

Designing a Prototype Model of the English Text-to-Speech System in the User's Voice

Saloni Sharma¹, Piyush Pratap Singh²

*M.tech student in Statistical Computing (Data science)*¹

*Associate Professor, SCSS, JNU*²

School of Computer and System Sciences, Jawaharlal Nehru University, Delhi, India

[*engineersaloni159@gmail.com*](mailto:engineersaloni159@gmail.com)¹, [*piyushpsingh@mail.jnu.ac.in*](mailto:piyushpsingh@mail.jnu.ac.in)²

Abstract: In this era, we know that the primary building block of any communication is speech, so there is no communication establishment without saying anything. So we see the importance of conveying the correct information to the right person; without that, there is no meaning in communication. In today's world, everything is going digital, like a smartphone, text messages, and many different websites, so accessing these things can be a significant issue for dumb or illiterate people. They don't know how to read something written on the screen. One more issue is someone who cannot understand the language of synthesized speech generated by the system due to the accent problem, and they can now listen to the written text in their own accent and their own voice. So from here, the motivation for my research came. Since many tools are developed using this idea, I aim to build a tool in python language for generating speech for the written text in the User's voice. For this purpose, I have used Audacity software to record the phoneme sound and save it to the directory in wave format (*.wav). After that, these phonemes are concatenated to form a meaningful word. Since it is a domain-specific prototype model, it doesn't require a lot of memory to store phoneme sounds because some specific terms are chosen from a particular domain developing the tool.

Keywords: Natural language processing, Text-to-speech, English language, Audacity Software, .wav format, Phoneme, Concatenative synthesis

1. Introduction

In the field of natural language processing, language plays a vital role in understanding things in a better way. When we talk about communication, speech is the fundamental building block because it is the most commonly used method for communication. There is a lot of information stored on the web in the form of text, video, audio, etc. We can ask anything and can get information about anything. But human-like to hear the information provided over the web rather than just reading or watching. But it is not an easy task for a system to read the text and generate the sound like a human does. To make a machine understand and respond after processing the human language was the main objective of this NLP. Language is our primary

mode of communication, and it allows us to communicate with new people and communicate our ideas to others. A language's secondary goal is to transmit someone's feelings, emotions, or attitudes. Only a few languages, including English, can do both of these functions. It has made its way into practically every business. It is the universal language that we utilize in various situations, including business and entertainment. People come from different backgrounds they face a lot of difficulties understanding the English accent because it is very different from what they used to speak in terms of pronunciation. People struggle to communicate in foreign languages because their brains are comfortable only speaking in their native language.

2. Literature Survey

Speech recognition is one of the most challenging research areas in NLP. In 1961, IBM 7094 was the first Singing Computer made that sings the "Daisy Bell" song in a computer-synthesized voice.

In the paper "[ⁱ] Neural Networks for Text-to-Speech Phoneme Recognition", they discussed two different Artificial Neural Network approaches for the text to speech application: Staged Back Propagation Neural Network and Self-Organizing Map. This paper focused on this approach Because this engine aids in the mapping of misspelled letters in the database and aids in the elimination of inconsistencies produced by different pronunciations of the very same word, and it is faster than a dictionary-based approach [ⁱⁱ]

In the paper "Evolution of Text-to-Speech Systems and Methods of Their Assessment" by Vlado Delić et al.[ⁱⁱⁱ], we reviewed the evolution of TTS where The first electronic TTS system that developed in the mid-1960s to 1970s, while pre-recorded based systems were designed later for TTS which is still in very high demand. Their paper tells about the architecture of the TTS system as shown below:

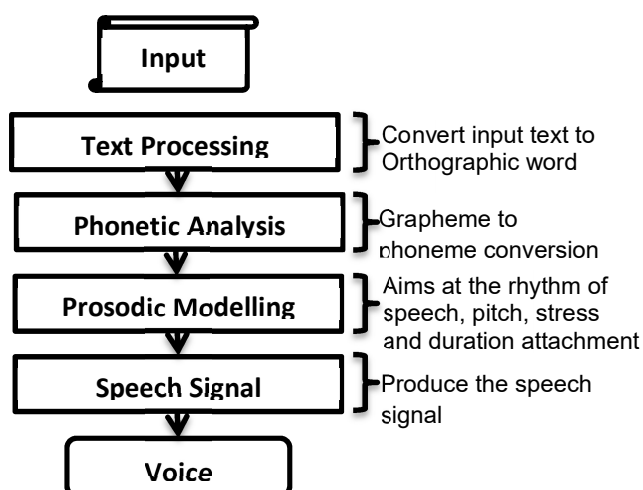


Figure 2 Architecture of the modern TTS system

In the case of a standard vocal tract model, around 20 multiplications are required to create a single voice sample. 'The initial stage in a text-to-speech conversion system is to translate such data into a standard text format' [^{iv}]. For text to phoneme transcription, the decision trees' strategy performs well. Still, 'the disadvantage is that as the size of the dictionary that [^v] is used to train the decision trees rises, so does the memory load on the decision trees'. Due to this problem, the neural network has become one of the popular ways for text to phoneme conversion because their size is not directly proportional to the size of the training [^{vi}] vocabulary. A voice synthesis system, by definition, produces synthetic speech. In paper "Text to Speech: A Simple Tutorial" [^{vii}] discussed the approaches and architecture of the TTS system shown below:

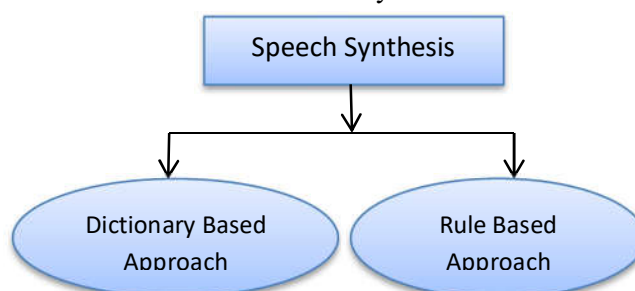


Figure 1 Speech Synthesis Approach

In 1973, [^{viii}] WILLIAM A. AINSWORTH worked on English text-to-speech conversion with a low-cost computer and a small amount of saved data. The text is divided into breath groups, the orthography is converted into a phonemic representation, lexical stress is assigned to appropriate syllables, and the resulting string of symbols is converted into parameter values for controlling an analog speech synthesizer using a synthesis by-rule [^{ix}]. The quality of the synthesized speech is assessed via listening tests.

In 2016, Priyanka M.C et al. worked with Praat software. The system is set up at the sentence level, with alphabets and words recorded and saved in a dictionary [^x] This paper focuses on improving the system's understanding ability and naturalness.

Nimisha Srivastava and colleagues worked on the TTS system for three Indian languages, namely Hindi, Malayalam, and Bengali, in the year 2020. They make available a text-to-speech corpus for several Indian languages. They train a neural text-to-speech model to achieve high-quality speech in these languages [xi].

3. Approaches for TTS conversion

In a **dictionary-based approach**, a dictionary is preserved where it saves all sorts of words with their correct pronunciation; it is just a question of checking in the dictionary for each word to be spelled out with proper pronunciation. It is fast and accurate with better pronunciation quality. The main disadvantage is that an extensive database is required to hold all words, and the system will stop if a term is not discovered in the dictionary [xii].

[xiii] In the **Rule-based approach**, Pronunciation rules are enforced to words to establish their terms based on their spellings. The advantage of this is that no database is required and will work with any input.

4. Methodology Analysis

The text is usually pre-processed in its initial stage in most text-to-speech conversion systems. The pre-processing stage is necessary before going further into next stage as it cleans the data before giving it to next module. The text is then segmented to differentiate the characters. Each word in the input text is converted into its corresponding phoneme. The text is then converted into speech.

Three techniques are often used for the TTS system, which are listed below:

1. Articulatory synthesis
2. Formant synthesis
3. Concatenative synthesis

Articulatory synthesis refers to computational speech synthesis techniques based on models of the human vocal tract and the articulation processes [xiv]. Its goal is to simulate the neurophysiology and biometrics of speech production computationally. Although the output of this synthesis is understandable synthetic speech, it is still far from natural sound.

The foundation for **Formant synthesis** is the speech source-filter model. Individual speech segment representations are parametrically stored in this system. Each parameter only has a single value. This refers to speech with a single acoustic segment.

Concatenative synthesis is a method of creating sound by concatenating recorded sound samples known as units. Unit duration is not strictly defined and can vary from 10 ms to 10 seconds depending on implementation. It is used in speech synthesis to generate a user-specific sound sequence from a database constructed from recordings of other sequences. The required speech units, such as syllables, diaphones, or phonemes, phonemes for concatenation, are extracted from the created inventory so that their context in the sentence under construction matches that in which they were recorded. The data flow diagram for the development process is shown below:

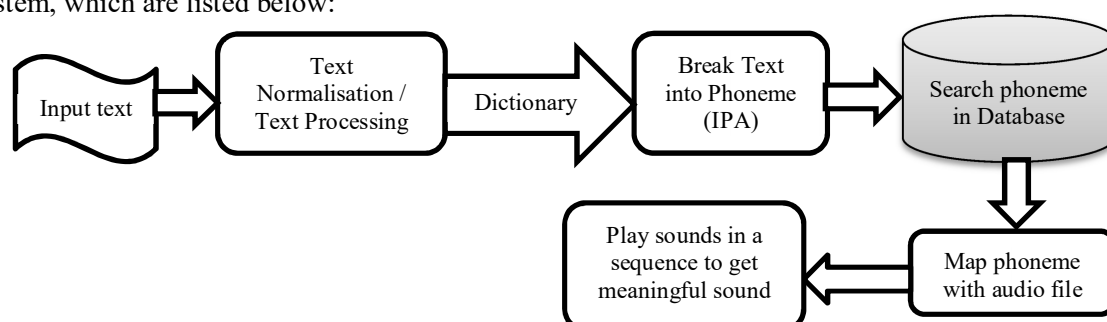


Figure 3 Data Flow Diagram of development process

Following is the flow chart where flow control of data and the next step can be easily seen

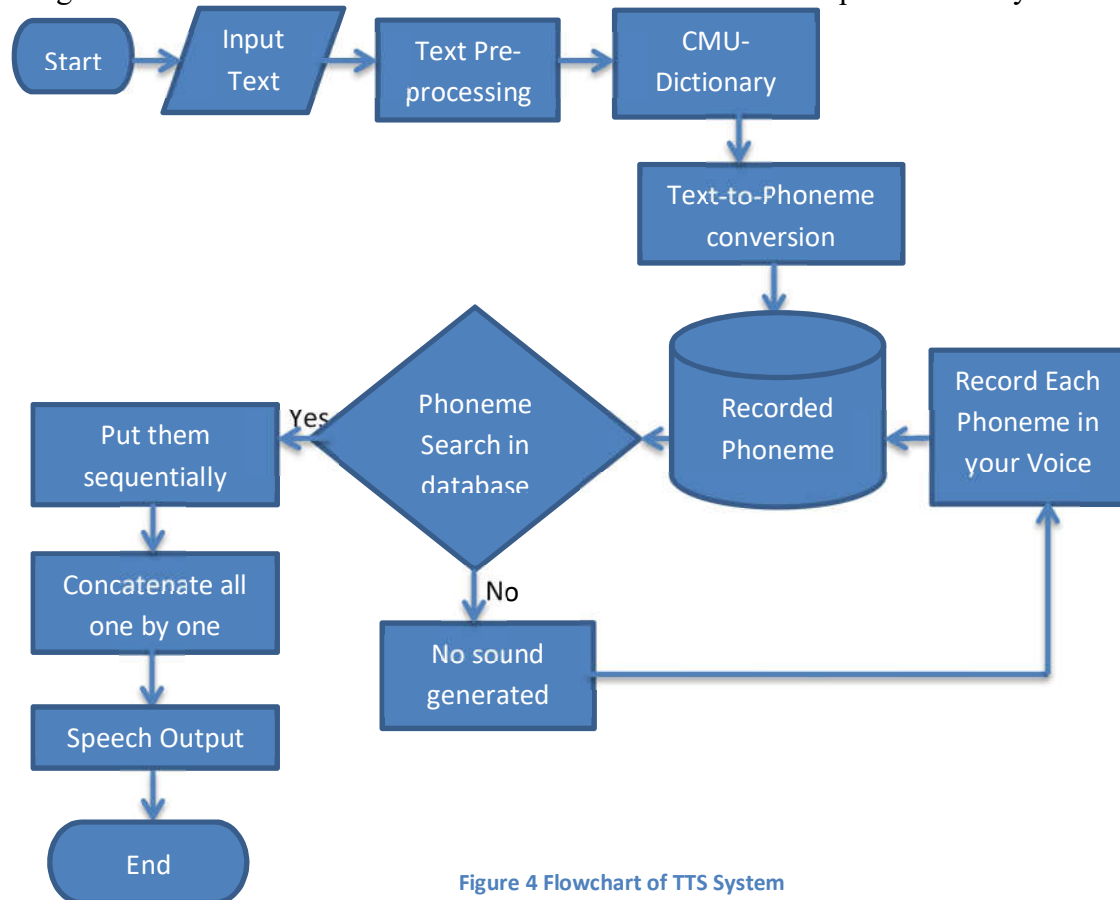


Figure 4 Flowchart of TTS System

4.1 Database Creation

To prepare a database where the phonetic sounds will be recorded and stored, the software that I will be using is **Audacity 3.1.3**. It is a free, open-source, cross-platform sound recording and editing software. For recording voice, a good quality Microphone is used, and all voices will be recorded in an empty room to avoid the unwanted sounds in the background. The database is created manually in my voice.

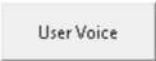
For developing this tool, the model that I have used is the Prototype model; according to this model, before creating the actual application or tool, a working prototype of the system must be developed. Compared to existing software, it is a toy or dummy model of the system, with restricted functional capabilities. Instead of creating specifications for a hypothetical system, prototyping is used to develop specifications for a real-world working system



Figure 5 Fig 10 Front look of the system

Text-To-Speech Section of Tool

Here I have provided six buttons where each button has a different role. Here all the written text will be converted into its corresponding speech. Since the database has already been created using Audacity software which we have discussed in the previous chapter, now accessing that database with the help of the program. CMU Dictionary is used for breaking the words into their corresponding phoneme.

- By clicking on the  button, you can listen to the written English text in the User's voice.

The entire English sentences that is shown in the TTS section will be broken into the following phoneme

Phonemes are:

['HH', 'IY', 'Z', 'B', 'OW', 'L', 'IH', 'NG',
'AH', 'G', 'UH', 'D', 'L', 'AY', 'N',
'HH', 'IY', 'Z', 'B', 'OW', 'L', 'IH', 'NG',
'G', 'UH', 'D',
'W', 'AH', 'T', 'AH', 'M', 'AE', 'CH',
'G', 'UH', 'D', 'M', 'AE', 'CH',
'AY',
'B', 'AY']

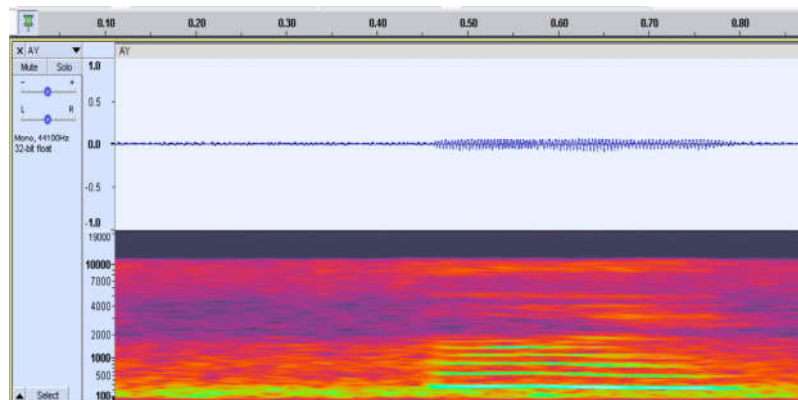


Figure 6 Phonetic Sound of Ay with its wave and Spectrogram plot

Word "Eye "consists of single phoneme ay, so let's analyze its frequency by Spectrum algorithm using Hann window function

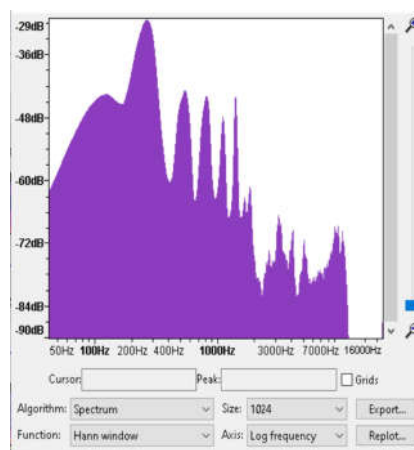


Figure 7 Phoneme Ay

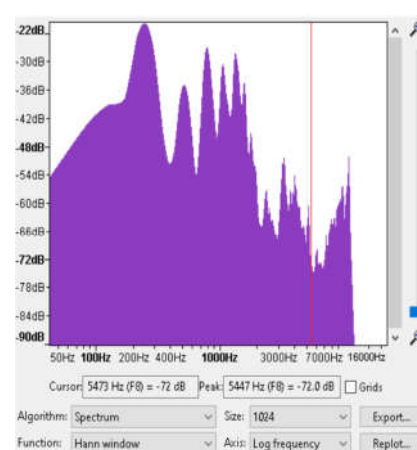


Figure 8 Word Eye

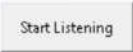
Speech-To-Text Section of Tool

Speech recognition systems process and learn spoken words before utilising computer algorithms to convert them to text.. A


programme translates sound acquired by a mic into a written language that machines and people can understand, following these four steps:

- 1) Play the audio file.
- 2) Break it down into pieces.
- 3) Transform it into a machine-readable format.
- 4) Using an algorithm, match it to the most relevant text representation.

In this section, whatever system will listen it will convert it into text, and the written text will appear in the text box section. When the

 button is pressed, the system will start listening to the input audio coming to the system. The initial component of voice recognition is, of course, speech. The actual sound is converted to an electrical signal by a microphone, then converted to digital data by an analog-to-digital converter. After the audio has been digitized, a variety of models may be used to convert it to text.

View Summary and Download Report Section of Tool

By clicking on the  button, this section will generate the summarized text

of the Speech-to-text section. Whatever system will listen to will be shortened to pass the intended message. It will be helpful when a person is not interested in reading the whole long text; instead, they want to know the summary of the very long text. The primary purpose of text summarising is to extract the most accurate and valuable information from a large document while eliminating irrelevant or irrelevant material.

^{xv} The process of text summarization is crucial in NLP with a wide range of applications. Extraction and abstraction are the two most frequent methods for text summary. To create an outline, extractive approaches select a portion of the original text's real words, phrases, or sentences.. Abstractive methods, on the other hand, create an internal semantic representation, then use natural language generation techniques to provide a summary. A summary like this will likely include words that weren't included explicitly in the original study.

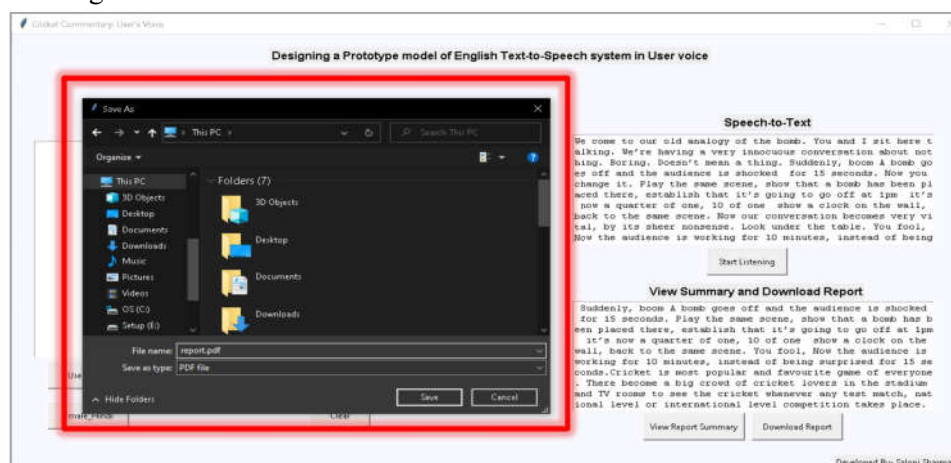



Figure 9 Saving the downloaded report in the system

By clicking on the  button, the tool will automatically start downloading the summarized report in PDF format. This generated report will be saved at your desired location in your

system with the name ending with ".pdf" format, then click on the "Save" button. You have now downloaded the report summary at your location. The below figure shows the illustration of saving the PDF report

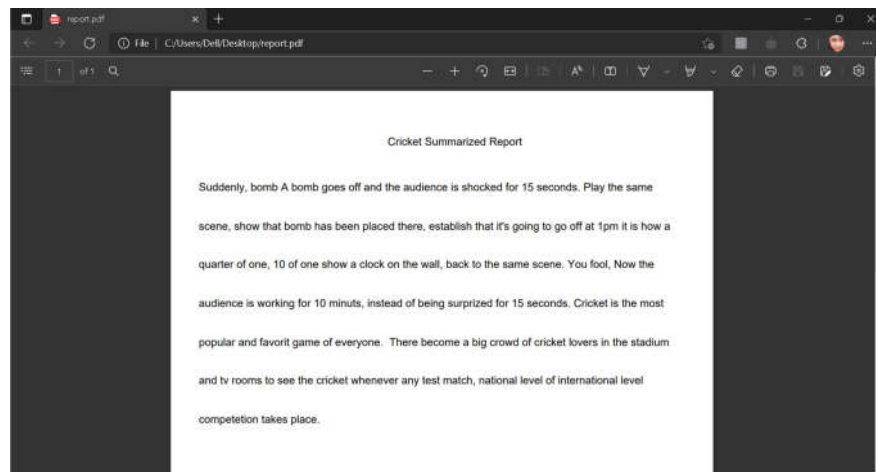


Figure 10 PDF format of the Summarized Report

5 Conclusion and remark

In this project, we have proposed a system that will help people to understand written text efficiently in their accent in English. This system will help overcome some drawbacks that were earlier faced by the people who cannot understand the synthesized auto-generated English speech generated by the system. To use this tool, users have to first cooperate with the developer to record the phonemes in their voice and accent. Words are broken into their corresponding phoneme, and these phoneme sounds are recorded using software and a microphone which later be concatenated using concatenative synthesis to produce meaningful speech. The delay between each phoneme

pronunciation is decreased to increase the naturalness of the speech.

The proposed system includes services like listening not only to texts in the User's voice but can listen to the auto-generated synthesized different Male/Female voices. Also, this system includes a Speech-to-text service where audio speech will be converted into text, which can be summarized into its brief format to know the summary of the speech, which can be downloaded into the User's system for future reference as per their requirement. Further services can also be added according to the User's needs in the forthcoming years.

References:

- ⁱ Delic, "Evolution of Text-to-Speech Systems and Methods of Their Assessment."
- ⁱⁱ Embrechts and Arciniegas, "Neural Networks for Text-to-Speech Phoneme Recognition."
- ⁱⁱⁱ O'Malley, "Text-to-Speech Conversion Technology."
- ^{iv} Bilcu, Suontausta, and Saarinen, "A Study on Different Neural Network Architectures Applied to Text-to-Phoneme Mapping."
- ^v Bilcu, Suontausta, and Saarinen.
- ^{vi} Sasirekha and Chandra, "Text to Speech: A Simple Tutorial."
- ^{vii} Ainsworth, "A System for Converting English Text into Speech."
- ^{viii} "31cbc6513755b4e9c2f0b065a194dc27.Conversion of English Text to Speech with Increasing Intelligibility.Pdf."
- ^{ix} "31cbc6513755b4e9c2f0b065a194dc27.Conversion of English Text to Speech with Increasing Intelligibility.Pdf."
- ^x Srivastava et al., "IndicSpeech"; Anto and Nisha, "Text to Speech Synthesis System for English to Malayalam Translation."
- ^{xi} Srivastava et al.
- ^{xii} "Speech Synthesis."
- ^{xiii} "Speech Synthesis."
- ^{xiv} Siddhi, Verghese, and Bhavik, "Survey on Various Methods of Text to Speech Synthesis."
- ^{xv} Gudivada, "Chapter 12 - Natural Language Core Tasks and Applications."