Text-to-speech and Voice-to-text by using API

Appana Venkata Naga Vineethsai[1], Divyanshu[2], Jyothi Pruthi[3], Ashish Mehra[4], Akshat Sharma[5]

1,2,3,4,5Chandigarh University

***Abstract*— Communication is a critical component of growth in today's environment. It also helps us in both business and personal level, it's critical to get information to the appropriate person at the right time. The world, like the means of communication, is headed toward digitalization. In today's technologically advanced society, phone conversations, emails, text messages, and other forms of communication have become indispensable. Many apps have emerged to serve the aim of successful communication between two parties without hindrances, acting as a mediator and assisting in the effective transmission of messages in the form of text or audio signals through kilometres of networks. The majority of these applications make use of features like articulatory and acoustic-based voice recognition, conversion of speech signals to text and text to synthetic speech signals, and language translation, among others. In this review paper, we will look at the various strategies and algorithms used to achieve the mentioned functionalities.**

Keywords*— *Text to speech, Speech Synthesis, Voice to Text, Web Speech API, communication, get voices, Speech recognition.

1. INTRODUCTION

Mobile phones have evolved into an essential form of communication in modern culture during the last several years. From a source to a destination, we can easily make calls and send texts. It is widely accepted the verbal communication is the more effective means the transmitting, conceiving accurate knowledge while avoiding citation. Verbal communication over the phone may readily fill the gap when communicating over a long distance. Speech recognition technology, which converts voice conversations to text messages, has lately emerged as a game-changing advance in SMS technology. TTS, VTT, and translation are all employed in a number of applications that help the impaired. It's already used in variety of applications, such as: Siri is help users communicate with the different devices and connect with local and/or remote services more successfully. Speech synthesis is effective at converting tokenized words into synthetic human speech. In this study, several machine translation approaches and engines will be evaluated and compared. The given below are some production types for speech that examined when apps employ various speech-related functions.

- Phonation (sound)
- Fluency
- Intonation
- Pitch variance
- Voice

We used the Web Server API for text-to-speech and voice-to-speech conversion in this project. SpeechSynthesis (TTS) and SpeechRecognition (VTT) are used with the Web Server API. In web speech API the main controller interface and for grammar view, output is Speech recognition. In a lot of cases, speech recognition will be used as default recognition system for devices, now a days this speech recognition system giving commands for latest OS.

2. METHODOLOGY

In this research we are mainly forced on voice to text, speech recognition, speech synthesis and text to speech.

Application Programming Interface (API)

API stands for application programming interface, It states the rules for communication to happen, it basically takes a request and gives the response, each app gives a different kind of request like in paytm or phone pay we request for money transaction and in zomato app requests for maps from google maps API.

A. Voice To Text Method

The voice to text conversion is a technique which converts language or words spoken by human into texts. Basically this is interchangeable with speechrecognition, but latter is used, to refer to the

process of speech perception. For speech recognition, VTT use the same stages and idea but each phase uses a different combination of approaches.

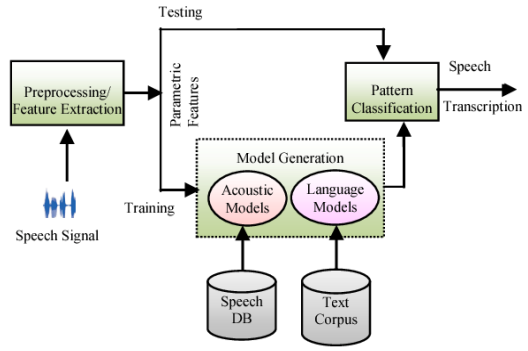

Fig.1. SpeechRecognitionProcess

SPEECH RECOGNITION

This method refers only program's or machine capability to recognise phrases and words that spoken by user and converts it into machines format as in figure 1. The following parameters [1] can be used to classify speech recognition systems.

- **Speaker:** Every speaker has different frequencies in voice. And this types are designed for only particular speaker.
- **Vocal Sound:** Speech recognition is influenced by the manner the speaker speaks. Some models can distinguish between single utterances and utterances separated by a pause.
- **Vocabulary:** The size of the vocabulary has a large impact on the system's complexity, performance, and precision.

1) **Pre-processing**: In preprocessing digital signals was converted from analogy signal for future processing. After converting this digital signal will enter to first order filters. This filters helps digital signal energy at a higher frequency.

2) **Feature Extraction:** The set of utterance parameters associated with speech signals is determined at this stage. The acoustic waveform is processed to determine these properties, which are referred to as features. The basic goal is to create a compact representation of the input signal by computing a series of feature vectors.

```
var SpeechRecognition = SpeechRecognition || webkitSpeechRecognition;
var SpeechGrammarList = SpeechGrammarList || webkitSpeechGrammarList;
var grammar = "#JSGF V1.0;";
var recognition = new SpeechRecognition();
var speechGrammarList = new SpeechGrammarList();

speechGrammarList.addFromString(grammar, 1);
recognition.grammars = speechGrammarList;
recognition.lang = "en-US";
recognition.interimResults = true;
recognition.continuous = true;


recognition.onresult = function (event) {
  var transcript = Array.from(event.results)
    .map((result) => result[0])
    .map((result) => result.transcript)
    .join("");
  message.textContent = transcript;
};
recognition.onerror = function (event) {
  message.textContent = "Error occurred in recognition: " + event.error;
  recognizing = false;
};

```

Fig.2. Speechrecognition, voice, grammarobject

Automatic punctuation & capitalization: You might not care if your transcripts are neatly formatted depending on what you want to do with them. However, if you want to make them public, having this feature included in the STT API can save you time.

Custom vocabulary: If your audio has a lot of bespoke phrases, abbreviations, and acronyms that an off-the-shelf model would not be familiar with, being able to define custom vocabulary is useful as shown in figure 2.

B. Text To Speech Method

It will convert spoken text into speech by analysing, understand, and prepared this whole process is called Text To Speech. In this method various steps have been involved and in above flow chart it shows in order. Although the most important phases of this method are [5]:

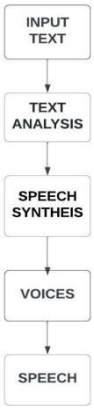

Fig.3.TTS System Flow

Text Processing: As shown in figure 3, First text will be analyse, normalised, and then written in the phonetic and linguistic forms.

Speech Synthesis: The types of speech synthesis which we are used in this project [5]:

i. **Articulator Synthesis:** To generate speech, a mechanical and acoustic model is used. Basically articulator synthesis will give speech but because of natural sound mostly it is not used by everyone.

ii. **Formant Synthesis:** Formant Synthesis stored on parametric basis by a separate speech segments. There are two basic types of structures:-

- Cascade
- Parallel.

Mainly these cascade and parallel structures are help to increase performance only.

The cascade synthesiser is made by series of band pass resonators and this cascade required only format frequencies as shown in figure 4.

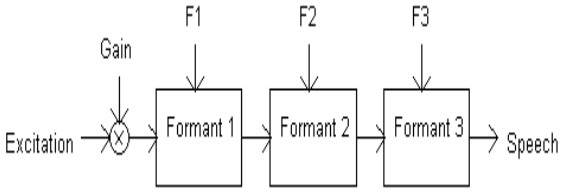

Fig.4. Basic structure of cascade formant synthesizer

The parallel synthesiser is made up of parallel resonators and as in figure 5 excitation signal is delivered at a time for all formants then output added together.

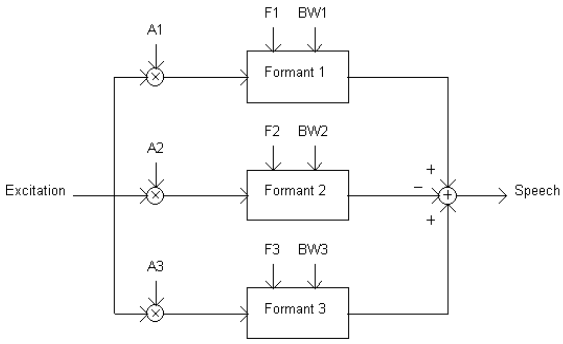

Fig.5. Basic structure of a parallel formant synthesizer

iii. **Concatenative Synthesis:** It is a method of speech generation. Concatenative speech synthesis is known to get the most natural sound in space primarily due to concatenation of national speech sound units. It is a TTS technique, where speech is synthesised by joining speed sound units together. The basic speech sound units can be a phoneme, diphone, syllable or even a word. The duration of the units is not strictly defined and may vary according to the implementation, roughly in the range of 10 milliseconds up to 1 second.

Voice function: The voices array will hold all of the voices available in the Browser Web speech API, while as mention in figure 6 the getVoices function will return all of the voices available, along with their names and languages. Finally, the voices are added to the drop-down menu on our webpage select option. The user will be able to choose their preferred voices.

```
// Init voices array
let voices = [];

const getVoices = () => {
  voices = synth.getVoices();

  // Loop through voices and create an option for each one
  voices.forEach(voice => {
    // Create option element
    const option = document.createElement('option');
    // Fill option with voice and language
    option.textContent = voice.name + '(' + voice.lang + ')';

    // Set needed option attributes
    option.setAttribute('data-lang', voice.lang);
    option.setAttribute('data-name', voice.name);
    voiceSelect.appendChild(option);
  });
};

getVoices();
if (synth.onvoiceschanged !== undefined) {
  synth.onvoiceschanged = getVoices;
}
```

Fig.6.shows the voice working

Speak Function: We'll create a function speak as shown in figure 7. Speaking events are handled by talk, which provides a wave background when a voice is speaking, manages the selected voice, controls the rate and pitch of the voice, and also handles error.

```
// Speak
const speak = () => {
  // Check if speaking
  if (synth.speaking) {
    return;
  }
  if (textInput.value !== '') {
    // Add background animation
  body.style.background =
      "#0acffe url(https://res.cloudinary.com/nonsoblip/image/upload/v1620236458/wave_n3rtre.gif)";
    body.style.backgroundRepeat = 'repeat-x';
    body.style.backgroundSize = '100% 100%';

    // Get speak text
    const speakText = new SpeechSynthesisUtterance(textInput.value);

    // Speak end
    speakText.onend = e => {
      body.style.backgroundImage = `linear-gradient(to right, #0acffe 0%, #495aff 100%)`;
      // body.style.background = '#141414';
    };

    // Speak error
    speakText.onerror = e => {
      console.error('Something went wrong');
    };

    // Selected voice
    const selectedVoice = voiceSelect.selectedOptions[0].getAttribute(
      'data-name'
    );

    // Loop through voices
    voices.forEach(voice => {
      if (voice.name === selectedVoice) {
        speakText.voice = voice;
      }
    });
    // Set pitch and rate
    speakText.rate = rate.value;
    speakText.pitch = pitch.value;
    // Speak
    synth.speak(speakText);
```

Fig.7.shows the speak-set rate&pitch

3. RESULT AND DISCUSSION

In this we developed the frontend by using HTML & CSS and it is connected to text to speech and voice to text by API.

We created a simple application and it is easy for users to use it as shown in figure 8.

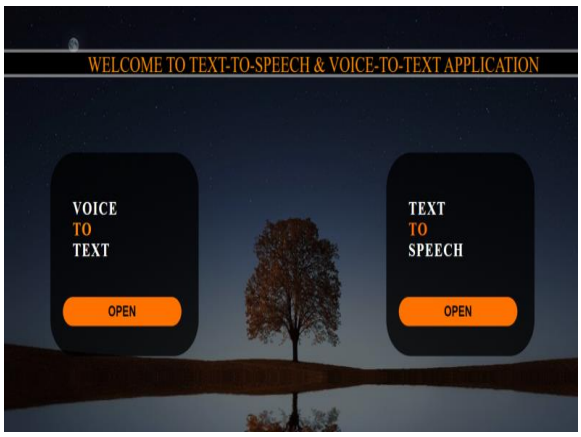

Fig.8.shows the frontend of the model

In Voice to text we used speech recognition. Here basically speech recognition converts English words spoken by human into text format. As shown in figure 9 first user needs to allow the microphone then only recording will be start, After staring recording if users wants to stop means we given option that users can copy their text till it. And also if user wants to start new conversion then we provide delete option so that users can start from starting on it.

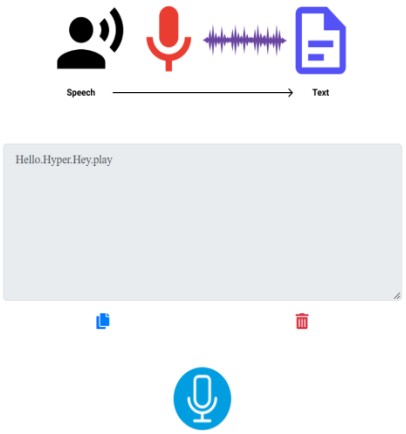

Fig.9.shows the output of VTT

In Text to speech we used speech synthesis. Here basically speech synthesis do analysing of English text and converts it into speech. First user needs to type or copy paste the text in text content box, after below that there will be rate and pitch because according to user convenience he/she will be set pitch and rate as shown in figure 10. And here contains all voices which are present in browser web speech api so from it user can select it.

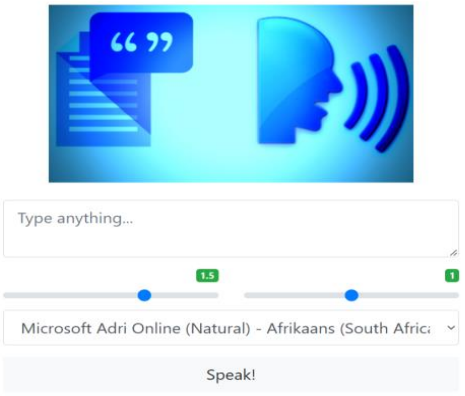

Fig.10.shows the output of TTS

4. CONCLUSION

In this project we learned the various methods for VTT & TTS as well as their applications and usage. We can draw the following conclusion after thoroughly analysing the various speech types, voice-Recognition, speech translation, conversion of VTT & TTS: In VTT, we say that, despite its drawbacks, API is a better speech signal to text converter than the other two due to its computational feasibility. Similarly, formant synthesis, which uses parallel and cascade synthesis, is the best converter in TTS systems. Speech Synthesis translation is widely used because it combines the advantages of rule-based and statistical machine translation techniques. It create and promotes the syntactically and grammatically content correctly and also considering text is in correct format, learning ability, and data acquisition.

REFERENCE

[1] Pratik K. Kurzekar, Ratnadeep R. Deshmukh, Vishal B. Waghmare, Pukhraj P. Shrishrimal, A Comparative Study of Feature Extraction Techniques for Speech Recognition System, International Journal of Innovative Research in Science, Engineering and Technology (An ISO 3297: 2007 Certified Organization) Vol. 3, Issue 12, December 2014.

[2] Ms. Anuja Jadhav, Prof. Arvind Patil, Real Time Speech to Text Con- verter for Mobile Users, National Conference on

Innovative Paradigms in Engineering Technology (NCIPET-2012) Proceedings published by International Journal of Computer Applications (IJCA)

[3] Sunanda Mendiratta, Dr. Neelam Turk, Dr. Dipali Bansal, Speech Recognition by Cuckoo Search Optimization based Artificial Neural Network Classifier, 2015 International Conference on Soft Computing Techniques and Implementations- (ICSCTI) Department of ECE, FET, MRIU, Faridabad, India, Oct 8-10, 2015.

[4] Suhas R. Mache, Manasi R. Baheti, C. Namrata Mahender, Review on Text-To-Speech Synthesizer, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 8, August 2015.

[5] Aditi Kalyani, Priti S. Sajja, A Review of Machine Translation Systems in India and different Translation Evaluation Methodologies, Interna- tional Journal of Computer Applications (0975 8887) Volume 121 No.23, July 2015

[6] Mouiad Fadiel Alawneh, Tengku Mohd Sembok Rule-Based and Example-Based Machine Translation from English to Arabic, 2011 Sixth International Conference on Bio-Inspired Computing: Theories and Applications

[7] Suman K. Saksamudre, P.P. Shrishrimal, R.R. Deshmukh, A Review on Different Approaches for Speech Recognition System, International Journal of Computer Applications (0975 8887) Volume 115 No. 22, April 2015.

[8] F. Seide, G. Li, D. Yu,Conversational Speech Transcription Using Context-Dependent Deep Neural Networks, In Interspeech, pp. 437440, 2011.

[9] Kamini Malhotra,Anu Khosla, Automatic Identification of Gender Ac- cent in Spoken Hindi Utterances with Regional Indian Accents, 978-1- 4244-3472-5/08/25.00 2008 IEEE

[10] Keiichi Tokuda, Yoshihiko Nankaku, Tomoki Toda, Heiga Zen, Speech Synthesis Based on Hidden Markov Models, Proceedings of the IEEE — Vol. 101, No. 5, May 2013. Junichi Yamagishi, Member IEEE, and Keiichiro Oura

[11] G. E. Dahl, D. Yu, L. Deng, A. Acero, Large vocabulary continuous speech recognition with context-dependent DBN-HMMs, In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4688-4691, 2011.

[12] Pere Pujol Marsal, Susagna Pol Font, Astrid Hagen, H. Bourlard, and C. Nadeu, Comparison And Combination Of Rasta-Plp And Ff Features In A Hybrid Hmm/Mlp Speech Recognition System, Speech and Audio Processing, IEEE Transactions on Vol.13, Issue: 1, 20 December 2004.