American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-6, Issue-3, pp-68-70
www.ajer.org
Research Paper Open Access

Voice Controlled Device to Read out Files and Control its Pace

Srihari Vasudevan[1], Sreenaath G.R.[2], Shree Sai Krishna R[3], M. Chithambarathanu[4]

[1,2,3]*(Computer Science and Engineering, Rajalakshmi Engineering College, Tamil Nadu, India)*
[4]*(Assistant Professor, Computer Science and Engineering, Rajalakshmi Engineering College, Tamil Nadu, India)*

ABSTRACT : *The aim of the project is to design and create an Internet of Things (IoT) device that can read out the documents, books or pdfs and control the flow of reading using Natural Language Processing. It enables users to listen to the audio read by the device and control the order of cascading while reading. The device also enables the users to mark and take notes of a concept and also review it later. This can be helpful for blind people in studying concepts and controlling the order in which they read.*

Keywords: *TTS (Text to Speech), STT (Speech to Text).*

I. INTRODUCTION

One of the most common method for communication between people is Speech. Speech being the base for language and communication is crucial for reading, learning, socializing, and understanding and expressing emotions and feelings. Most of the information that are shared on digital media is available only to a few who can read and understand them through a scrupulous language. To overcome this, the text can in turn be processed and delivered using text to speech conversion in such a way that many peoples can share and understand them. Text to Speech Services understands the text and uses natural language to generate synthesized audio output with appropriate cadence and intonation. Our main intention is to provide a system through which blind people can get the knowledge of text they can't see. Blind people use braille to learn the words, but a study states that only 10 % of them know to read braille. So the text to speech can be a good alternative for the blind, further by implementing speech to text they can interact easily with the system. Speech processing is the process of differentiating speech signals and converting these signals to text keywords so the machine can process them accordingly. The speech recognition is the important function. For Speech to Text we use Watson, a STT engine. The audio file which is to be converted is saved in .wav format. Specific words have been used as the keywords so whenever a user speaks the word corresponding action will be performed. For example, the keyword START will start reading a text file.

II. IMPLEMENTATION

1. **Speech to text**:

Speech is the best platform for human computer interaction. Since it is hand free it does not require any typing and only needs a microphone or a mic as an input. In speech conversion the command from the user is recorded and saved as a .wav file and is then processed by the STT engine to extract text. We use python coding platform to do this process. The python code connects the audio file to IBM Watson cloud where the audio file is decoded using the speech to text conversion module. The text is then stored in the form of a text file.

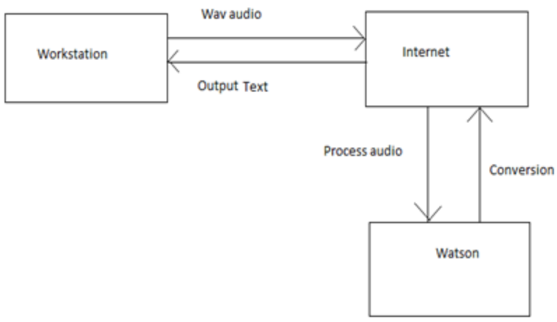

Fig.1

2. **TextToSpeech**:

Now a days all the books and information are made digital so the text to speech can be of use to everyone in their day to day life. This benefits blind people the most as they can get the information they need just by listening to the text. The Text to Speech conversion is done using a TTS Engine called “espeak”. This TTS engine will parse the document type file and convert them to a .wav file which will then be heard by the user.

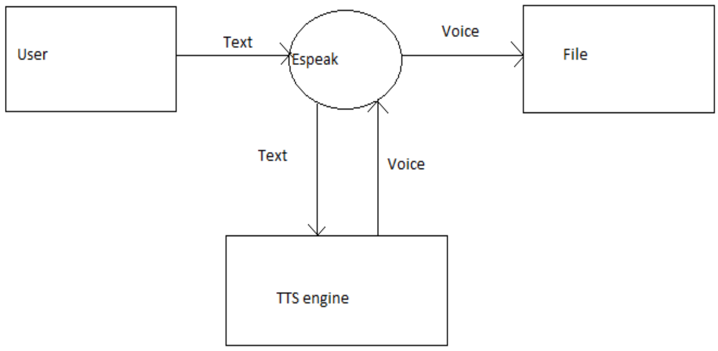

Fig.2

III. ARCHITECTURE

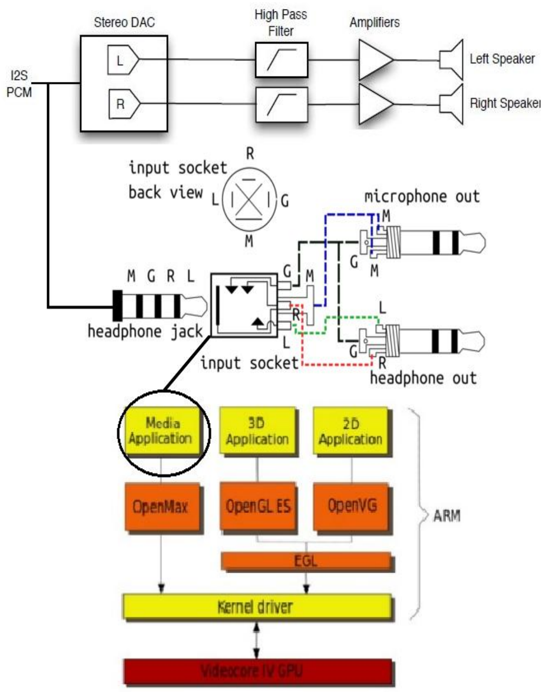

Fig.3

The architecture above can be split into three parts: Raspberry-Pi Architecture, Microphone Architecture and Speakers Architecture. In the Raspberry-Pi Architecture, the Media Application layer consists of an on-board chip which has ports to connect a microphone and a speaker to record and play audio. This helps to establish a connection for the same so we can use it for our communication purposes. The Speaker Architecture consists of various audio filters, amplifiers, etc. to make the audio more audible. The Microphone Architecture has mic tat records the various sounds that are produced by and around the user. Thus the whole architecture required for the functioning of the project is constructed.

IV. ALGORITHM

1. Start by downloading the text file to the system
2. Load or pass the file to be read to python to parse line by line and save as an array "**Text[ ][ ]**"
3. Cascade through the array "**Text**" such that

    for (i=0 to range(Text))

    Read **Text[i]** using eSpeak to the user

    Pause for 3 secs while recording for the user to say any command

    If (voice detected) then go to step 4

    Continue till **i = range (Text)**

    Goto step 6
4. **If (voice detected)**

    Receive the voice input

    Convert the voice input to .wav file

    Pass the obtained .wav file to Watson using the tts engine and wait for a response

    Save the response to a variable "**Response**"

    Compare the text in "**Response**" with the database to check if it matches any commands

    If (command found) then go to step 5

    Else Continue
5. **If (command found)**

Execute the appropriate command (Start, Stop, Pause, Resume, Next, Previous, Note, Revise, Delete, Continue)

a) Start – Start reading the Text file from the beginning.
b) Stop – Stops reading a Text file.
c) Pause – Pauses reading a Text file at that sentence.
d) Resume – Continue reading a Text file from the last paused spot.
e) Next – Skips to the next sentence.
f) Previous – Repeats the previous sentence.
g) Note – Takes note of the current sentence for future reference.
h) Revise – Reads out all the noted sentences.
i) Delete – Removes the noted sentence.
j) Continue – Continue reading the Text file after Revising.

6. Using eSpeak say "You have reached the end of the file"
7. Stop

V. CONCLUSION

- In this paper we have worked on two modules namely text to speech and speech to text. For text to speech we have used a TTS engine called "espeak" which takes English words and converts the text to voice related sounds with added features such as accent, frequency, and speed.
- For speech to text we have used a STT engine called IBM Watson which takes in audio files in .wav format which is one of the basic audio file format. These wav files contain recorded speech which are then processed by Watson to give text by cross referencing with predefined English dictionary.

ACKNOWLEDGEMENTS

Our hearty thanks to Mr. M. Chithambarathanu, AP/SS, and Assistant Professor of our department who gave an idea on "Internet Of things" and their importance in the World of Web.

REFERENCES

[1]. Miss. PrachiKhilari and Prof. Bhope V.P A Review on Speech to Text conversion methods International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 4 Issue 7, July 2015.

[2]. NandakishoreC.Patil and YaminiD.Patil Text to Speech Synthesis for Blind People Global Journal for Research Analysis International Volume-4, Issue-8, August-2015, ISSN NO-2277-8160.

[3]. Bahl, Lalit R., Frederick Jelinek, and Robert L. Mercer. A Maximum Likelihood Approach to Continuous Speech Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 5(2) (March 1983): pp. 179-190.

[4]. Padmanabhan, Mukund, and Michael Picheny. Large-Vocabulary Speech Recognition Algorithms. Computer, Vol. 35(4) (2002): pp. 42-50.

[5]. Padmanabhan, Mukund, and Michael Picheny. Large-Vocabulary Speech Recognition Algorithms.Computer, Vol. 35(4) (2002): pp. 42-50.

[6]. Jelinek, Frederick. The Development of an Experimental Discrete Dictation Recognizer. Proceedings of the IEEE, Vol. 73(11) (November 1985): pp. 1616-1624.