Benchmarking Chinese Stuttered Speech Recordings Against Speech Recognition Models

Wai Yuen Cheng
Computer Science & Biomedical Engineering
Boston University
College of Engineering & College of Arts & Sciences
wycheng@bu.edu

Mary Choe
Computer Science
Boston University
College of Arts & Sciences
marychoe@bu.edu

Lingjie Yuan
Computer Science
Boston University
Metropolitan College
yuanlj@bu.edu

Yol-Emma Veillard
Computer Science
Boston University
College of Arts & Sciences
chimmay@bu.edu

***Abstract*—In recent years, automatic speech recognition (ASR) models have achieved significant accuracy, particularly utilizing transformer, encoder & decoder architectures. Those models signify pioneering success researchers have reached. However, due to various limitations in datasets that have been used to train those state-of-the-art (SOTA) models, biases have to be studied closely in order to address unfair prediction that fails to acknowledge potential imbalances underrepresented marginalized communities in disadvantages when using ASR models. Furthermore, as many ASR models target the ambition of multilingual transcription capability, the models' ability to establish unbiased, fair performances on different languages also present great needs for further studies. This paper evaluates 5 SOTA models on their performance with Chinese Stuttered Speech, benchmarking the underlying concerns on their corresponding performance against speech disabilities and Chinese linguistics.**

***Index Terms*—Stuttered Speech, Chinese Automatic Speech Recognition, Whisper Large, Whisper Tiny, Wav2Vec, WeNet, Google Cloud, Azure**

I. Introduction

Our goal with this project was to identify and quantify the fluency biases of stuttered Mandarin speech in ASR models and visualize how the results change with different types of stuttering and across each model. We have built on the work a group of Boston University students had completed during the summer. They created a pipeline with multiple popular speech recognition models, did some data preprocessing, and drew preliminary conclusions with many different visualizations. Our two main objectives are as follows.

1) Analyze and provide a deeper understanding of previous results
2) Work with the data to understand and further quantify fluency biases in existing ASR models

We worked with Shaomei Wu, founder and CEO of AImpower, to contribute research to this underrepresented area.

II. Background

A. Speech Disabilities — Stuttering Phenomenon

About 1 percent of the world has stuttered speech [1]. Stuttering is a speech impediment that affects people socially and mentally. It makes basic communication difficult and can be accompanied by body movements and facial grimaces [2]. Stuttering often carries significant social consequences, such as negative reactions from listeners, bullying, teasing, social exclusion, and rejection. People who stutter (PWS) may also face harmful stereotypes, including perceptions of being less intelligent, less capable, less attractive, less socially skilled, and more anxious compared to fluent speakers. As a consequence, PWS frequently experience intense emotional and cognitive responses to their stuttering, such as fear, guilt, shame, helplessness, social anxiety, self-stigma, and a tendency to avoid specific sounds, words, situations, people, and even relationships. Despite the discrimination and difficulties PWS face, they have made incredible accomplishments such as in the case of Biden even becoming the president of the United States of America [3]. As only a small percentage of people stutter, there is a major lack of data and research on stuttering in newer applications such as ASR models. Specifically ASR models and applications such as voice assistants like Alexa and Siri have limited research and basically no accommodations for PWS.Although speech-to-text APIs can produce written transcriptions much faster than human transcribers, there are significant concerns about the potential bias and inaccuracies in automated transcription especially for stuttered speech [4].

B. Different Forms of Stuttering

There are many forms of stuttering such as:

1) Repetitions: sounds or words are repeated (e.g. "wor-wor-words" or "many many many words")
2) Prolongations: sounds are held (e.g. "wwwwwords")

3) Blocks: long pauses between sounds or words (e.g. "words")
4) Interjections: sounds or noises in between words (e.g. "many uh um words")

These disruptions can have a significant impact on the accuracy and effectiveness of speech recognition models, which are generally trained on fluent speech. The mismatch between the models' training data and real-world speech patterns can lead to poor performance when dealing with stuttered speech. The dataset we are working with includes all of these kinds of stuttering, and we studied how different ASR models performed on the different types of stuttering.

C. Multilingual Automatic Speech Recognition

Research into multilingual capabilities of ASR systems involves a combination of linguistic expertise, data collection, model architecture design, and machine learning innovations. Researchers are working to improve ASR models that can seamlessly handle multiple languages, dialects, and accents, recognizing that the diversity of human speech poses significant challenges. To develop truly multilingual systems, researchers incorporate vast and diverse multilingual datasets, often compiling them from multiple languages with varying levels of prevalence in the global speech community. Advancements in deep learning techniques such as transfer learning, multilingual pre-training, and cross-lingual embeddings are being employed to create systems that can efficiently process and transcribe speech across multiple languages simultaneously. The need for multilingual ASR systems is driven by the growing demand for inclusive, global communication technologies. With an increasingly interconnected world, users expect digital assistants, transcription services, and voice-based technologies to work across different languages.

D. AImpower

AImpower.org is committed to promoting public interest technology through research and advocacy. Its mission is to ensure that technological advancements are inclusive and fair, especially for underserved and marginalized groups. This project supports AImpower.org's objectives by addressing the challenges faced by individuals who stutter and proposing ways to reduce fluency bias in speech recognition models. AImpower.org believes that technology must serve everyone and works alongside marginalized communities to create inclusive, empowering technologies that break down barriers, generate positive impacts, and advance social justice.

III. Dataset

The dataset is from AImpower, the first stuttered speech dataset in Mandarin Chinese. It collects 72 stuttering speakers' speech samples. All the speeches are divided into segmentation, and there are 37250 segmentation in total. We received all these 37250. We got all the annotations and the transcriptions for 6 different models from the previous team. We preprocessed both the annotations and transcriptions and then viewed the distribution of the valid dataset.

A. Preprocess

We want to analyze the performance of each ASR model, which means to compare the annotations (ground truth) with the transcriptions from each model. We contacted ground truth and all the transcriptions at first and removed segmentation that don't have transcriptions (annotations equals non). Then we removed all the punctuations both in the annotations and transcriptions. Punctuations might negatively affect the model performance since models cannot generate punctuations correctly. After that, we constructed another ground truth set, which keeps all the repetition words from stutter, in order to analyze model performance more accurately. So far, the dataset contains 37202 pieces of data.

ground_truth	ground_truth_cleaned	ground_truth_cleaned_with_rep	transcription
资深/p的口/b[口/r/b]吃患者。	资深的口吃患者	资深的口口吃患者	资深的口口吃患者

Fig. 1. Example for Cleaned Ground Truth

We also found that some model's transcriptions contain traditional Chinese characters and Arabic numerals (123), while all of these in annotations are simplified Chinse characters. We used "opencc.OpenCC('t2s') " and "cn2an.transform(x, "an2cn")" to finish this transformation.

ground_truth	ground_truth_cleaned	transcription_origin	transcription
你好，米雅，六百五十韩元是多少法郎。	你好米雅六百五十韩元是多少法郎	你好米娅, 650韩元是多少法郎	你好米娅六百五十韩元是多少法郎

Fig. 2. Example for Transferred Dataset

Besides, we calculated some necessary attributes for further analysis. We used the segmentation's start and end time to get the segmentation duration. We counted the stutter times in each segmentation. We also counted each segmentation's characters. Then we calculated the speech speed for each segmentation. After that, we used these attributes to do the variable normalization. We got the stutter frequency via stutter count divided by character count or duration.

variables	meaning
duration	end_time - start_time
stutter_count	stutter times in each segmentation
character_count	characters in each segmentation
WPS (speech speed)	characters per second
stutter_frequency_by_character	stutter times per character
stutter_frequency_by_duration	stutter times per second

TABLE I
NECESSARY VARIABLES AND MEANING

B. Distribution

To get the data overview, we constructed the distribution for the different attributes of the data set.

First, without considering whether the data is stuttered or not, we will focus on understanding the overall distribution of the data through three aspects: duration, character count, and words per second (WPS).

The duration of the dataset is primarily distributed between 1 to 15 seconds, with long segments being relatively less frequent compared to short ones. This is influenced by the inherent characteristics of the dataset. Human speech often includes a significant number of short responses, such as ”okay” or ”yes.” Additionally, the presence of command-type data in the dataset further contributes to the higher frequency of short utterances.

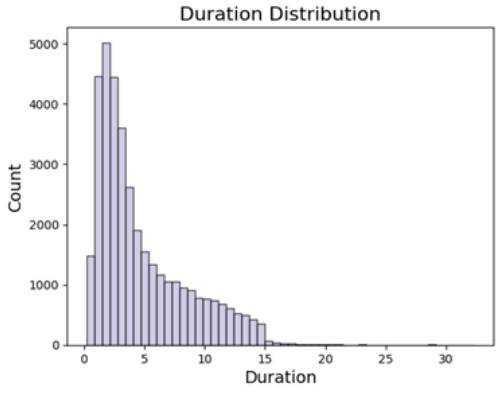


Fig. 3. Duration Distribution

The character count distribution closely aligns with that of the duration. We observe that most sentences are less than 20 characters long. Additionally, as the length of a sentence increases, the corresponding data amount in the dataset gradually decreases.

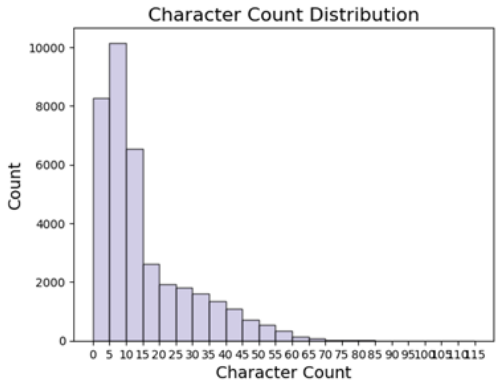


Fig. 4. Character Count Distribution

We also attempted to normalize sentence lengths by constructing the WPS attribute to determine whether speaking speed will affect the model’s transcription accuracy. From the WPS distribution chart, we find that the WPS distribution tends to resemble a normal distribution, with most values concentrated between 2 to 4 words per second. And there are few outliers. Based on this observation, we preliminary conclude that speaking speed is unlikely to have a significant impact on the model’s performance.

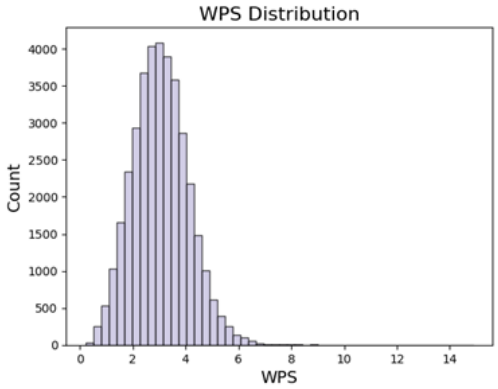
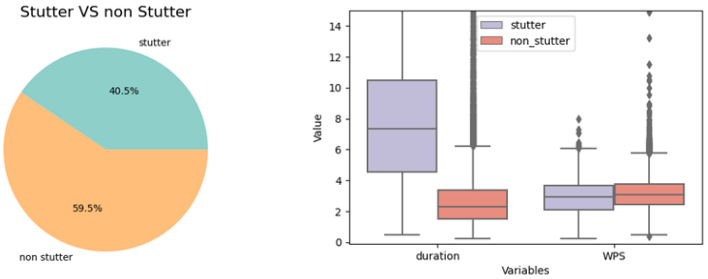


Fig. 5. WPS Distribution

We then categorized the data based on whether it contained stuttering. Approximately 60% of the segments were non-stuttering, while the remaining 40% contained stuttering, indicating that the dataset is relatively balanced. Regarding the distribution of the two categories, we observed that the average duration of stuttering segments was around 8 seconds, whereas non-stuttering segments were generally shorter, averaging only about 2 seconds. However, the speech rate (WPS) distribution was largely consistent between the two datasets.



(a)Amount Distribution (b)Attributes Comparison

Fig. 6. Stutter VS non-Stutter Dataset Comparison

For the stuttering part, we constructed a statistical distribution of the number of stuttering occurrences per sentence(Fig7). The result shows that most sentences have between 1-3 stuttering instances. However, this distribution does not account for related factors such as sentence length. Therefore, we further normalized the stuttering variable to better analyze its impact.

We employed two methods for normalization: stutter count per duration and stutter count per character count. (Fig8) It is evident that the frequency normalized by duration exhibits a more normal distribution. However, our primary focus is on the frequency normalized by character count. In this distribution, the average stuttering frequency per character is concentrated between 0 and 0.25. This indicates that the overall stuttering tendency in the dataset is not particularly severe.

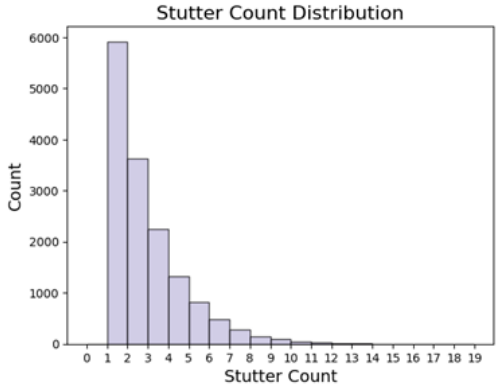
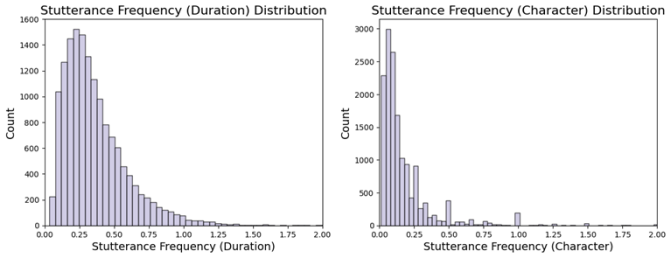


Fig. 7. Stutter Count in Each Segment



(a)Frequency with Duration (b)Frequency with Count

Fig. 8. Stutter Frequency

IV. Interpretation of Previous Results

A. Objective

Our first goal is to learn from the work done by the summer team and use their findings as a foundation to expand upon and build further insights. We selected a few key graphs that provided a foundation for the data we gathered, helping to frame our analysis and guide the direction of our project.

B. Interpretation

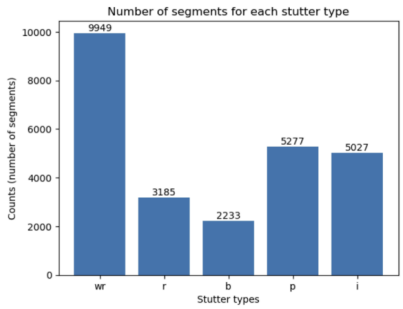


Fig. 9. Count of the Stutter Types in the Dataset

Figure 9 counts how many times each stutter type appears in the dataset we are working with. The x-axis on the graph above shows word repetition, sound repetition, blocking, prolongation of sounds, and interjections respectively. We can clearly see that word repetition is the most common and appears at least twice more often than the other stutter types.

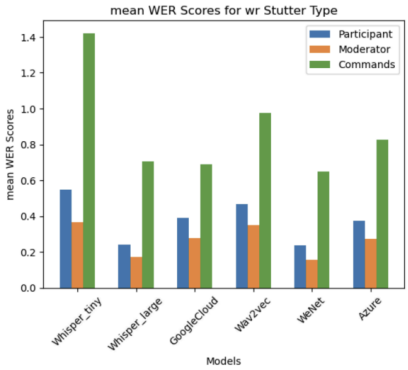


Fig. 10. Average Word Error Rate Of Different ASR Models

Figure 10 shows how well the Whisper, Google Cloud, Wav2vec, WeNet, and Azure models performed on the dataset. Whisper tiny had the highest word error rate, and WeNet had the lowest. The dataset consisted of a diverse range of audio recordings, capturing different types of speech interactions. A portion of the audio consisted of voice commands, while the remaining recordings were interviews conducted between a moderator and a participant. In these interviews, the moderator generally displayed a less pronounced stutter compared to the participant. As a result, the moderator's speech exhibited a lower word error rate than the participant's, reflecting the difference in fluency between the two speakers. This contrast in fluency provided valuable insight into how varying levels of stuttering can influence speech recognition accuracy and error rates.

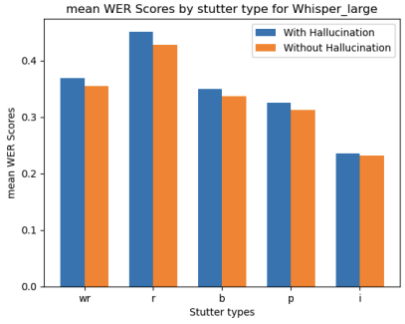


Fig. 11. Average Word Error Rate Of Different Stutter Types for Whisper large

Figures 11 and 12 show how well Whisper large and WeNet performed on the different stutter types. The x-axis on the graph above shows word repetition, sound repetition, blocking, prolongation of sounds, and interjections respectively. Both graphs clearly show that sound repetition has the highest error rates and interjections have the lowest error rates. The with hallucination and without hallucination was a part of an attempt to help remove hallucinations from the dataset during the data processing. Unfortunately it was unable to improve the performance of the ASR models as the with and without hallucination data have a very similar error rate.

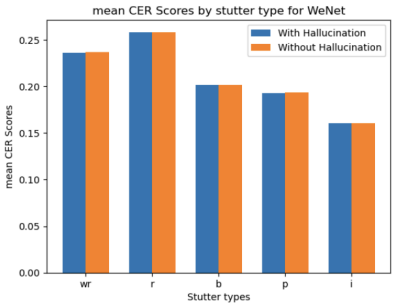


Fig. 12. Average Character Error Rate Of Different Stutter Types for WeNet

C. Summary

The summer team made significant progress that laid the groundwork for our current work, allowing us to draw valuable conclusions. Their efforts formed the foundation on which we were able to build further insights. They initiated the data preprocessing phase, carefully preparing the dataset, and developed a pipeline that enabled us to run the audio through several widely-used automatic speech recognition (ASR) models. This pipeline was crucial for getting results, and we were able to leverage all of their work to conduct our own analysis. They uncover some interesting preliminary findings, particularly regarding the types of stuttering that are more prominent and the ones that ASR models struggle to interpret accurately. By examining these patterns, we gained a clearer understanding of the challenges stuttered speech presents to speech recognition systems, shedding light on specific areas where improvements are needed.

V. Approaches

In our analysis, we implemented the word error rate and the rouge score to evaluate the overall performance of the model. We visualized the two performance metrics as functions of stuttering counts, audio lengths, and stuttering frequencies by time and character count, respectively. After that, we compared all analyzes for each model and calculated the correlations between metrics and parameters.

A. Word Error Rate (WER)

To evaluate the performance of the models based on their ability to generate matching transcripts, we implemented the calculation of Word Error Rate (WER). WER is a surface-level evaluation metrics for languages aiming at quantifying the mismatch of predicted transcripts and the ground-truth transcript of the audio transcribed. To obtain the WER, we implemented the Levenschtein Distance Algorithm, specifically addressing the following conditional formula. Given 2 strings, a ground truth transcription and model-generated transcription, we will compute:

$$lev(x) = \begin{cases} |a| & \text{if } |b| = 0, \\ |b| & \text{if } |a| = 0, \\ lev(a[1:], b[1:]) & \text{if } a[0] = b[0], \\ 1 + min \begin{cases} lev(a[1:], b) \\ lev(a, b[1:]) \\ lev(a[1:], b[1:]) \end{cases} & \text{otherwise} \end{cases} \tag{1}$$

Levenschtein Distance Algorithm is a recursive algorithm that compares two strings and has been a consensus for quantifying natural language processing linguistics results. We therefore select it for the calculation to fundamentally compare the overall matching percentage between the model generated transcript and the ground truth transcript.

B. ROUGE score

The contextual retainment capabilities of ASR models are also critical to the understanding of their tolerances and performances. For such purpose, we implemented 3 variants of ROUGE score evaluations:

- ROUGE-1
- ROUGE-2
- ROUGE-L

We chose 3 variants because we want to understand the models' capabilities to retain the unigram and bigram overlaps, as well as longest common sequence overlaps. Those 3 parameters allow us to review the full contextual awareness of the result considering ASR models are composed of an encoder to handle audio input and a decoder for the LLM purpose.

C. Statistical Analysis — Spearman Correlation

After obtaining the metrics (WERs and ROUGE scores), we proposed to observe their relationships with various settings of the audio inputs. Therefore we created a series of visualizations in the VI. Ultimately, for us to show the statistical relationships between each pairs of metric and input settings, we computed the Spearman correlation and evaluated the monotonic correlation relationship. In our interpretation, to show that an ASR model does not have direct influence from stuttering phenomenon, we should expect correlations closer to 0. In another words, if the absolute value of the correlation is higher, the model is more affected by the stuttering phenomenon.

VI. Results

We obtained results showing interesting findings. Both our WER and rouge score results did not reveal a particularly clear relationship of performance.

A. Word Error Rate (WER)

For the visualization of WER with respect to stuttering count settings, we have created 24 scatter plots **??**. Due to the amount, we will put an example below addressing key challenges and place the rest in the supplementary material.

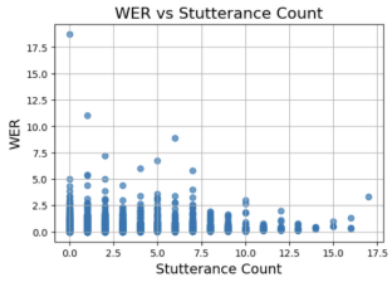


Fig. 13. Word Error Rate (WER) w.r.t. Stuttering Count for Whisper Large

As shown above, a common observation we obtained from the WER scatter plots is that they are often not portrayed as data points that can be modeled by a regression method. However, we can see that there are a high concentration of data points at lower stuttering counts. Furthermore, a few outliers can be observed from the plot as well. Those two phenomenons are commonly found with the plots we created.

After that, we carried out analysis using the averages and standard deviations of the WER, which would be included as part of the supplementary material section as well.

B. ROUGE score

As we computed the ROUGE-1, ROUGE-2 and ROUGE-L scores, we have obtained 72 plots of precision scores, recall scores, and f1 scores respectively, which are included as part of the supplementary material section.

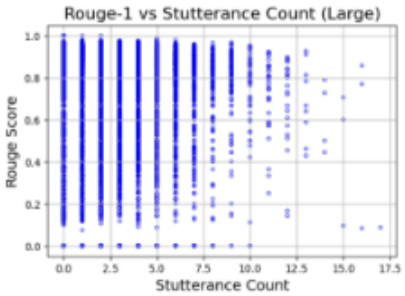


Fig. 14. F1 Scores from ROUGE-1 w.r.t. Stuttering Count for Whisper Large

While we do not see obvious relationship in the F1 scores plot, we can once again see high density of data in the lower value range of stuttering count. This is once again common with other results we have obtained, which we proposed to be due to the distribution of the dataset.

We finally calculated the averages and standard deviations to proceed with analysis. Those values are included at the Supplementary Material section.

C. Statistical Analysis

We finally computed the correlations of WERs and ROUGE scores to the four audio setting variables we used for analysis. The result is as shown below.

As depicted in Fig. 3, we are seeing different correlation patterns for each model respectively. With Azure and Whisper

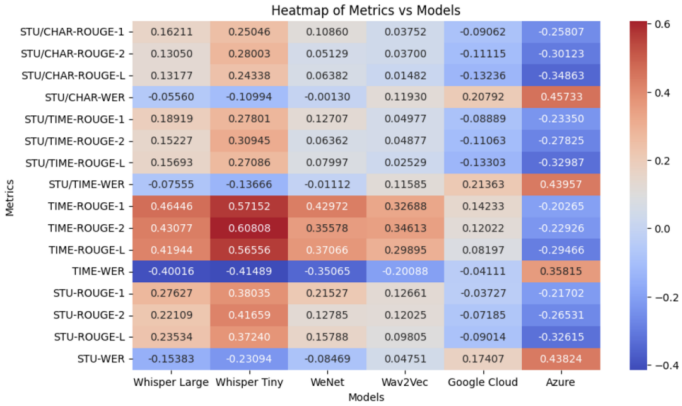


Fig. 15. Correlations of Metrics and Audio Input Variables for Different Models

Tiny showing two extremes of correlations, we can conclude that Whisper Tiny can perform well even with stuttering speeches while Azure cannot tolerate stuttering phenomenon.

VII. Discussion

From our results, we observe the different influences of stuttering phenomenon on the 6 different models. Notably overall, we see Azure performing the worst with stuttered speech while Whisper Tiny performing the best with stuttered speech. As a contrast against our initial hypothesis, which proposed that the higher amount of stuttering phenomenon or more frequent the stuttering happens, the worse the performance would become. Whisper Tiny, particularly, showed a counterintuitive result against our proposal. Out of 6 models, 3 of which showed negative correlation between the stuttering count, or the frequency of stuttering phenomenon, to the WER, which means more common stuttering phenomenons would drive a lower WER.

In terms of biases driven by stuttering phenomenon, we concluded that the influences of stuttering are mostly low against models. With the intolerance of ranking from high to low as below. (1: highest, 6: lowest)

1) Whisper Tiny
2) Whisper Large
3) Azure
4) WeNet
5) Google Cloud
6) Wav2Vec

Whisper Tiny, although showing a positive correlation for rouge scores and stuttering phenomenon and negative correlation for WER and stuttering phenomenon, can be observed to be the most influenced by the stuttering phenomenon. It depicted its variability heavily due to its distinct prediction trend portrayed by the correlation with the high magnitudes.

Azure, on the other hand, can be observed to have the highest biases at the other extreme. Its bias against the stuttering phenomenon drives a negative influence toward the model's performance overall.

To name the model with the least bias against stuttering phenomenon, we are proposing Wav2Vec. It consists of correlations with the lowest magnitudes of correlation, indicating the least direct linear relationship between the stuttering phenomenon and the the performance among all models. With this analysis result, we believe that Wav2Vec is the least biased when tackling speech input with stuttering features.

VIII. Conclusion

Our findings indicated a relationship between the stuttering phenomenons and performance of ASR models. On top of the observable relationships driven by statistical analysis, we also found that models of various architectures and engineering methods yield different levels of biases against stuttered speeches. However, our findings require further work through evaluations of data from other settings, in order to fully quantify the bias against speech disabilities. Lastly, with the finding that different architectures yield performances and biases of various levels, more effort is needed for the engineering of the novel transformer model that could account for the stuttering speech disability community.

IX. Recommendations for Future Steps

For future steps in this project, it is essential to expand the training data to include a broader range of stuttering types. This would ensure that the models are exposed to more varied instances of disfluencies, improving their ability to recognize and process stuttered speech. Additionally, developing algorithms to detect and correct disfluencies is a critical next step. These algorithms could identify the specific type of disfluency, such as repetitions, prolongations, or blocks, and apply context-based corrections by analyzing surrounding words, as well as considering the duration and frequency of the disfluencies. Collaborating with other organizations and companies focused on speech recognition and accessibility could also help enhance model training, especially for stuttered speech. By working together, it would be possible to create more accurate and inclusive models, potentially leading to the development of a dedicated model specifically trained to handle the unique characteristics of stuttered speech.

Acknowledgment

This paper is established in collaboration between Boston University Spark! Department and AImpower.org. The dataset used in this study is provided by AImpower.org. All analysis code snippets are created using Python.

References

[1] O. Bloodstein, N.B. Ratner, and S.B. Brundage. 2021. A Handbook on Stuttering, Seventh Edition. Plural Publishing, Incorporated. https://books.google.com/books?id=Abw0EAAAQBAJ

[2] J. Prasse and G. Kikano, “Stuttering: An overview,” American family physician, vol. 77, pp. 1271–6, 2008.

[3] W. Shaomei, ”“The World is Designed for Fluent People”: Benefits and Challenges of Videoconferencing Technologies for People Who Stutter,” https://www.shaomei.info/pdfs/Stuttering_VC_preprint.pdf.

[4] Kunal B. Dhir, John A. H. S. Frank, and Rajiv R. Shah. 2024. Investigating Bias in Automated Speech Recognition Systems. Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FACCT 2024), 111-123. Retrieved from https://facctconference.org/static/papers24/facct24-111.pdf.