Text-to-Speech Synthesis Using Concatenative Approach

Mustapha Oloko-oba
Coperative Information Network, (COPINE),N…
14 PUBLICATIONS **9** CITATIONS

Osagie Samuel
University of Abuja
1 PUBLICATION **2** CITATIONS

Text-to-Speech Synthesis Using Concatenative Approach

[1]Oloko-Oba Mustapha O, [2]Ibiyemi T.S, [3]Osagie Samuel E.
[1] Cooperative Information Network, National Space Research and Development Agency, Obafemi Awolowo University, Ile-Ife, Nigeria.
[2] Department of Electrical & Electronics Engineering, University of Ilorin, Kwara State, Nigeria.
[3] ICT Centre, University of Abuja, Abuja, Nigeria.

Abstract— A text-to-speech (TTS) synthesizer is a computer based system that should be able to read any text aloud. Most text-to-speech synthesis lacks naturalness and intelligibility. This study is aimed at achieving the ease with which the output is understood and how closely the output sounds like human speech referred to as intelligibility and naturalness. In this research, an algorithm is developed in C-programming language capable of recognizing 50 isolated English vocabularies and produces the corresponding sound output using concatenative technique which employs natural human voice and produces the most natural-sounding speech. The vocabularies used as inputs were pre-recorded in .wav format and stored in the database. A user input the text which is searched through the database and the corresponding sound is played if a match is found otherwise an error message is given to check your spelling and try again.

Keywords: *Pre-processing, Phoneme, Syllable, Prosody, Acoustic, Text, Onset, Coda, Nucleus.*

I. INTRODUCTION

Text-to-speech (TTS) synthesis technology gives machines the ability to convert arbitrary text into audible speech, with the goal of being able to provide textual information to people via voice messages.

Speech is one of the most vital forms of communication in our everyday life. Since speech is a primary mode of communication among human beings, it is natural for people to expect to be able to carry out spoken dialogue with computers. This involves the integration of speech technology and language technology. Speech synthesis is the automatic generation of artificial speech signal by the computer. In the last few years, this technology has been widely available for several languages for different platform ranging from personal computer to stand alone systems [1, 2].

Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a speech synthesizer. Speech synthesizer is a computer-based system that should be able to read any text aloud, whether it was directly introduced in the computer by an operator or scanned and submitted to an Optical Character Recognition (OCR) system. The goal of a text to speech system is to convert an arbitrary given text into a spoken waveform. Main components of text to speech system are: Text processing and Speech generation [3].

Recent progress in speech synthesis has produced synthesizers with very high intelligibility but the sound quality and naturalness still remain a major problem. The dependence of human computer interaction on written text and images makes the use of computers impossible for visually impaired and illiterate masses. However, automatic speech generation from natural language can overcome these obstacles in the present era of human computer interactions.

A. Methods of Synthesizing Speech

The most important qualities of a speech synthesis system are ***naturalness*** and ***intelligibility***. 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 is both natural and intelligible. Speech synthesis systems usually try to maximize both characteristics [4].

Speech synthesis can easily be produced in different ways with various advantages and disadvantages. The different methods of speech synthesis are classified into:

- Articulatory synthesis, which attempts to model the human speech production system directly.
- Formant synthesis, which models the pole frequencies of speech signal or transfer function of vocal tract based on source-filter-model.
- Concatenative synthesis, which uses different length pre-recorded samples derived from natural speech.

In this research, Emphasis is on the concatenative methods.

B. Concatenative Synthesis

Concatenative synthesis is based on the concatenation of segments of recorded speech. It is characterized by storing, selecting, and smoothly concatenating pre-recorded human utterances (phonemes, syllables, or longer units) [5].

Concatenative synthesis produces the most natural-sounding synthesized speech. However, concatenative synthesizers are usually limited to one speaker and one voice and usually require more memory capacity than other methods [3].

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 labeling procedures become more difficult and complex. In present systems units used are usually words, syllables, demisyllables, phonemes, diphones, and sometimes even triphones.

There are three main subtypes of concatenative synthesis: Unit selection synthesis, Diphone synthesis and Domain-specific synthesis.

Word is perhaps the most natural unit for written text and some messaging systems with very limited vocabulary. Concatenation of words is relatively easy to perform and coarticulation effects within a word are captured in the stored units. However, there is a great difference with words spoken in isolation and in continuous sentence which makes the continuous speech to sound very unnatural [6]. Because there are hundreds of thousands of different words and proper names in each language, word is not a suitable unit for any kind of unrestricted TTS system.

C. English Phoneme

Table 1: List of English Phoneme Set

Phoneme	Spelling(s) and Example Words
/A/	a (table), a_e (bake), ai (train), ay (say)
/a/	a (flat)
/b/	b (ball)
/k/	c (cake), k (key), ck (back)
/d/	d (door)
/E/	e (me), ee (feet), ea (leap), y (baby)
/e/	e (pet), ea (head)
/f/	f (fix), ph (phone)
/g/	g (gas)
/h/	h (hot)
/I/	i (I), i_e (bite), igh (light), y (sky)
/i/	i (sit)
/j/	j (jet), dge (edge), g[e, i, y] (gem)
/l/	l (lamp)
/m/	m (my)
/n/	n (no), kn (knock)
/O/	o (okay), o_e (bone), oa (soap), ow (low)
/o/	o (hot)
/p/	p (pie)
/kw/	qu (quick)
/r/	r (road), wr (wrong)
/s/	s (say), c[e, i, y] (cent)
/t/	t (time)
/U/	u (future), u_e (use), ew (few)
/u/	u (thumb), a (about), e (loaded), o (wagon)
/v/	v (voice)
/w/	w (wash)
/ks/ or /gz/	x (box, exam)
/y/	y (yes)
/z/	z (zoo), s (nose)
/OO/	oo (boot), u (truth), u_e (rude), ew (chew)
/oo/	oo (book), u (put)
/oi/	oi (soil), oy (toy)
/ou/	ou (out), ow (cow)
/aw/	aw (saw), au (caught), a[l] (tall)
/ar/	ar (car)
/sh/	sh (ship), ti (nation), ci (special)
/hw/	wh (white)
/ch/	ch (chest), tch (catch)
/th/ or /th/	th (thick, this)
/ng/	ng (sing), n (think)
/zh/	s (measure)

II. METHODOLOGY

The algorithm for the text-to-speech system in this research is described towards enabling designated vocabulary of 50 isolated English words to be recognized and spoken out. C-programming language was used as the language for coding the text-to-speech algorithm. Text-to speech consists of **Input**, **Process** and **Output.**

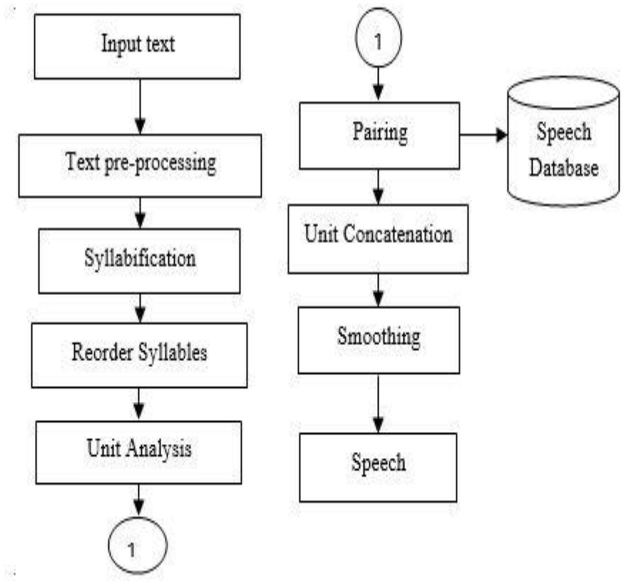


Figure 1: Block Diagram of a Text-To-Speech System

Fig.1 above indicates the steps involved in the text-to-speech algorithm. The system takes (text) as input and produces the corresponding (sound) output.

The input text for the algorithm is the America standard code for information interchange (ASCII) which is pre-process by converting raw text containing symbols like into the equivalent written out word i.e Normalized.

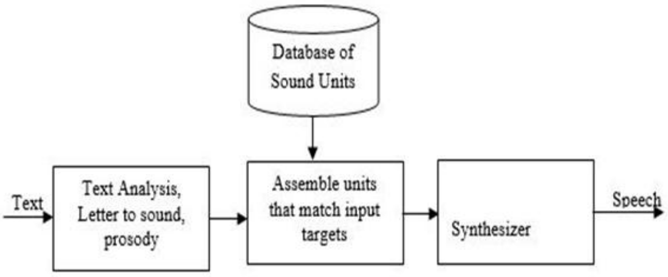

Database of Sound Units
Text
Text Analysis, Letter to sound, prosody
Assemble units that match input targets
Synthesizer
Speech

Figure 2: Block Diagram of Concatenative TTS System

A. Preparation Of Speech Database

In this approach, to prepare *speech database*, the small pieces are either cut from the recordings or recorded directly and then stored. Then, at the synthesis phase, units selected from the speech database are concatenated and, the resulting speech signal is synthesized as output.

The database consists of sound files in ".wav" format resenting each of the phonemes have been recorded and stored in the Phoneme Folder. Each input text is converted to the primitive phoneme for correct pronunciation for example

Professor = prəfɛsər
University = junɪvərsəti

This is further broken down to the individual phoneme units in the word. The word is then composed by concatenating the audio units representing the word from audio files in the database.

B. *Text Analysis*

Text analysis is all about transforming the input text into a 'speakable' form. At the minimum, this contains the normalization of the text so that numbers and symbols become words, abbreviations are replaced by their corresponding whole words or phrases, and so on. This process typically employs a large set of rules that try to take some language-dependent and context-dependent factors into account. The most challenging task in the text analysis block is the linguistic analysis which means syntactic and semantic analysis and aims at understanding the content of the text. Of course, a computer cannot understand the text as humans do, but statistical methods are used to find the most probable meaning of the utterances. This is important because the pronunciation of a word may depend on its meaning and on the context (for instance, the word record is pronounced in different ways depending on whether it is a verb or a noun). Finally, the text analysis block is supposed to provide prosodic information to the subsequent stages. It can, for example, signify the positions of pauses based on the punctuation marks, and distinguish interrogative clauses from statements so that the intonation can be adjusted accordingly [8].

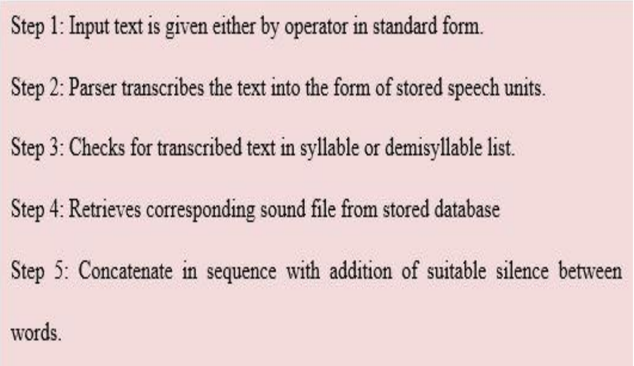
C. *Letter-to-Sound*

The letter-to-sound also known as grapheme-to-phoneme conversion means the translation of a written text into the corresponding stream of phonemes. Thus, grapheme-to-phoneme conversion refers to the process of converting a stream of orthographical symbols into an appropriate symbolic representation of the corresponding sequence of sounds in the form of a series of phonemic symbols.

D. *Prosody*

Prosody is a concept that contains the rhythm of speech, stress patterns and intonation. The attachment of certain prosodic features to synthetic speech employs a set of rules that are based on the prosodic analysis of natural speech. Prosody plays a very important role in the understandability of speech.

E. *Synthesis*



```
Step 1: Input text is given either by operator in standard form.
Step 2: Parser transcribes the text into the form of stored speech units.
Step 3: Checks for transcribed text in syllable or demisyllable list.
Step 4: Retrieves corresponding sound file from stored database
Step 5: Concatenate in sequence with addition of suitable silence between words.
```

Figure 3: Synthesizer Pseudo Code

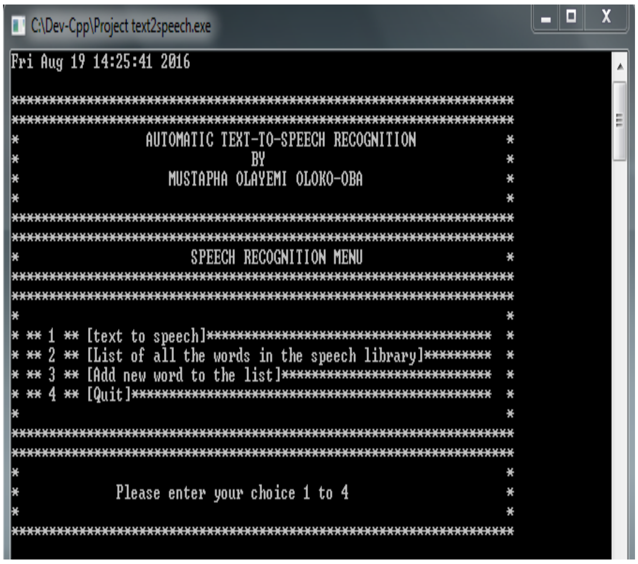
Speech synthesis block finally generates the speech signal i.e converts the symbolic linguistic representation into sound. This can be done by selecting speech units from the database. A sophisticated search process is performed in order to find the appropriate phoneme, diphone, triphone, or other unit at each time. The resulting short units of speech are joined together to produce the final speech signal. One of the biggest challenges in the synthesis stage is actually to make sure that the units connect to each other in a continuous way so that the amount of audible distortion is minimized.

III. RESULT AND DISCUSSION

The core of the system is to transcribe the input text into the form of the stored units, search the speech library for a match of the transcribed text in syllables, retrieve and map the corresponding sound file to the input word and be pronounced. The result which the software produces is analyzed and presented objectively. C-programming language was used to code the algorithm.

A. *Text-to-Speech Interface*

The text-to-speech interface shows the title of the study with the researcher name and also contains the speech recognition menu from 1 to 4 where users can easily navigate as shown in fig 4 below.



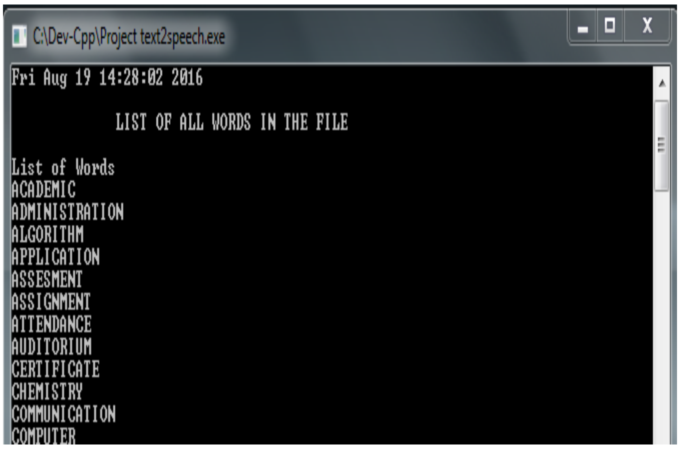
C:\Dev-Cpp\Project text2speech.exe

```
Fri Aug 19 14:25:41 2016

********************************************************************
********************************************************************
*                  AUTOMATIC TEXT-TO-SPEECH RECOGNITION            *
*                                BY                                *
*                     MUSTAPHA OLAYEMI OLOKO-OBA                   *
*                                                                  *
********************************************************************
********************************************************************
*                       SPEECH RECOGNITION MENU                    *
********************************************************************
********************************************************************
*                                                                  *
* ** 1 ** [text to speech]***************************************  *
* ** 2 ** [List of all the words in the speech library]**********  *
* ** 3 ** [Add new word to the list]*****************************  *
* ** 4 ** [Quit]*************************************************  *
*                                                                  *
********************************************************************
********************************************************************
*                                                                  *
*           Please enter your choice 1 to 4                        *
*                                                                  *
********************************************************************
```

Figure 4: Text-to-Speech Menu Interface

B. *List of Input Word*

This menu displays all the list of words stored in the speech library. It guides the users against typing wrong words or spellings in the system. This menu is displayed when we press 2 on the keyboard as shown below.



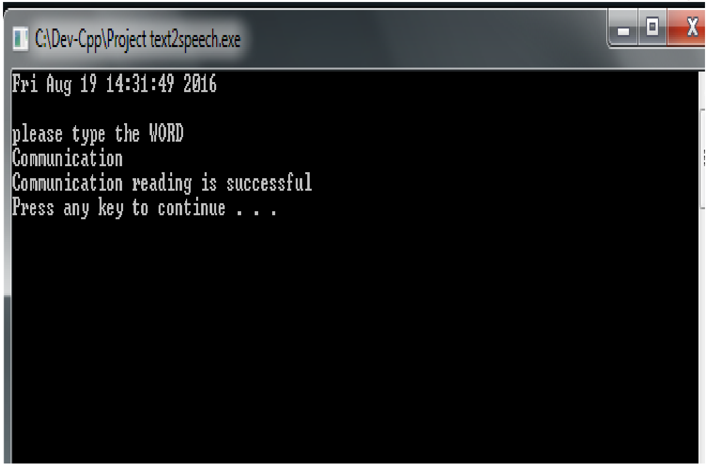
C:\Dev-Cpp\Project text2speech.exe

```
Fri Aug 19 14:28:02 2016

            LIST OF ALL WORDS IN THE FILE

List of Words
ACADEMIC
ADMINISTRATION
ALGORITHM
APPLICATION
ASSESMENT
ASSIGNMENT
ATTENDANCE
AUDITORIUM
CERTIFICATE
CHEMISTRY
COMMUNICATION
COMPUTER
```

Figure 5: List of Input Word

C. *Text Input Reading*

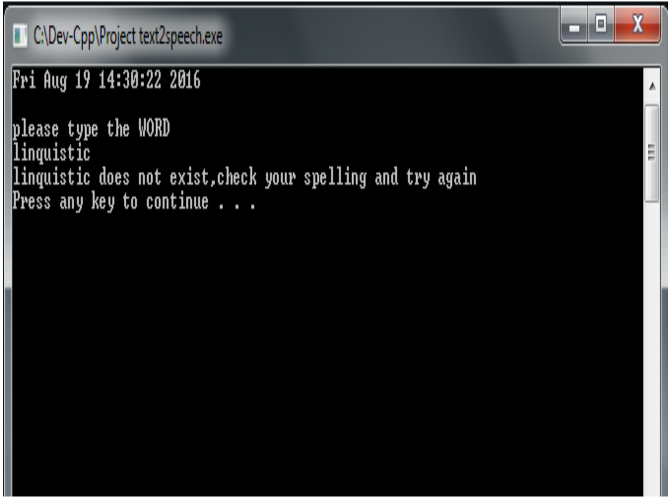
Text-to-speech menu is displayed when we press 1 and then hit the enter key on the keyboard. It shows an interface with the instruction "please type the word" then any prerecorded word in the library can then be entered and the system responds whether the reading is successful or not. If the reading is successful, the system speaks out the word which sounds natural and intelligent as shown in fig 6.



```
Fri Aug 19 14:31:49 2016

please type the WORD
Communication
Communication reading is successful
Press any key to continue . . .
```

Figure 6: Text Input Reading Successful.

The system shows an error message when it searched through the database and could not find a match for the input word. Example the word linguistic was not found in the database and system respond is “linguistic does not exist. Check your spelling and try again”.



```
Fri Aug 19 14:30:22 2016

please type the WORD
linguistic
linguistic does not exist,check your spelling and try again
Press any key to continue . . .
```

Figure 7: Text Input Error Message

CONCLUSION

Since the most important qualities of a text-to-speech synthesis system are naturalness and intelligibility, a concatenative approach of speech synthesis was used to develop an interactive graphical user interface that enable users to carry out all the needed operations ranging from displaying list of available words, inputting the words for the natural speech to be spoken, adding new vocabulary to the database as well as quitting the operation. Further studies would be needed to use local Nigeria dialect as input text to be transcribed to English and vice versa.

References

[1] Santen J., Sproat R., Olive J., Hirschberg J. “Progress in Speech Synthesis”, Springer-Verlag New York Inc, 1997.

[2] Kleijn K., Paliwal K. “Speech Coding and Synthesis”. Elsevier Science B.V., the Netherlands, 1998.

[3] S.D Shirbahadurkar and D.S.Bormane. “Speech Synthesizer Using Concatenative Synthesis Strategy for Marathi language” (Spoken in Maharashtra, India). International Journal of Recent Trends in Engineering, 2009. Vol 2, No. 4. Pp 80-82.

[4] Taylor, Paul. “Text-to-speech synthesis”. Cambridge, UK: Cambridge University Press 2009. p. 3. ISBN 9780521899277.

[5] Cerrato Loredana. “Introduction to Speech Synthesis”: available on

http://stp.lingfil.uu.se/~matsd/uv/uv05/motis/lc_synt.pdf 2005.

[6] Allen, Jonathan; Hunnicutt, M. Sharon; Klatt, Dennis (1987). “From Text to Speech”: The MITalk system. Cambridge University Press 1987. ISBN 0-521-30641-8 .

[7] John Kominek and Alan W. Black. “CMU ARCTIC databases for speech synthesis”. CMU-LTI-03-177. Language Technologies Institute, School of Computer Science, Carnegie Mellon University 2003.

[8] Huang X., Acero A., Hon H.-W. “Spoken Language Processing: A Guide to Algorithms and System Development”. Prentice Hall PTR, 2001.

[9] Schwarz, D. "Current Research in Concatenative Sound Synthesis" Proceedings of the International Computer Music Conference (ICMC), Barcelona, Spain, September 5-9, 2005.