SPARC: An AI-based Speech Processing and Real-time Correction System

TingRay Chung[1] , **Pin-Yu Chen**[2]
[1]Horace Greeley High School
[2]IBM Research
tingray.chung@gmail.com, pin-yu.chen@ibm.com

Abstract

In the world of audio narration and video production, maintaining clear and accurate dialogue is crucial. However, most work done in dubbing mistakes is done in post-production which is often not applicable to live broadcasts. This project aims to develop a real-time voice correction system that automatically detects and corrects speech errors in near real-time while integrating the adjusted audio into ongoing conversations without disrupting the natural flow. This paper utilizes various AI tools like the Nous Hermes 2-Mistral 7B DPO large language model to first generate the reference script for Coqui's XTTS-V2 zero-shot text-to-speech voice cloning model. After the correction is generated, it goes through a series of filters to replace the mistake and seamlessly integrates it. The experiment's user survey demonstrates that the corrected audio is of high quality.

1 Introduction

Past work has focused on correcting speech errors in post-production and editing. Researchers started to address this problem by aligning the waveform with the speech and putting it into a usable interface in a convenient way ([Jin *et al.*, 2017]; [Rubin *et al.*, 2013]; [Whittaker and Amento, 2004]). Modern video editing systems use audio-aligned transcript editors to streamline search and editing tasks ([Berthouzoz *et al.*, 2012]; [Meyer *et al.*, 2024]). However, there are many times when there are only a couple of misspoken words in a script and it is too difficult to re-record the entire dialogue. Moreover, recording the overdub can be extremely hard to do without the same environment and surroundings as well as the same or similar microphone, etc. Even modern stronger tools that use complicated equalization algorithms are done in post-processing and still require a user to manually record the corrected speech ([Venkataramani *et al.*, 2017]). One possible solution is to use voice cloning to mimic the user's voice to create a replacement.

While previous research focused on correcting speech errors during post-production editing, advancements in voice cloning and equalization techniques offer the potential for near real-time solutions that generate natural-sounding speech corrections. This paper presents a novel real-time voice correction system that can automatically identify when a user misspeaks, generate the corrected audio, and seamlessly integrate it into the ongoing dialogue without interrupting the flow of speech. A key AI-driven innovation is the use of the Nous Hermes 2-Mistral 7B DPO language model, which generates phonemically diverse scripts optimized for zero-shot training of Coqui AI's XTTS-v2 voice cloning model. This ensures high-quality voice replication with minimal training data. Additionally, the system employs OpenAI's Whisper model for robust transcription of spoken audio. By combining advancements in voice cloning with audio processing and equalization techniques, the study aims to create a solution that is both efficient and natural-sounding to correct speech errors within a five-second interval.

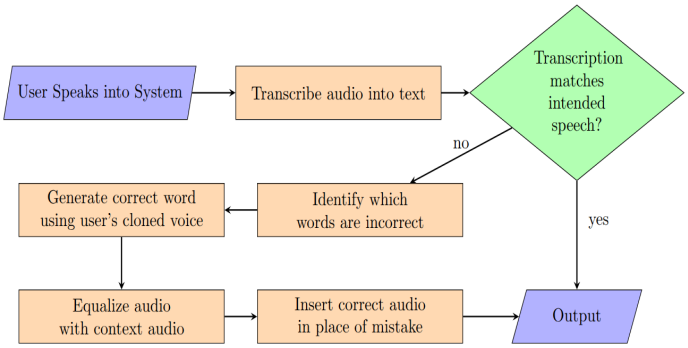

Figure 1: Overview of SPARC system that repeats as the user speaks.

2 Methodology

By utilizing a large language model (Nous Hermes 2-Mistral 7B DPO) to generate optimized scripts, the system trains Coqui AI's XTTS-v2 voice cloning model to replicate the speaker's voice with high fidelity using minimal data. Afterwards, going off of Figure 1, OpenAI's Whisper model is employed for robust transcription of spoken words, even in noisy conditions, for accurate detection of discrepancies between the spoken audio and reference text. Once errors are identified, corrected audio is dynamically adjusted in terms of volume and timbre to match the original context, with spec-

tral shaping ensuring consistency in the audio.

A speaker will use our system simultaneously while speaking. Lets say they make a mistake while talking, the model will generate the correction using text-to-speech model. However, to ensure that the integration is smooth, the system passes the corrected speech through a series of filters.

Reference Audio Script To get the best samples of audio to clone the user's voice, a system was designed to produce highly optimized voice cloning scripts in a short amount of time. At first we had ChatGPT 4o to generate the reference script. However, our internal study found that it was more effective to utilize a large language model prompted with several key words to produce the script with the best tonal diversity. It uses a greedy approach to generate sentences that maximize phoneme diversity while maintaining a syllable limit for brevity and usability. Starting with the CMU Pronouncing Dictionary, the algorithm identifies phonemes for each word and assigns scores based on phoneme diversity, word frequency (calculated using Zipf's law), and syllable count. It penalizes overly long or short words to ensure the final script is concise and easy to read.

The system rapidly selects high-scoring words and updates the set of covered phonemes, continuously optimizing for diversity while maintaining script brevity. By combining this method with a large language model, the Nous Hermes 2-Mistral 7B DPO, to generate easy-to-read sentences, the system can create effective scripts in just a few seconds. This ensures that the voice cloning system achieves high-quality results with minimal data and training time. Lastly, the reader reads this script before the system is run to ensure the model has a representation of the user's voice.

Audio Transcription To make the audio clearer, the study applied Spleeter, a tool meant to separate music into the instrumental and singing sections [Hennequin *et al.*, 2020]. However, the study utilized Spleeter in a unique way, to separate background audio from dialogue. Spleeter would only take the speech out of the audio to ensure that the background audio would not interfere with the audio processing techniques and only the pure speaking sounds were analyzed.

The system utilized Open AI's Whisper model to transcribe the audio into English text ([Radford *et al.*, 2022]; [Louradour, 2023]; [Giorgino, 2009]). The first part of Whisper is to do voice activity detection (VAD) which is to isolate the time frames in which there is active dialogue occurring. It then employs a transformer architecture to convert spoken language into text. It is trained off of a large dataset that encompasses diverse audio samples in a variety of environments. Whisper works well in challenging conditions such as noisy environments or non-standard accents making it ideal for audio transcription. The runtime of this function takes up to 1.5 seconds. The study created a function that takes in an audio file and returns a list with each word and their respective timestamps as to when each word was spoken

Identifying Mistakes The audio transcription is then compared against the desired dialogue to be spoken. The system scans through both texts and isolates which words in the transcription do not match the desired dialogue and creates a list containing the indices in the audio array indicating the start and stop points of where the incorrect words were spoken as well as the correct words that should go in each spot.

Voice Cloning To replicate the user's voice, there is a reference audio that is fed into the voice cloning model to replicate the voice. For this experiment, the system used Coqui AI's XTTS-v2, a zero-shot model to replicate the user's voice with a short reference audio clip. However, the generated speech does not perfectly fit in with the context audio of where the mistake occured. We solve this by passing it through a series of filters to ensure it replicates the current environment the speaker is currently talking in. The major audio characteristics to alter are volume and timbre as they are the most unique to how the speaker is talking at the current moment.

Volume To first normalize the volume, the system computes the minimum and maximum values for both arrays. Following this, the original audio is normalized into a range between 0 and 1. This is done by adjusting the values by subtracting the minimum value of the original audio and dividing by the range of the array. This gives the audio a uniform transformation no matter the original scale. Afterward, the values are scaled to match the replacement audio by multiplying the normalized values by the range of the replacement audio and then adding the minimum value of the replacement audio array.

Timbre However, the cloned word generated may not have the same spectral properties (e.g., frequencies, timbre) as the original word, causing noticeable differences. This is why the study implemented spectral shaping, which is designed to take the newly generated word and make it sound more consistent with the original word. The overall spectral structure (such as the frequency balance) is modified, but the distinctive features of the cloned voice still exist. The function starts by applying a Short-Time Fourier Transform (STFT) to both the original word and the new cloned word. STFT breaks the signals into time-frequency representations. The frequency bins (which represent different frequency components) and the time bins (which represent when each frequency component occurs) are created for both words. This transforms both audio signals into their frequency-domain representations, which allows for manipulation of their magnitudes and spectral features. The function modifies the magnitudes of the new word's STFT to match those of the original word's STFT. Specifically, the magnitudes are scaled by the ratio of the old word's magnitude to the new word's magnitude. It works based off a cosine wave where the equalization gets weaker at the beginning and end of the audio and stronger in the middle. After the spectral shaping is applied, the function performs an Inverse Short-Time Fourier Transform (ISTFT) on the modified STFT of the new word to get the waveform

Integration Lastly, the replacement audio is crossfaded to have a smooth transition between the two audio segments. The function applies a linear fade and decreases in amplitude of the original audio while at the same segments increasing the amplitude of the replacement audio at the beginning of the clips. At the end of the clips, the replacement audio is faded back out and the original audio is faded back in to prevent abrupt auditory changes. The replacement audio replaces the

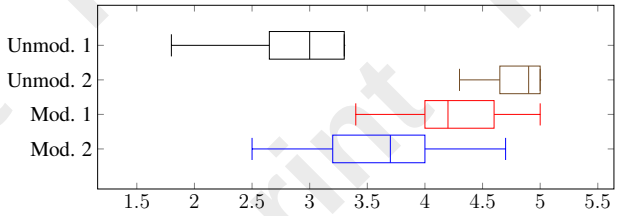

Figure 2: Boxplots of Audio Opinion Scores. Modified audio clips have comparable ratings to the unmodified audio clips

	Modified 1	Modified 2	Unmodified 1	Unmodified 2
Mean	4.1	3.645	3.036	4.755
Std Dev	0.833	0.650	1.016	0.314

Table 1: Mean and standard deviation of the audio clip ratings.

incorrect audio and the background noise that was previously removed by Spleeter is overlayed.

Runtime This study's overall speech correction workflow is streamlined, utilizing a single TTS model enhanced with algorithms, making it compatible with most hardware and capable of near real-time operation. In contrast, other rely on complex, bidirectional models combined with synchronized decoders. These alternatives demand high computational resources and cannot achieve near real-time performance. In fact, our system takes only 3 seconds per correction whereas other models take around 8-13 seconds.

Evaluation To evaluate the speech correction model, the study conducted a mean opinion score (MOS) survey. The study created 4 audio clips of which two were modified by the system and the other two placebo unmodified clips. The modified clips were created by taking a five-second segment of audio and replacing one word in the audio with another word of similar length. The survey asked participants to rate each audio clip on a scale of 1 to 5 based on the quality and clarity of the audio. Afterward, participants were asked to predict which of the audio clips they believed were modified. To analyze the results, due to a sample size of twenty participants, a Mann-Whitney U Test was done to determine if there was a significant rating difference between the modified and unmodified audio clip ratings. The Mann-Whitney U test was used because it is a non-parametric statistical test for comparing two independent groups, like the ratings for generated and natural audio. By applying a two-tailed test, the analysis checked whether participants rated one type of audio significantly higher or lower than the other.

3 Results and Discussion

The study analyzed the survey of the speech correction system and compared the ratings between the generated and natural audio. First, the study compared the participant's predictions of which clips were modified against the clips that were modified.

The boxplots in Figure 2 show the MOS for each audio type, illustrating differences in perceived quality. The natural audio (Unmodified 1 and Unmodified 2) received the highest median scores, with Unmodified 2 achieving a nearly perfect median score of 4.9 and tightly clustered quartiles. In contrast, the generated audio (Modified 1 and Modified 2) showed more variability. Modified 2 achieved a median score of 4.2, near the natural audio ratings, and its upper quartile extended to 4.6, reflecting that many participants found it satisfactory. However, the spread of scores was wider compared to Unmodified 2, with the lower whisker reaching 3.4. Modified 1, while still rated favorably with a median of 3.7, had the broadest range of scores, with a lower whisker of 2.5. The boxplot results indicate the effectiveness of the speech correction system in generating high-quality audio, particularly with Modified 2, which closely matched the ratings of natural audio. The proximity of Modified 2's median to the nearly perfect Unmodified 1 demonstrates the system's potential to produce high-quality outputs. The means in Table 1 further support this result with relatively clustered means.

Statistic	Value
Significance Level (α)	0.05
Test Type	Two-tailed
U-value	210
Z-Score	-0.73939
P-value	0.4593

Table 2: Mann-Whitney U Test Results Between Generated and Natural Audio. There are no significant rating differences between the Generated and Natural Audio

The results of the Mann-Whitney U Test in Table 2 indicate no significant difference between the MOS for the generated and natural audios with $p = 0.4593$. The Z score of -0.73939 shows that there were little to no differences. This means that participants did not consistently rate the unmodified audio than the modified audio which further highlights the success of this model.

4 Limitations

While our pipeline shows the feasibility of combining multiple systems like Whisper and XTTS-v2 for speech translation and synthesis, there are still limitations. The system's reliance on zero-shot voice cloning means that it depends on having a strong enough reference audio clip to accurately replicate the user's voice. One possible solution is to take a larger sample of the user's voice and fine-tune Whisper and XTTS-v2 on speaker-specific data.

5 Conclusion

This paper demonstrates the development and evaluation of the AI-based SPARC system using voice cloning, transcription, and audio equalization techniques. The system's transcribing, detecting, and correcting spoken errors achieved seamless audio integration, as seen by the indistinguishability of generated corrections from natural audio during testing. Results show that participants were mostly unable to discern AI-corrected audio from natural samples, suggesting the system's potential for live usage in areas like television, podcasts, online presentations, and many more. Future work should explore reducing variability in correction quality and expanding to other languages.

Ethical Statement

This study was approved by the Horace Greeley Local IRB Committee on June 1, 2024, due to the involvement of human participants.

Acknowledgments

TingRay would like to first thank his parents for their unwavering support and encouragement throughout his journey. Next, he would also like to thank his mentor, Dr. Pin-Yu Chen, a Principal Research Scientist at the IBM Thomas J. Watson Research Center, for his invaluable guidance. Dr. Chen's support in allowing him to explore his curiosity and providing helpful resources has been essential to the success of this project. This research was done in conjunction with Horace Greeley High School's research program for students and TingRay would also like to thank his research teacher Jerry Zupan for his help.

References

[Berthouzoz *et al.*, 2012] Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. Tools for placing cuts and transitions in interview video. *ACM Trans. Graph.*, 31(4), July 2012.

[Giorgino, 2009] Toni Giorgino. Computing and visualizing dynamic time warping alignments in r: The dtw package. *Journal of Statistical Software*, 31(7), 2009.

[Hennequin *et al.*, 2020] Romain Hennequin, Anis Khlif, Felix Voituret, and Manuel Moussallam. Spleeter: a fast and efficient music source separation tool with pre-trained models. *Journal of Open Source Software*, 5(50):2154, 2020.

[Jin *et al.*, 2017] Zeyu Jin, Gautham J. Mysore, Stephen Diverdi, Jingwan Lu, and Adam Finkelstein. Voco: text-based insertion and replacement in audio narration. *ACM Trans. Graph.*, 36(4), July 2017.

[Louradour, 2023] Jérôme Louradour. whisper-timestamped. https://github.com/linto-ai/whisper-timestamped, 2023.

[Meyer *et al.*, 2024] David Meyer, Eitan Abecassis, Clara Fernandez, and Christopher Schroers. Rast: A reference-audio synchronization tool for dubbed content. pages 67–71, 09 2024.

[Radford *et al.*, 2022] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision, 2022.

[Rubin *et al.*, 2013] Steve Rubin, Floraine Berthouzoz, Gautham J. Mysore, Wilmot Li, and Maneesh Agrawala. Content-based tools for editing audio stories. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, page 113–122, New York, NY, USA, 2013. Association for Computing Machinery.

[Venkataramani *et al.*, 2017] Shrikant Venkataramani, Paris Smaragdis, and Gautham Mysore. Autodub: Automatic redubbing for voiceover editing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, page 533–538, New York, NY, USA, 2017. Association for Computing Machinery.

[Whittaker and Amento, 2004] Steve Whittaker and Brian Amento. Semantic speech editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, page 527–534, New York, NY, USA, 2004. Association for Computing Machinery.