.

b) **Rule based approach**
In this approach, the phonemes of the words are blended together based on some rule. This approach does not require any database but the complexity of the system grows with the irregularity in the input.

The dictionary based approach is preferred as the quality of the speech synthesized has superior quality when compared to the rule based approach. Primarily, the dictionary based approach involves the creation of an audio dictionary. The TTS system is then linked to the audio dictionary to enable speech conversion.

III. IMPLEMENTATION

To implement the three modules the following software are used:

a) Matlab
b) Apache PDF Box library
c) Java SE

Every module is developed and tested separately, and integrated one by one. Performance is tested at every stage. The final system is then assembled and checked for the desired functionality and performance

Text Analysis: First, input text is converted into a finite-state acceptor which is then composed sequentially with a set of transducers that go from the surface representation to lexical analysis. Since this yields all possible lexical analyses for a given input, a language model helps find the presumably ‘correct’ or most appropriate analysis. The best path through the language model is then composed with a transducer that goes from lexical analysis to phonological representation and pronunciation. Given equal probabilities for both alternatives, only information provided by a part-of-speech tagger or parser can help disambiguate and determine the correct pronunciation. The text analysis component also performs a tokenization of the input text into sentences and words. End-of-sentence detection, abbreviation, acronym and number expansion, word tokenization and other pre-processing problems are typically solved using a set of heuristics. Other linguistic information as derived by text processing includes information on parts of speech as well as on phrasing and accenting, and jointly forms the input to subsequent modules: segmental duration,intonation, unit selection and concatenation, and synthesis.

In our multilingual systems, the new generalized text analysis component replaces all the modules up to segmental duration.

Segmental duration: The duration module assigns duration to each phonetic segment. The module as such is language-independent, with all language specific information being stored in tables. Table construction is performed in two phases: inferential statistical analysis of the speech corpus, and parameter fitting. In the case of the German TTS system, we first designed a factorial scheme, i.e. the set of factors and distinctions on these factors that are known or expected to have a significant impact on segmental durations. An important requirement was that the factors can be computed from text. We then applied a quantitative duration model that is implemented as a particular instantiation of a 'sums-of-products' model whose parameters are fitted to a hand-segmented speech database.

The data show rather homogeneous patterns in that speech sounds within a given phone class generally exhibit similar durational trends under the influence of the same combination of factors.

Among the most important factors are:

a) Syllable stress (for nuclei, and to some extent for stops and fricatives in the onset);

b) Word class (for nuclei);

c) Presence of phrase

Intonation: The intonation module computes a fundamental frequency contour 0 by adding three types of time dependent curves: a phrase curve, which depends on the type of phrase, e.g., declarative vs. interrogative; accent curves, one for each accent group (accented syllable followed by zero or more non-accented syllables); and perturbation curves, which capture the effects of obstruent on pitch in the post-consonantal vowel. This approach shares some concepts with the so called super positional intonation models that have been applied to a number of languages. These models analyze the 0 contour as a complex pattern that results from the superposition of several components, each of which has its own temporal domain.

Acoustic inventory: The majority of units in the acoustic inventory are diphones, i.e., units that contain the transition between two adjacent phonetic segments, starting in the steady state phase of the first segment and ending in the stable region of the second segment. Contextual or co-articulatory effects can require the storage and use of context-sensitive 'allophonic' units or even of triphones. For example, the current acoustic inventory of the German TTS system consists of approximately 1250 units, including about 100 context-sensitive units. This inventory is sufficient to represent all phonotactically possible phone combinations for German. However, it will have to be augmented by units representing speech sounds that occur in common foreign words or names, e.g., the interdental fricatives and the /w/ glide for English, or nasalized vowels for French. For acoustic inventory construction we use a new procedure that performs an automated optimal element selection and cut point determination.

Selection, concatenation, synthesis: The unit selection and concatenation modules select and connect the acoustic inventory elements. These modules retrieve the necessary units, assign new durations, pitch contours and amplitude profiles and pass parameter vectors on to the synthesis module which uses one of the synthesis methods described below to generate the output speech waveform. The parametric waveform synthesis module provides flexible engines to assure the highest quality speech output for a given hardware platform and number of parallel channels running on that platform. Since usually more than 60% of the computational effort of the total TTS system is spent on waveform synthesis, hardware constraints can be met most easily by trading off quality *vs.* complexity in the algorithms used by the synthesizer.

Once the various modules are programed unit testing is performed to the individual modules to determine their efficiency. Following the unit testing, integration testing is performed by running the modules together to determine the errors and inefficiencies involved in executing the integrated modules. Program texting is also performed to determine the logical and syntactical errors. The system is then tested in a user's environment and their feedback is used to make further changes to the system.

IV. CONCLUSION

The next generation TTS systems are asked to deal with emotions in speaking styles. These narrative characteristics as a future addition would enhance the output of this application. Although an exhaustive audio dictionary of 7000+ words has been taken, continuous evolving of this database would make the speech output more accurate. TTS systems can also be implemented to develop smart phones which can be used by the visually challenged people. It can also be implemented to develop systems to read public sign boards and posters.

V. REFERENCES

[1]. Text-To-Speech: A Simple Tutorial by E.Sasirekha and Chandra,International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[2]. http://www.mathworks.in/videos/creating-a-gui-with-guide-68979.html

[3]. Recent advances in multilingual text-to-speech synthesis, Bernd Mobius, JuergenSchroeter, Jan van Santen, Richard Sproat, Joseph Olive,AT&T Bell Laboratories, Murray Hill, NJ, USA

[4]. http://shtooka.net Audio Dictionary database

[5]. http://www.mathworks.in/matlabcentral/answers/69945Matlab Question and Answers Forum.

[6]. www.pdfboxlibrary.com PDF to text converter software.