Advanced Fluency Augmentation Framework for Stuttered Speech Recognition and Articulation Correction

*Jagadisha K R., Anjali Manoj Phadthare., Harshitha N P., Likhitha S., Bhavana H L

Department of E&EE, Sri Siddhartha Institute of Technology, SSAHE, Tumkur

ABSTRACT

Stuttering which is also known as stammering, is a speech disorder in which people suffer while communicating with disorders like prolonged words, syllables or phrases, repetitions and also sometimes stop while speaking or make no sound for a certain syllables. Current speech recognition systems such as Google Assistant, Apple's Siri etc have very efficient Speech recognition system for normal speech, but they fail to recognise when the person stutters. This paper proposes to create a better algorithm for improving speech recognition in stuttering people. The suggested method uses an amplitude threshold obtained through neural network analysis to preprocess speech input in order to identify and eliminate disfluencies [2]. The major problem until now is that a system stops recognizing after a pause is encountered in the speech and hence, the average accuracy of stuttered speech recognition is around 70%. With a new algorithm which will take into account the words or characters after the pause and then use that also for recognition, the accuracy can be improved. This system is implemented in five stages namely, Amplitude thresholding and filtering, Silence ejection, speech to text conversion, repetition removal and text to speech (TTS) conversion.

Keywords: Stuttered Speech, Speech Recognition, Neural Networks, Amplitude Thresholding, Signal Processing.

INTRODUCTION

Stuttering is a communication disorder which causes sounds, syllables, or words to be repeated and lengthened without meaning to, which can make speech less fluent. These kinds of issues with speech can make it hard to communicate with other people and make speech-based technologies like virtual assistants and voice-to-text systems less useful. Speech recognition models that are already out there work best with fluent speech and have trouble accurately processing stuttered input. The Advanced Fluency Augmentation Framework for Stuttered Speech Recognition and Articulation Correction desires to help people who stutter by making their speech clearer and easier to understand. This framework includes algorithms for detecting and fixing disfluencies in real time that focus on long sounds and repeated words. This makes both articulation and speech-to-text conversion better. The proposed framework makes voice technologies more accessible and inclusive by fixing the problems with existing systems.

Importance of Stuttered Speech Correction in Speech Recognition as speech recognition becomes more common in daily life, it's important that these systems work for everyone—including people who stutter. traditional models often struggle with disfluent speech, leading to poor user experiences and limited accessibility. by correcting stuttered speech in real time, we can improve recognition accuracy and help individuals with speech impairments interact more naturally with voice-based technologies. this not only enhances usability but also supports more inclusive and equitable human-computer interaction.
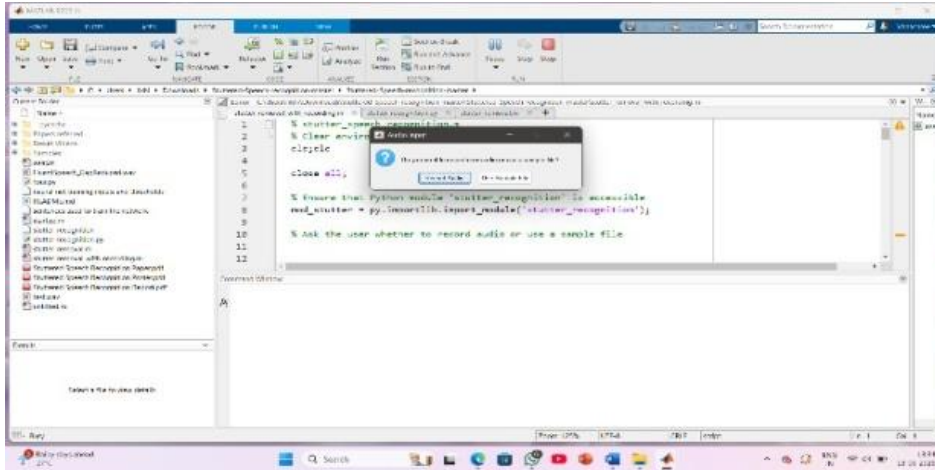
METHODOLOGY

Fig. No.1: Taking audio input.

Step-1: Audio Input

The MATLAB script stutter removal with recording. m is executed. This script:

• Clears the environment using clc; close all;.

• Loads the Python module stutter recognition using py. importlib. Import module.

• Prompts the user with a dialog box asking:

"Do you want to record new audio or use a sample file?" The user has two options:

• Record Audio – to capture live input using a microphone.

• Use Sample File – to select a pre-recorded audio file in .wav format.

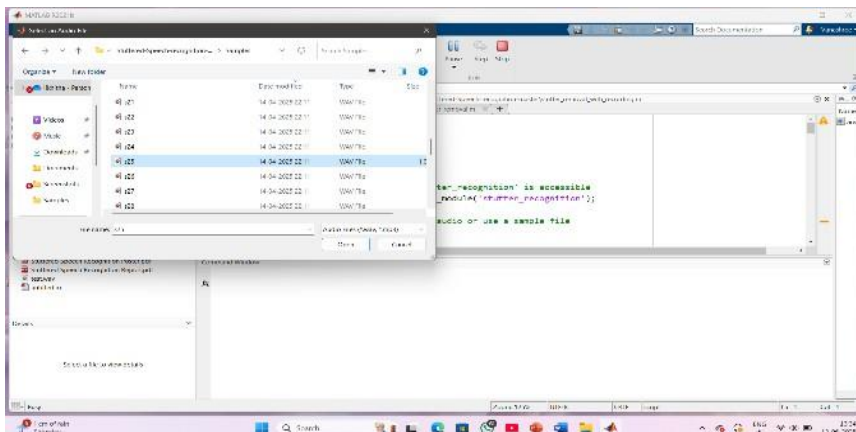This makes the interface interactive and user-friendly.

Fig. No.2: Taking Samples.

Step-2: Selecting a Sample Audio File

Once "Use Sample File" is clicked, a file browser opens inside the Samples directory.

• The user selects the file s25.wav, which contains speech with stuttering.

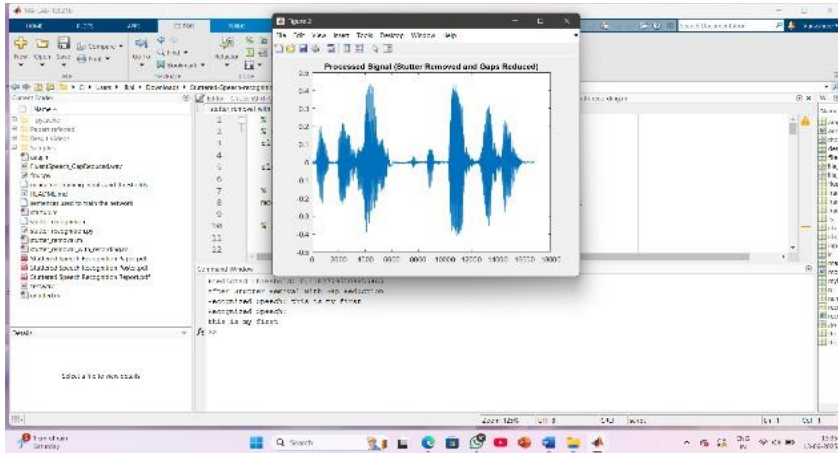- These sample .wav files are used for testing the stutter removal and speech recognition functionality.

Fig. No.3: Output of the selected sample.

Step-3: Processed Output and Results

After selecting the audio file, the system processes the input to:

Detect and remove stuttered parts.

Reduce unnecessary gaps in the signal.

- Display the processed waveform with the title: "processed Signal (Stutter Removed and Gaps Reduced)"Font Sizes for Papers
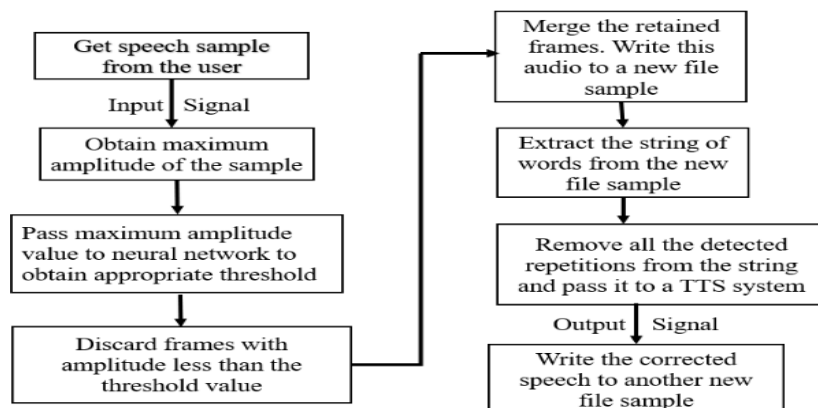

Fig. No.4: Flow of TTS(Text-to-Speech) correction sequence.

This system is designed in helping stuttered speech sound more fluent and natural by combining basic signal processing with simple machine learning and language correction techniques.

The entire process is carried out in the following steps:

1. Getting the Sample Speech

We start by having the user record or upload a sample of their speech. The process for enhancing fluency begins with this audio file.

2. Identifying the Loudest Point

After that, the system analyzes the audio to determine its maximum amplitude, which is essentially the recording's loudest point. This helps us gauge the general vigor of the user's speech.

3. Setting up a Threshold with a Neural Network

A tiny neural network model receives the maximum amplitude and determines an appropriate amplitude threshold. This threshold aids the system in distinguishing between the speech's strong and significant passages and its weak and potentially disfluent passages.

4. Eliminating Weak Sections

Any audio segments that are below this cutoff are eliminated. These are typically low-energy passages that may contain filler sounds, needless pauses, or stutters.

5. Restoring the Audio After Cleaning

To produce a smoother version of the original recording, the remaining "clean" audio frames—those with sufficient energy— are stitched back together. This new file serves as the foundation for additional processing.

6. Transforming Speech into Text

The system then transforms the cleaned audio into written text using speech-to-text technology. The remaining disfluencies can be more easily identified with this step.

7. Correcting Repetitions and Producing Original Speech

The text is examined for stuttering-related repetitions of words or syllables. To increase fluency, these are eliminated. A text-to-speech (TTS) engine receives the revised text and produces a speech that is clear and sounds natural.

8. Saving the Finalized outcome

Finally, an output audio file containing the fluid speech produced by the TTS system is saved. The user's initial message is reflected in this file in a more polished and assured manner.

RESULTS & EVALUATION

The Advanced Fluency Augmentation Framework was tested using both recorded and sample audio files containing stuttered speech. Evaluating the system's capacity to identify disfluencies like prolongations and repetitions and improve speech fluency without compromising voice naturalness was the main goal.

To detect and correct the performance of the speech sample a visual waveform is generated to compare the original speech and processed speech :
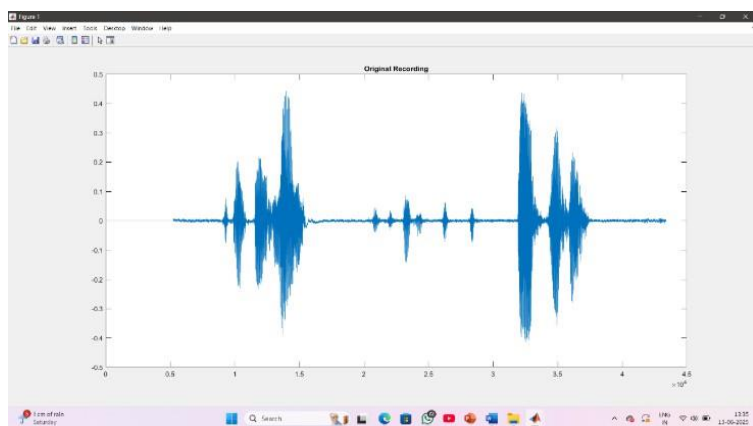

Fig. No. 5: Waveform of Original Speech.

As we can see there are visible pauses, repetitions, and extended segments in the original waveform.
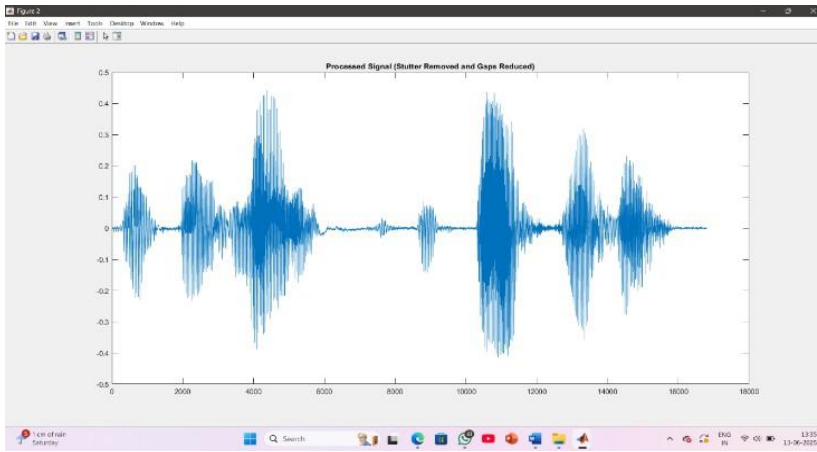
Fig. No. 6: Waveform of Processed Speech (Stutter Removed & Gap reduced)

Here we can see there is a smoother & more continuous signal is shown in the processed waveform, suggesting that the speech fluency has improved.

The accepted speech output was taken with the help of Google's Speech Recognition API to run the final processed audio through a speech recognition module. When compared to unprocessed stuttered audio, the output text from the processed speech was more accurate and demonstrated a discernible decrease in transcription errors.

For Example:

Input Speech (Stuttered) - "Thi-thi-this is m-m-my first t-t-time" Processed Output - "This is my first time"

CONCLUSION

An important step toward closing the gap between accessibility for people with stuttering and speech recognition technology is the Advanced Fluency Augmentation Framework. Clearer, more fluid speech output is made possible by the system's intelligent detection and correction of disfluencies like prolongations, repetitions, and unnatural pauses. For users who struggle with fluency, it improves the communication experience and boosts the performance of speech-based systems.

This framework shows great promise for inclusive voice-driven applications, educational tools, assistive technology, and speech therapy support. The system's sensitivity to background noise and reliance on high-quality training data are just two of its drawbacks, despite its many benefits, which include real-time correction, enhanced recognition accuracy, and simplicity of integration.

All things considered, this paper shows how useful it is to combine speech processing methods with clever algorithms to produce communication technologies that are more encouraging and inclusive. The framework can be extended for multilingual support and deeper integration into practical applications with additional development and optimization, thereby contributing to universal access to digital communication.

The future scope of this paper although the current framework provides a solid basis for speech enhancement and stutter correction, there is a great deal of room for future development and expansion:

1 Support for multiple languages and accents:

The system can be used in a wider range of linguistic contexts in the future since it can be trained to support multiple languages and regional accents.

2. Integration of Deep Learning:

The accuracy of disfluency detection and fluency restoration may be increased by incorporating sophisticated

deep learning models like RNNs, CNNs, or transformers.

3. Implementation of Mobile and Embedded:

Real-time fluency augmentation will be possible in more portable and accessible forms if the system is optimized for smartphones, tablets, and low-power devices.

4. Analysis of Prosody and Emotion:

To make the corrected speech sound more expressive and natural, future systems might examine pitch, stress, and emotional tone.

5. Customized Learning:

By using adaptive learning strategies, the system will be able to gradually learn and adapt to the speech patterns of a particular user, increasing the accuracy of corrections.

6. Real-time and cloud-based communication tools:

Real-time stutter correction during talks or meetings may be possible through integration with cloud services and real-time video/audio communication platforms (such as Zoom or Google Meet).

7. Clinical Research and Data Gathering:

In order to better understand stuttering patterns and treatment outcomes, the system may be able to gather anonymized speech data from users with their consent.

8. Support for Gamified Speech Therapy:

Younger users or patients may be encouraged to practice fluency more frequently by incorporating this framework into gamified platforms or interactive therapy tools.

9. Integration with Commonplace APIs for Speech Recognition:

To enhance the recognition of stuttered speech, the framework might serve as a pre-processing module for for-profit APIs such as Microsoft Azure, IBM Watson, or Google Speech.

10. Enhanced Robustness to Noise:

Advanced filtering and noise reduction methods may be added in the future to improve the system's dependability in practical settings.

REFERENCES

1. T. Kourkounakis, A. Hajavi, and A. Etemad, "FluentNet: End-to-end detection of stuttered speech disfluencies with deep learning," IEEE/ACM Trans. Audio Speech Lang. Process., vol. 29, pp. 2986–2999, 2021.
2. The paper titled "A Novel Approach for Stutter Speech Recognition and Correction", published in IJRASET (International Journal for Research in Applied Science & Engineering Technology) in July 2020.
3. Liam Barrett , Graduate Student Member, IEEE, Junchao Hu , and Peter Howell "Systematic Review of Machine Learning Approaches" for Detecting Developmental Stuttering IEEE/ACM Trans. Audio Speech Lang. Process vol. 30, 2022.
4. T. Kourkounakis, A. Hajavi, and A. Etemad, "Detecting multiple speech disfluencies using a deep residual network with bidirectional long short term memory," in Proc. IEEE Int. Conf. Acoust. Speech Signal Process, 2020.
5. J. Santoso, T. Yamada, and S. Makino, "Classification of causes of speech recognition errors using

attention-based bidirectional long short-term memory and modulation spectrum," in Proc. Asia-Pacific Signal Inf.Process. Assoc. Annu. Summit Conf., 2019.

6. Arya A Surya, Surekha Mariam Varghese, "Automatic Speech Recognition System for Stuttering Disabled Persons", International Journal of Control Theory and Applications, Volume 9 Number 43, 2016.

7. Gunjan Jhawar, Prajacta Nagraj, P. Mahalakshmi, "Speech Disorder Recognition using MFCC", International Conference on Communication and Signal Processing April 6-8, 2016, India.

8. Ankit Dash, Nikhil Subramani, Tejas Manjunath, Vishruti Yaragarala, Shikha Tripathi, "Speech Recognition and Correction of a Stuttered Speech", International Conference on Advances in Computing, Communications and Informatics (ICACCI), 19-22 Sept., 2018.

9. Y. Wang et al., "Transformer-based acoustic modeling for hybrid speech recognition," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 2020, pp. 6874–6878.

10. S. Alharbi, M. Hasan, A. Simons, S. Brumfitt, and P. Green, "A lightly supervised approach to detect stuttering in children's speech," in Proc. Interspeech, 2018.

11. W. Han et al., "Contextnet: Improving convolutional neural networks for automatic speech recognition with global context," 2020, arXiv: