ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR1984

Artificial Intelligence Powered Voice to Text and Text to Speech Recognition Model – A Powerful Tool for Student Comprehension of Tutor Speech

Sonali Padhi[1], Kranthi Kiran[2], Ambica Thakur[3], Adityaveer Dhillon[4], Bharani Kumar Depuru[5]

[1] Research Associate, AiSPRY, Hyderabad, India
[2] Research Associate, AiSPRY, Hyderabad, India
[3] Mentor, Research and Development, AiSPRY, Hyderabad, India
[4] Team Leader, Research and Development, AiSPRY, Hyderabad, India
[5] CEO and Director, AiSPRY, Hyderabad, India

Abstract:- Speech-to-Text and Text-to-Speech are both NLP(natural language processing) powered models which transform speech to text and vice versa, providing an increased scope of learning for the parties involved. For the past couple of years it's been observed that students have been moving abroad for quality education and better financial aid. Since there is an accent gap between students and tutors which reduces the understanding of students. Our work is done to solve the aforementioned problem. With its state-of-the-art STT(speech-to-text) and TTS(text-to-speech) softwares this work intends to ease the learning curve of the students.

The key targets of this work are international students, individuals with disabilities. It can also be used to transcribe meetings for quick conversion of meeting discussion points into text. Companies can also use the model to get the data for the call recordings and further perform sentiment analysis and various such activities.

This research aims to give a detailed walk through of the product as it stands, and provide details regarding all aspects of the product. This covers the various tech stacks used, the implementation of the said technologies, the reports shown to the different end users. This provides the workflow of the product.

Keywords:- *Artificial Intelligence, Deep Learning, Large Language Models, Automatic Speech Recognition, Transcription, Whisper AI, gTTs.*

I. INTRODUCTION

Speech to text is a NLP(Natural Language Processing) powered system which aims to transcribe the speech data into text and Text to speech is a system which aims to convert text into natural sounding audio. Our model uses WhisperAI [1]which was introduced by OpenAI for transcription and gTTS [2] for producing audio. Due to its simple architecture and high accuracy it is considered by far the best model for speech recognition. Unlike traditional speech recognition models, WhisperAI uses minimal fine tuning[3]. Since it is trained on 10x more data than traditional models, the accuracy is unbeatable. It also comes with an added benefit of predicting punctuation and casing which is not possible with traditional speech recognition systems.

Ours is a system designed to understand students' difficulty in understanding the local accent[4]. It aims to solve the accent issue by using different softwares. There is a lack of such products online and especially in off-line classes. This provides the opportunity for the product to capture the niche market. There may be an increase in demand if the students are exposed to the product.

A certain project methodology was followed by 360DigiTMG called CRISP-ML(Q) [5][Fig.1]which stands for Cross Industry Standard Process for Machine Learning with Quality assurance. It was mainly designed to approach any machine learning related project.

As per CRISP-ML(Q), the first step is to gain a thorough understanding of the business objective[Fig.1]. The aim of this paper is to find a solution for students to understand the speech of the tutor.

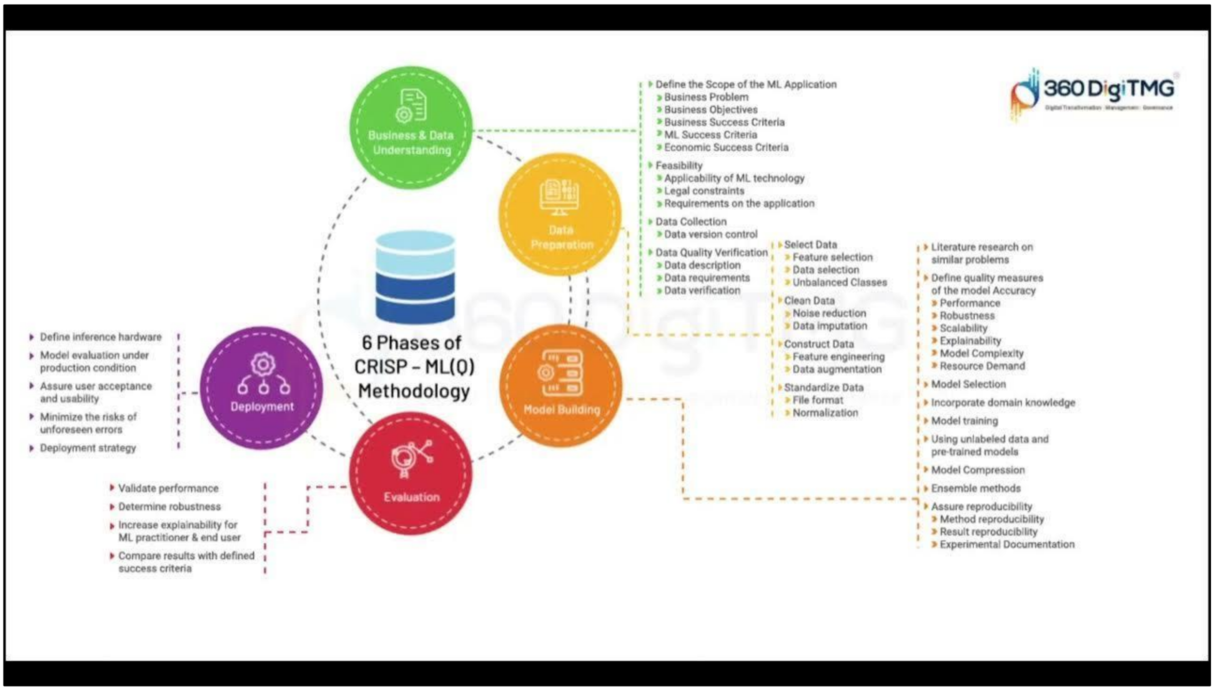

Fig.1: CRISP-ML(Q) Methodological Framework, outlining key components and steps visually (Source:- Mind Map - 360DigiTMG)

As data plays a paramount role in development of any ML or deep learning solution. Since, the goal is to solve the accent problem faced by students, the data is collected from various sources through the internet. The dataset was constructed by combining various audio files available on the internet, majorly focusing on audio with thick accent. A diversity was maintained to get a generalized transcription for different audio files. Since diversity in the dataset can make the model more robust, however, this was not the case for transcription. On initial assessment, it was observed that the transcripts generated for thick accents were of poor quality or below-average standard. To overcome this issue, several preprocessing and filtering techniques are used to boost the transcript quality.

Students travel to other places for quality education which will eventually boost their financial status. However, not able to understand the course due to different native places is a major issue most of the student faces and hence reducing their understanding

II. METHODS AND TECHNIQUES

A. Data Dimensions

The data is constructed from various sources on the internet. The main focus is on Asian accents including Chinese, Singaporean, south Asian and other accents. This results in a diverse dataset covering a broad audio distribution from various environments. However, diversity in audios can help give a robust model but diversity in transcript will only lead to misleading outputs.

Below are details of data we used to train the algorithm.

Table 1 Data used to Train the Algorithm.

Number of audio hours	20
Number of text files	20
Size of all the files	401 MB

B. Model architecture

Since the focus of the research is to come up with a good transcript for better understanding of students, a simple architecture is maintained by using some pre-trained models[6]. The main aim for the usage of pretrained models is that the model gets deployed easily on a local computer without the hassle of worrying about huge parameters.

For a high level picture the user will provide a link and the model will automatically find its corresponding transcript. The generated transcript is passed to Google text to speech which will convert the texts into spoken audio files.

Since the model uses WhisperAI for the transcription, its architecture uses the encoder-decoder part of the Transformer[7]. The audio is resampled to 16000 Hz and an 80 channel log-magnitude mel spectrogram is computed on 25 millisecond windows with 10 milliseconds stride. Once a log mel spectrogram[8] is computed, the input then passes through 2 convolution layers having filter width of 3 and GELU[9] function. The two convolution layer has 0 and 2 stride respectively. The output is passed to a positional encoding after which the encoder of Transformer follows. Encoder output is normalized before passing it to decoder blocks. The decoder uses position embeddings and the token representations. The encoder and decoder have the same number of blocks.

Once the transcript is generated using the above model, it is made to pass through a Google Text-to-Speech (gTTS)[10], a Python library and unofficial Google API for converting text to speech. Once a request has been made by the user for the input text, the API returns audio data in the requested format. The supported formats are mp3 or wav as provided. The language of the audio file can be set by using the hyper parameter ‘lang’. The whole flow is automatically done using pipelines.[Fig.2]

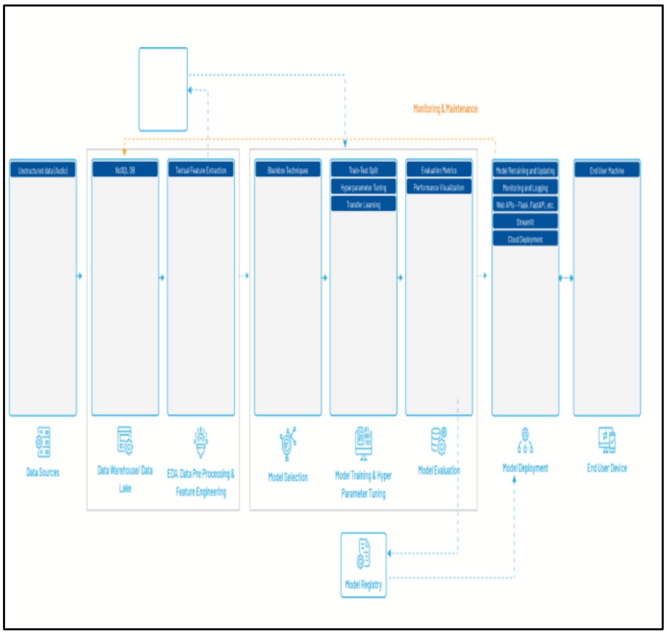

Fig.2: Architecture Diagram: Explain the Workflow of the Text-To-Speech and Speech-to-Text Module
(Source: ML Architecture Assistant - 360DigiTMG)

C. Data Preprocessing

For a simpler approach the data preprocessing is done using NLP techniques[11]. Once transcription is generated we sort out data preprocessing which follows multiple steps starting from tokenization, stemming and bag of word representation. Each sentence is broken down into word tokens and special characters are removed using the Regex library[12] in Python, a powerful library to normalize any text. Once our tokens are generated, base word is found using stemming. To understand the patterns in the text, we visually represent the words using Word cloud.

Since the aim is to convert the text into audio, we further do some simple audio preprocessing to understand the nature of sound visually. For doing this, Librosa[13] is used. It is a powerful package for music and audio analysis, widely used in fields such as music transcription, speech recognition and information retrieval. Once the audio signals are loaded using Librosa, it converts the audio files into NumPy arrays making further computation easy and fast. After the audio signals are converted into numerical representation, a waveform can be plotted to visually compare the original audio and the generated audio. We further plotted a spectrogram to retrieve the frequency content of a signal as it changes over time. For feature extraction MFCC[14] is used. It stands for Mel-Frequency Cepstral Coefficients. It is majorly used in speech recognition and speaker identification tasks. Derived from short-time Fourier transform (STFT), it is used for spectral characteristics of the signal. The x-axis represents time index and y-axis represents MFCC coefficients. The color intensity represents the value of the corresponding MFCC coefficients. The higher intensity represents higher MFCC values and the lower intensity represents lower MFCC coefficients. Overall MFCC graph helps to identify patterns in MFCC values for future interpretation.[Fig.3]

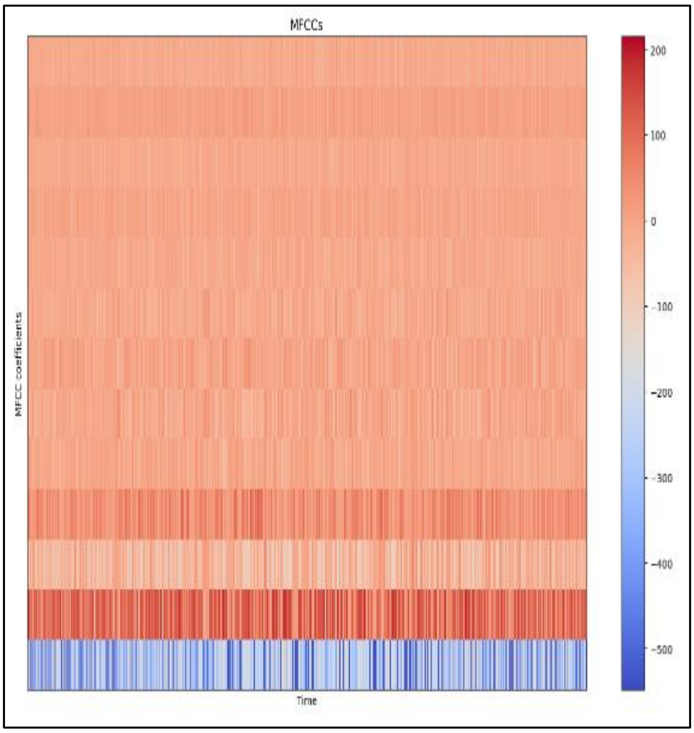

Fig. 3: MFCC Coefficients Chart, showing How the Spectral Information of the Sound and Showing the Power Spectrum for any Analyzed Sound

D. Deployment

For easy usability and accessibility, organizations use cloud deployments. Here, we used AWS as it provides a wide variety of tools. One major aspect when it comes to deployment is to avoid data leakage. AWS comes with powerful data security along with not having to deal with investing in hardware.[Fig.2]

E. Implementation

The model is a solution which translates based on the understanding of the local accent. It generates output as well as learns from the user input and user reported errors. The product is deployed in amazon web services(aws), it allows users the freedom of deployed region, scale, security, isolation etc. The model can be run on both cpu and gpu which offers different options for end users as well as cost saving measures.

The product caters to various students, teachers, educational institutions and other interested groups. Such models also learn from constant feedback loop which makes it better as time passes. This allows the students to focus on studying and improving their grades and allows for much better communications between students and teachers, thus leading to overall development in the society. The models used in the paper are open source platforms, this reduces the overall time and cost taken as we train only on small dataset using transfer learning.

- Frontend: it is a user interface designed using streamlit application in which the user either uploads either text or audio
- Middleware: amazon web service (aws) is where the all compute takes place, it is the heart of the project
- Backend: this is the place where we upload all the files including python scripts, models and their weights as well as streamlit files.
- Database: It is used to store both structured and unstructured data. NoSQL uses unstructured data while SQL uses structured data.

III. RESULTS AND DISCUSSION

A good model is that which will generate a high-quality transcription. A detailed report will help the student to grasp good knowledge of a particular topic. The end result generates a transcription along with an audio file. Any accent is stripped off in the audio. To boost the confidence of the student regarding the reliability of the system, accuracy is shown at the end.

The overall workflow starts by asking the user to enter a URL of the audio file. Once the corresponding file is received, transcription takes place. Here, whisper API is used to access the speech-to-text model. The API key used is a paid key and hence poses a usability challenge. To deal with video files FFmpeg is used which is a powerful open-source software known for its audio extraction capabilities. The audio file is then sent to the Whisper model to get the transcription. [Fig.2]

WhisperAI, uses word error rate[15] to calculate the loss. To find the loss the transcription and original text is taken. The formula is given in the following figure. It is the most common way to calculate the transcription quality. We achieved a WER of 37 with our 20 hours of training data. [Fig.4]

$$\mathbf{WER} = \frac{\mathbf{S + D + I}}{\mathbf{N}}$$

where...
S = number of substitutions
D = number of deletions
I = number of insertions
N = number of words in the reference

Fig.4: The Image here is the Formula for Word Error Rate used for ASR
(Source - https://sonix.ai/articles/word-error-rate)

Once transcription is generated, our next aim is to take the transcription and generate an audio. For a seamless flow, gTTs is used. The output will show a transcription, followed by an audio file and accuracy. Here, we have provided both transcription and audio for better understanding.

We measure the quality of the synthetic audio generated with the metric called Mean Opinion Score(MOS)[16] It is an arithmetic mean of rating given by human experts who give their opinion after they listen to the sample audio based on the naturalness of the audio and intelligibility. it is of a scale from 1 to 5 where 1 means bad and 5 means excellent[Fig.5]

$$MOS = \frac{\sum_{n=1}^{N} R_n}{N}$$

Where R are the individual ratings for a given stimulus by N subjects.

Fig. 5: The Image shown here is the Formula for Mean Opinion Score
(Source - https://en.wikipedia.org/wiki/Mean_opinion_score)

We had around 4 people who listened to the audio of different parts and rated them based on their naturalness, clean accent, intelligibility, pleasantness, articulation, overall impression etc…[17] We achieved a MOS of 4.2. [Fig.5]

IV. FUTURE SCOPE

To ensure the model works well in the real world, it has to be trained and fine-tuned on larger and much more diverse dataset. We recognized that at least 1000 hours of training is required. More training will help the model reduce WER[Fig.4] as close to zero as possible. For a robust model which delivers high quality content, further fine tuning is required to meet the performance standards.

To optimize model performance for real-world applications, we recognize the need for further training on a larger and more diverse dataset, ideally encompassing at least 1000 hours of data. By expanding the training data, we aim to enhance the model's accuracy, aiming for a Word Error Rate (WER) as close to zero as possible. Additionally, we plan to fine-tune subsequent models to ensure they meet our performance standards. This approach ensures that our models are robust and capable of delivering high-quality results in various contexts.

V. CONCLUSION

The implementation of this paper aims for a future where children don't have to experience any hardships when they pursue education. as more of the youth around the globe starts to pursue higher studies from the best available educational institutions and travel both local and abroad to pursue them. There are also online courses where we can access the best education without language barriers. Our paper aims to bridge the skill gap with the use of artificial intelligence and transform the way students learn. This paper is only a fist step towards where difference of language and accent does not matter.

REFERENCES

[1]. Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023, July). Robust speech recognition via large-scale weak supervision. In International Conference on Machine Learning (pp. 28492-28518). PMLR.

[2]. gTTS — gTTS documentation

[3]. Chang Jungwon, Nam Hosung. Exploring the feasibility of fine-tuning large-scale speech recognition models for domain-specific applications: A case study on Whisper model and KsponSpeech dataset. Phonetics Speech Sci. 2023;15(3):83-88. https://doi.org/10.13064/KSSS.2023.15.3.083

[4]. Sally Boyd (2003) Foreign-born Teachers in the Multilingual Classroom in Sweden: The Role of Attitudes to Foreign Accent, International Journal of Bilingual Education and Bilingualism, 6:3-4, 283-295, DOI: 10.1080/13670050308667786

[5]. Studer, S.; Bui, T.B.; Drescher, C.; Hanuschkin, A.; Winkler, L.; Peters, S.; Müller, K.-R. Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology. Mach. Learn. Knowl. Extr. 2021, 3, 392-413. https://doi.org/10.3390/make3020020

[6]. Qian, Yao & Bianv, Ximo & Shi, Yu & Kanda, Naoyuki & Shen, Leo & Xiao, Zhen & Zeng, Michael. (2021). Speech-Language Pre-Training for End-to-End Spoken Language Understanding. 7458-7462. 10.1109/ICASSP39728.2021.9414900.

[7]. Verma, P., & Berger, J. (2021). Audio transformers: Transformer architectures for large scale audio understanding. adieu convolutions. arXiv preprint arXiv:2105.00335.

[8]. A. Meghanani, A. C. S. and A. G. Ramakrishnan, "An Exploration of Log-Mel Spectrogram and MFCC Features for Alzheimer's Dementia Recognition from Spontaneous Speech," 2021 IEEE Spoken Language Technology Workshop (SLT), Shenzhen, China, 2021, pp. 670-677, doi: 10.1109/SLT48900.2021.9383491. keywords: {Neural networks;Speech recognition;Predictive models;Mel frequency cepstral coefficient;Root mean square;Spectrogram;Dementia;log-Mel spectrogram;MFCC;transfer learning;Alzheimer;dementia;MMSE;CNN;LSTM;ResNet18},

[9]. Hendrycks, D., & Gimpel, K. (2016). Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415.

[10]. Mankar, Shruti & Khairnar, Nikita & Pandav, Mrunali & Kotecha, Hitesh & Ranjanikar, Manjiri. (2023). A Recent Survey Paper on Text-To-Speech Systems. International Journal of Advanced Research in Science, Communication and Technology. 77-82. 10.48175/IJARSCT-7954.

[11]. Davide Falessi, Giovanni Cantone, and Gerardo Canfora. 2010. A comprehensive characterization of NLP techniques for identifying equivalent requirements. In Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '10). Association for Computing Machinery, New York, NY, USA, Article 18, 1–10. https://doi.org/10.1145/1852786.1852810

[12]. Uzun, Erdinç & Yerlikaya, Tarık & Kirat, Oğuz. (2018). Comparison of Python Libraries used for Web Data Extraction. 24. 87-92.

[13]. McFee, Brian & Raffel, Colin & Liang, Dawen & Ellis, Daniel & Mcvicar, Matt & Battenberg, Eric & Nieto, Oriol. (2015). librosa: Audio and Music Signal Analysis in Python. 18-24. 10.25080/Majora-7b98e3ed-003.

[14]. Tiwari, Vibha Tiwari. (2010). MFCC and its applications in speaker recognition. Int. J. Emerg. Technol.. 1.

[15]. Ahmed Ali and Steve Renals. 2018. Word Error Rate Estimation for Speech Recognition: e-WER. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 20–24, Melbourne, Australia. Association for Computational Linguistics.

[16]. Streijl, R.C., Winkler, S. & Hands, D.S. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. Multimedia Systems 22, 213–227 (2016). https://doi.org/10.1007/s00530-014-0446-1

[17]. M. Seufert, "Fundamental Advantages of Considering Quality of Experience Distributions over Mean Opinion Scores," 2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX), Berlin, Germany, 2019, pp. 1-6, doi: 10.1109/QoMEX.2019.8743296.