SELF-SUPERVISED SPEECH MODELS FOR WORD-LEVEL STUTTERED SPEECH DETECTION

Covering the author identities

ABSTRACT

Clinical diagnosis of stuttering requires an assessment by a licensed speech-language pathologist. However, this process is time-consuming and requires clinicians with training and experience in stuttering and fluency disorders. Unfortunately, only a small percentage of speech-language pathologists report being comfortable working with individuals who stutter, which is inadequate to accommodate for the 80 million individuals who stutter worldwide. Developing machine learning models for detecting stuttered speech would enable universal and automated screening for stuttering, enabling speech pathologists to identify and follow up with patients who are most likely to be diagnosed with a stuttering speech disorder. Previous research in this area has predominantly focused on utterance-level detection, which is not sufficient for clinical settings where word-level annotation of stuttering is the norm. In this study, we curated a stuttered speech dataset with word-level annotations and introduced a word-level stuttering speech detection model leveraging self-supervised speech models. Our evaluation demonstrates that our model surpasses previous approaches in word-level stuttering speech detection. Additionally, we conducted an extensive ablation analysis of our method, providing insight into the most important aspects of adapting self-supervised speech models for stuttered speech detection.

Index Terms— Self-supervised Speech model, Stuttering, Speech Pathology

1. INTRODUCTION

Stuttering is a complex, multifactorial disorder characterized by atypical disruptions in the forward flow of speech [1]. It affects approximately 1 percent of the population and has a significant negative impact on all aspects of an individual's life including social, educational, emotional, and vocational [2, 3]. A stuttering assessment is conducted by a licensed speech-language pathologist. It includes measurement of the impact of stuttering on the person's quality of life as well as analysis of the individual's speech to determine stuttering severity. Unfortunately, over the past decades, speech-language pathologists have consistently reported limited competence in working with individuals who stutter (e.g., [4, 5, 6, 7] , with less than 5 percent of licensed professionals in the United States reporting expertise working with individuals who stutter [8, 6]. Another challenge is that the process of speech transcription and disfluency annotation is labor-intensive, as speech-language pathologists primarily transcribe and annotate speech disfluencies manually.

A potential approach to address the above challenges is to use deep learning models to serve as an initial screening for stuttering, which could be deployed on edge devices such as smartphones or laptops. Early work [9, 10] has used simple deep learning models to detect stuttering behavior from speech utterances. However, the main disadvantage is that the system is highly dependent on the scale of training data. Unfortunately, the largest publicly available stuttering dataset is SEP-28K [11], which only contains 15.6 hours of utterance level labeled data, which is relatively small compared to standard datasets like LibriSpeech [12], which has a total 960 hours for training.

A promising recent approach in the speech community to deal with limited labeled data is Self-Supervised Learning (SSL). Speech SSL models are first pretrained on a large amount of untranscribed speech and then finetuned on a smaller amount of transcribed/annotated speech for specific downstream tasks. Due to the task-agnostic nature of the pretraining, speech SSL models demonstrate high generalizability across different speech processing tasks [13]. Hence, speech SSL models are regarded as foundation models in many applications nowadays, such as Automatic Speech Recognition [14, 15] and Speaker Verification [16]. Due to the advantage of reducing the need for disfluency-labeled data, speech SSL models are a promising approach to tackling stuttered speech detection problems. Specifically, prior works [17, 18, 19] utilize Wav2vec 2.0 [20] as their model backbone for stutterred speech detection.

Despite progress in this field, previous work has mainly focused on utterance-level stuttering detection. However, this is too coarse for clinical application as stuttering-like disfluencies are also present in typically fluent individuals who are not diagnosed as a Person who Stutters (PWS). Clinically, stuttering is diagnosed based on meeting a certain frequency threshold of stuttering-like disfluencies. Hence, we are interested in developing a more fine-grained stuttering detection model. To our best knowledge, [21] is the only previous

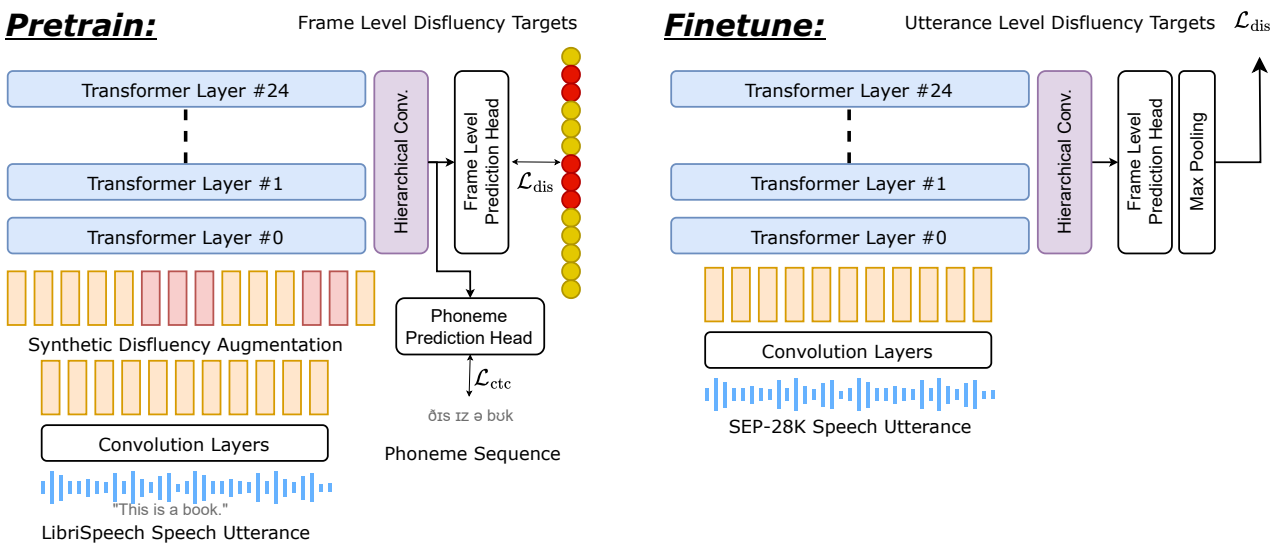

Fig. 1. Overall framework diagram. The transformer layers and Convolution Layers are initialized from off-the-shelf WavLM Large and remain frozen during pretrain and finetune stage. "Hierarchical Conv." stands for "Hierarchical Convolution".

work that investigates the task of frame-level stuttering detection. However, they evaluated their model by removing the segments that are predicted as stuttering and conducted a user study on Amazon Mechanical Turk, which was not assessed on clinical data collected and annotated by speech-language pathologists (SLPs).

In our work, we first collect a set of speech recordings with stuttering events annotated by speech-language pathologists and use it as a word-level stuttering detection evaluation benchmark. Then we propose a word-level stuttering detection model that utilizes WavLM [22] as our backbone. Following [21], we first pretrain on LibriSpeech with synthetic disfluency augmentations and then finetune our model on SEP-28K dataset. We show that our model not only shows strong performance on utterance-level stuttering detection on SEP28K but also outperforms prior work on word-level stuttered speech benchmark by a large margin. In addition, we study the effect of the utilization of Speech SSL models on word-level stuttering detection with extensive experiments. In conclusion, our contribution can be summarized as the following:

1. We are the first to focus on word-level stuttering detection, which is more closely aligned to clinical applications.
2. We propose a word-level stuttering detection model that achieves state-of-the-art performance.
3. We study the effect of utilizing Speech Self-supervised models on word-level stuttering detection with clinical data evaluation and extensive ablations.

We open source our code on Github[1]

[1] www.github.com/atosystem/SpeechSSLStutterDetect

2. RELATED WORK

In the field of stuttered speech detection, there are several proposed datasets, such as UClass [23], KSoF [24], and Sep-28K [11]. UClass is an unlabeled dataset while the others are labeled. KSoF and Sep-28K share the same format. Both of them consist of 3-second clips with utterance-level annotations. Specifically, Sep-28K also includes a subset of utterance level annotation from Fluency Bank. Not only the size of the dataset are relatively small, but they only provide utterance-level labels, which we believe is one of the main obstacles in this field.

Several recent works have used Speech SSL models in stuttering detection. In [18], they adopted Wav2vec 2.0 as their backbone and trained different models using different upstream layers of Wav2vec 2.0 for stuttering detection. They also proposed to employ gender prediction as an auxiliary task for enhancing performance. In [19], they pointed out that stuttering types do not happen independently, so they framed the detection problem as a multi-label classification problem. In [17], they improved the quality of Wav2vec 2.0 embeddings for stuttered speech detection using Siamese network and contrastive training. Despite these impressive results, their models predict at the utterance level, which is insufficient for clinical usage. Furthermore, the optimal strategies for utilizing Speech SSL models for stuttering detection are still not fully understood. Several prior works also incorporate Speech SSL models into their model design, such as for detecting dysarthria [25] and aphasia [26, 27], but they focus on other types of speech disorders and are not directly applicable to our task.

To the best of our knowledge, [21] is the only study that focused on fine-grained stuttering detection, more specifically frame-level stuttering detection. Hence, we chose it as our

baseline. It is first pretrained on LibriSpeech with synthetic disfluencies augmented during training with frame-level supervision. Then, they added a max pooling layer to finetune on utterance-level SEP-28K dataset.

3. METHOD

3.1. Model Architecture

Following prior work using Wav2vec 2.0 on stuttered speech detection, we chose another Speech SSL model, WavLM Large [22] as our backbone since it significantly outperforms Wav2vec 2.0 on most SUPERB [13] benchmark tasks. Our model structure is shown in Figure 1. Furthermore, we adopt the recent Hierarchical Convolution Interface (HConv.) as our method for utilizing the WavLM backbone according to recent work [28], which was shown to improve overtaking a learnable weighted sum of layer activations. As pointed out in [18], the information for stuttering detection exists in multiple layers, so we decided to use HConv. due to its ability to aggregate information across multiple layers. After that, we add a few non-linear layers as the prediction head to classify individual frames as stuttered or non-stuttered.

In [18], it was shown that phoneme recognition and stuttering detection tasks utilize the same set of layers in the upstream Speech SSL model. Motivated by this, we decided to add an auxiliary Connectionist Temporal Classification (CTC) Loss to predict the phoneme sequences. The CTC loss is only applied in the pretraining stage where we have the ground truth phoneme sequences.

We roughly follow the training pipeline in [21], our framework consists of two stages: Pretrain on synthetic augmentation of LibriSpeech 360 and Finetune on SEP-28K.

Pretrain on LibriSpeech: Different from previous works, we add our synthetic augmentations between the convolution layers and the transformer layers of WavLM rather than adding augmentation directly on the input waveform. We keep the augmentation types the same as [21], which includes artificial prolongation, and word/sound repetition. By doing so, we significantly save on computation and speed up the online augmentation. Suppose the outputs of the prediction head for each input utterance are $\mathbf{o} = [\mathbf{o}_1, \mathbf{o}_2, \dots \mathbf{o}_T]$, where $\mathbf{o}_i = [o_i^p, o_i^n]$, corresponding to the probability of positive and negative for frame i respectively. The labels are $\mathbf{l} = [\mathbf{l}_1, \mathbf{l}_2, \dots \mathbf{l}_T]$. The loss is calculated as a cross entropy between the prediction head outputs and the label sequence.

$$\mathcal{L}_{\text{dis}} = \frac{1}{T}\sum_{i=1}^{T} -o_i^p \log l_i^p - o_i^n \log l_i^n, \tag{1}$$

$\mathcal{L}_{\text{ctc}}$ is calculated between the frame level output of Phoneme Prediction head and the ground truth phoneme sequence in LibriSpeech (which is obtained from a force aligner). The overall pretrain loss is:

$$\mathcal{L}_{\text{pretrain}} = \mathcal{L}_{\text{dis}} + w_{\text{ctc}} \cdot \mathcal{L}_{\text{ctc}}, \tag{2}$$

where w_{ctc} is a scalar hyperparameter.

Finetune on SEP-28K: For the fine-tuning stage, since we do not have frame-level labels for SEP-28K, we simply take the timestep with the largest positive class prediction to calculate cross entropy with the utterance level target in SEP-28K, which is same as [21]. Formally,

$$t' = \text{argmax}_{1 \le i \le T}\, o_i^p, \tag{3}$$

$$\mathcal{L}_{\text{dis}} = -o_{t'}^p \log l_{t'}^p - o_{t'}^n \log l_{t'}^n. \tag{4}$$

3.2. Implementation Details

Model and training details: The scalar hyperparameter w_{ctc} is set to 0.3 empirically to balance the magnitude between the two losses. For the finetuning stage, we feed the entire audio without augmentation and calculate a global level loss with its corresponding label. In both stages, the HConv. interface and the stuttering detection head are trainable. We use Adam optimizer with a learning rate of $1e-4$. We select the checkpoint with the lowest validation loss for the pretraining stage and for the finetuneing stage, we select the one with the highest F1 score on the validation set.

Dataset: For pretraining, we use LibriSpeech 360hr and for SEP-28K, we follow the split specified in [21] for SEP-28K as they make the dataset balanced across positive and negative classes.

4. WORD LEVEL STUTTERING SPEECH DATASET

To make our system clinically applicable, students in speech-language pathology, trained in fluency disorders and stuttering, transcribed and annotated all the speech samples using CHAT, a transcription program that is part of CLAN and the Talk-Bank initiative [29]. We used a set of codes developed to be used with CHAT [30] to annotate stuttering disfluencies and typical disfluencies. Our dataset comes from two sources:

FluencyBank: This dataset included interview data from 36 adult individuals who stutter from the FluencyBank [31] English Voices-AWS Corpus.

Bilinguals Speech: This dataset included narrative samples produced in English by 62 bilingual adults, 6 of whom stuttered. Participants were asked to generate a story based on a wordless picture book. In our evaluation, we have two partitions for this dataset: *Stuttering Bilinguals Speech*, *Non-stuttering Bilinguals Speech*.

To conduct a word-level stuttering evaluation on the CHAT annotations, we need timestamp information. To this end, we first leverage Whisper Large [32] to transcribe the corresponding audio file and get the transcripts as well as the

timestamps for each word. Then, we employ Dynamic Time Warping(DTW) to align the Whisper ASR transcripts with the human transcriptions in CHAT files and slice the audio into multiple utterances according to the lines in CHAT files. For each aligned word between the Whisper ASR transcript and CHAT file transcript, we label the word as stutter if there is any stuttering annotation in its CHAT annotation, which includes: Prolongation, Epenthesis, Broken Word, Block, Sound/Syllable Repetition. We add a little margin (less than 0.2 sec) on the Whisper timestamps of each word due to the fact that Whisper truncates the audio segment where Block manifests. Notice that we only evaluate those words with exact alignment in DTW since we do not have word-level timestamps for unaligned words. We end up obtaining about 14.43 min of *Non-stuttering Bilinguals Speech*, 8.28 min of *Stuttering Bilinguals Speech*, 45.69 min of *FluencyBank* and a total of 1.14 hr recordings.

5. EVALUATION

While we aim to evaluate the performance of our model on word-level stuttered speech detection, we also want to compare it with previous works that studied utterance-level stuttering detection. Hence, we evaluate under both the utterance-level and word-level stuttering detection conditions. For the former, we evaluate on SEP-28K testing set and also report the F1 score for each stuttering type on Fluency Bank (a subset of SEP-28K). For the latter, we evaluate models on our word-level stuttering dataset and report F1, precision, and recall on each partition separately. Additionally, we report the F1 score and Average precision on the entire dataset. We choose to report F1 score instead of other metrics such as accuracy due to the fact that the data is highly imbalanced (There are many more non-stuttered speech segments than stuttered segments). We sweep the detection threshold from 0 to 1 with a step of 0.05 to find the optimal threshold for all settings. Throughout our evaluation, we found out that the word-level dataset is very sensitive to the threshold selection. Hence, we also report Average Precision on the entire word-level stuttered speech dataset. F1 score on the one hand is more intuitive but sensitive to threshold selection especially on this dataset, while Average Precision is a threshold-independent metric that evaluates the classification model under all possible thresholds. With these two evaluation benchmarks, we further conduct ablation experiments on the effect of CTC loss, the SSL interface/model selection, and the size of the finetuning dataset. Finally, we show some qualitative examples of our model's outputs.

5.1. Utterance Level stuttered speech detection

Since our model is only trained to detect stuttering events and not to distinguish between different types of stuttering, we report the type-specific stuttering detection F1 score by selecting different subsets of the testing dataset. For example, to test the F1 score for blocks on SEP-28K - Fluency Bank, we filter the testing set to include only utterances with blocks and utterances without stuttering.

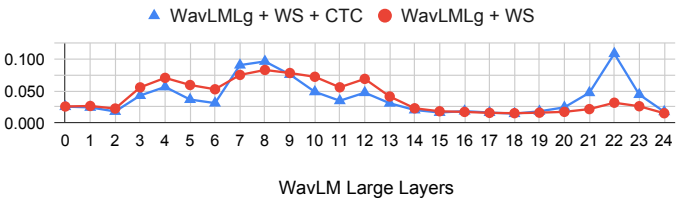

Fig. 2. The learned weights distribution of weighted sum interface in WavLMLg + WS and WavLMLg + WS + CTC.

As shown in Table 1, overall, our model surpasses the baseline systems. Specifically, [18, 19] and our model outperforms [21] by a large margin, indicating the benefit of using Speech SSL models and probably the transformer architecture. Interestingly, our model beats [18, 19] in general, except for interjections. This could potentially be improved by adding synthesized sound interjections in the pretraining stage.

5.2. Word Level stuttered speech detection

In Table 2, overall "WavLMLg + HConv. + CTC" outperforms the baseline model and other variations. In terms of different partitions, Fluency Bank has the highest F1 scores regardless of method. The reason might be the difference between native speakers and bilingual speakers. For Bilinguals Speech, stuttering bilinguals tend to have higher F1 scores compared to non-stuttering bilinguals. The reason is that our model tends to have a high recall and a low precision in most cases, which is also true for the baseline model. Compared to the baseline method, our model improves more on recall than precision. One avenue for future work would be to incorporate an improved prior over where stuttered speech is likely to occur within an utterance or a better way to condition the model on different patient demographics.

5.3. The effect of CTC loss and SSL interface selection

To study the effect of CTC loss and HConv. Interface we design several ablation experiments. For the utilization of WavLM, we design models using weighted sum (which is a set of learnable weights that is multiplied layer-wisely on WavLM Large hidden layers and sum up) or simply select a single layer. Due to the extensive computing, we select every 4 layers of WavLM Large for our experiment. For HConv. and Weighted Sum, we also try adding CTC loss or not.

On the utterance level (See Table 2), we see that with the help of CTC loss, the performance with both HConv. and Weighted Sum layer aggregation strategies improved. Interestingly, without CTC loss, both interfaces perform roughly the same overall. We hypothesize that HConv. could benefit more from the guidance of CTC loss in terms of word-level stuttered speech detection. The same trend can also be

Model	SEP-28K Test	SEP-28K - Fluency Bank					
		All	Blocks	Interjections	Prolongation	Sound Rep.	Word Rep.
Baseline [18]†	–	–	0.33	**0.84**	0.60	0.60	0.43
Baseline [19]†	–	–	0.17	0.82	0.60	0.63	0.47
Baseline [21]	0.700	0.68	0.34	0.62	0.28	0.39	0.39
WavLMLg + HConv.	0.800	**0.85**	**0.63**	0.59	**0.63**	**0.75**	**0.70**
WavLMLg + HConv. + CTC	**0.803**	0.84	0.62	0.57	0.61	0.74	0.67
	Using Weighted Sum (WS)						
WavLMLg + WS	0.789	0.82	0.58	0.63	0.54	0.70	0.63
WavLMLg + WS + CTC	0.790	0.82	0.59	0.59	0.55	0.69	0.62
	Different Finetune Dataset Size using "WavLMLg + HConv. + CTC"						
Data Ratio: 0%	0.685	0.70	0.36	0.59	0.31	0.43	0.37
Data Ratio: 25%	0.774	0.82	0.62	0.58	0.61	0.77	0.72
Data Ratio: 50%	0.798	0.84	0.65	0.57	0.64	0.76	0.71
Data Ratio: 75%	0.793	0.83	0.66	0.56	0.65	0.79	0.74
Data Ratio: 100%	0.803	0.84	0.62	0.57	0.61	0.74	0.67

Table 1. Utterance Level Stuttered Speech Detection Evaluation on SEP-28K and Fluency Bank. The numbers are F1 scores. Notice that for models with †, they are trained to distinguish different stuttering types while training. Also the numbers are directly reported from their paper.

Model	All		Stut. Blngl. Speech			Non-stut. Blngl. Speech			Stut. Fluency Bank		
	F1	Avg. P	F1	P	R	F1	P	R	F1	P	R
Baseline [21]	0.411	0.864	0.389	0.360	0.423	0.342	0.245	0.565	0.440	0.315	0.728
WavLMLg + HConv.	0.536	0.899	0.484	0.364	0.723	0.496	0.391	0.680	0.585	0.496	0.715
WavLMLg + HConv.+ CTC	**0.554**	**0.927**	**0.490**	0.375	0.708	**0.518**	0.439	0.633	0.591	0.508	0.708
	Using Weighted Sum (WS)										
WavLMLg + WS.	0.536	0.898	0.425	0.356	0.529	0.425	0.356	0.529	0.594	0.507	0.718
WavLMLg + WS. + CTC	0.540	0.888	0.460	0.336	0.730	0.403	0.344	0.486	**0.603**	0.514	0.729
	Select Single WavLM Large Layer										
WavLMLg Layer 0	0.439	0.864	0.340	0.214	0.825	0.279	0.192	0.507	0.519	0.415	0.694
WavLMLg Layer 4	0.479	0.876	0.403	0.292	0.650	0.332	0.256	0.471	0.539	0.434	0.709
WavLMLg Layer 8	0.499	0.875	0.442	0.389	0.511	0.424	0.341	0.561	0.541	0.476	0.628
WavLMLg Layer 12	0.522	0.891	0.458	0.331	0.745	0.417	0.334	0.554	0.581	0.493	0.707
WavLMLg Layer 16	0.548	0.910	0.480	0.418	0.562	0.513	0.462	0.576	0.589	0.513	0.692
WavLMLg Layer 20	0.552	0.903	0.486	0.378	0.679	0.488	0.415	0.594	0.592	0.521	0.684
WavLMLg Layer 22	0.550	0.912	0.488	0.361	0.752	0.484	0.391	0.633	0.590	0.522	0.679
WavLMLg Layer 24	0.504	0.901	0.464	0.366	0.635	0.429	0.345	0.568	0.546	0.470	0.651
	Different Finetune Dataset Size using "WavLMLg + HConv. + CTC"										
Data Ratio: 0%	0.469	0.872	0.377	0.299	0.511	0.305	0.259	0.371	0.545	0.457	0.674
Data Ratio: 25%	0.530	0.892	0.471	0.405	0.562	0.362	0.452	0.302	0.579	0.487	0.713
Data Ratio: 50%	0.533	0.904	0.470	0.338	0.774	0.493	0.413	0.612	0.579	0.487	0.713
Data Ratio: 75%	0.516	0.888	0.460	0.368	0.613	0.386	0.417	0.360	0.557	0.472	0.678
Data Ratio: 100%	0.554	0.927	0.490	0.375	0.708	0.518	0.439	0.633	0.591	0.508	0.708

Table 2. Word Level Stuttered Speech Detection Evaluation. "WavLMLg" stands for WavLM Large. HConv. indicates Hierarchical Convolution Interface and WS means weighted sum interface. For the metrics, "Avg. P" stands for Average Precision while "P" and "R" indicate precision and recall respectively. "Stut. Blngl. Speech" and "Non-stut. Blngl. Speech" stands for Stuttering Bilinguals Speech and Non-stuttering Bilinguals Speech.

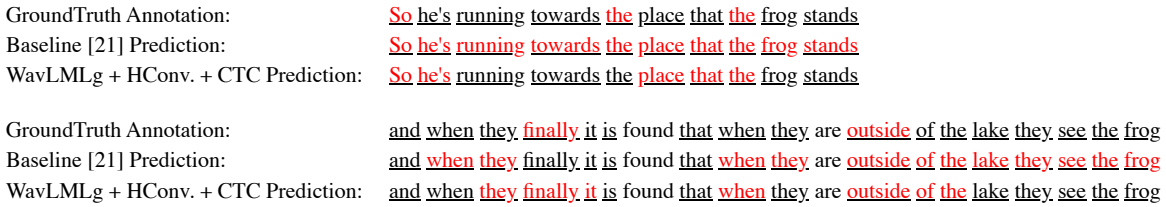

Fig. 3. Examples of our model and baseline model on word-level stuttering speech detection. Only the words with underscores are considered when evaluating. Words in red indicate that it is categorized as stuttered speech.

Model	SEP-28K Test	Word Level Stuttered
Baseline [21]	0.700	0.411
WavLM Large	**0.803**	**0.554**
WavLM Base	0.747	0.484
Wav2vec2 Large	0.760	0.524
Wav2vec2 Base	0.753	0.483
Data2vec Large	0.751	0.549
Data2vec Base	0.745	0.530

Table 3. The F1 score for using different Speech SSL Models. All models are trained using Hierarchical Convolution and CTC loss.

observed on the word-level stuttering speech dataset in Table 2. Hence, we further visualize the learned weights of the weighted sum in "WavLMLg + WS" and "WavLMLg + WS + CTC" in Figure 2. With the help of CTC loss, the weight distribution becomes more concentrated on the peaks, indicating that CTC loss is helping the model to learn a better way of utilizing upstream model features.

We also report the performance of using a single layer from WavLM-Large in Table 2. Besides experiments with every four layers of WavLM-Large, we also include an experiment using the 22nd layer of WavLM-Large according to the layer with the highest learned weight in Figure 2. Interestingly, layers 8 and 22 have the highest learned weights, but overall the best performance falls roughly on the upper half of WavLM-Large layers (from layer 12 to 24). We note that "WavLMLg Layer 22" has an overall F1 score close to our best model "WavLMLg + HConv.+ CTC", but falls behind in terms of average precision, indicating that the HConv. interface is an effective approach to utilizing WavLM-Large on word-level stuttered speech detection.

5.4. The effect of different Speech SSL models

In addition to the interface selection, we also tried using different sizes and types of upstream Speech SSL models. As shown in Table 3, overall, the Speech SSL models outperform the baseline model, indicating the effectiveness of self-supervised training. Furthermore, Large size models surpass Base size models in all cases. Last but not least, WavLM performs the best compared to Wav2vec2 [20] and Data2vec [33]. From our results, we believe that WavLM is a more suitable choice for stuttering detection and that previous works [17, 18, 19] might also benefit if they change their backbone from Wav2vec2 to WavLM.

5.5. The effect of fine-tuning dataset size

To test our model on the effect of fine-tuning dataset size, we tried using different proportions of our fine-tuning data: $0\%, 25\%, 50\%, 75\%, 100\%$. Notice that we keep the ratio of positive and negative examples in the training set the same while using different proportions of the data. We report both utterance and word level results on Table 1 and Table 2. On both datasets, we observe a large improvement from 0% to 50%. Surprisingly, there is a little drop in the performance when using 75% of the data, but there's a large leap when using 100 $\%$ of the data. Overall, this again corroborates the problem of data scarcity in stuttered speech detection and shows our model's scalability.

5.6. Qualitative results of Stuttering Detection

To visualize our detection results, we show some of the examples in our word-level stuttered speech detection dataset. As shown in Figure 3, both the baseline and our model tend to predict stuttering events more frequently than the speech pathologist. However, compared with the baseline model, our model's predictions tend to be of higher precision.

6. CONCLUSIONS

In this paper, we shed light onto a more fine-grained evaluation of stuttered speech detection which is also closer to clinical usage. We curated a clinical word-level stuttering speech dataset and further proposed a word-level stuttered speech detection model which exhibits significant improvements compared to prior work. Additionally, through our comprehensive ablation studies, we investigated the utilization of Speech SSL models on stuttered speech detection and demonstrated the potential scaling of our method.

For future work, we think that increasing the dataset size and diversity would bring additional improvements for this field. Finally, we would also like to incorporate word-level stuttering type classification in our model, to better assist speech pathologists for screening and diagnosis.

7. ACKNOWLEDGEMENT

This research has been supported by NSF Grants AF 1901292, CNS 2148141, IFML CCF 2019844 and research gifts by Cisco, WNCG IAP and the UT Austin Machine Learning Lab (MLL).

8. REFERENCES

[1] Anne Smith and Christine Weber, "How stuttering develops: The multifactorial dynamic pathways theory," *Journal of speech, language, and hearing research : JSLHR*, vol. 60, pp. 1–23, 08 2017.

[2] Ashley Craig, Elaine Blumgart, and Yvonne Tran, "The impact of stuttering on the quality of life in adults who stutter," *Journal of Fluency Disorders*, vol. 34, no. 2, pp. 61–71, 2009.

[3] Caroline Koedoot, Clazien Bouwmans, Marie-Christine Franken, and Elly Stolk, "Quality of life in adults who stutter," *Journal of Communication Disorders*, vol. 44, no. 4, pp. 429–443, 2011.

[4] Deborah J. Brisk, E. Charles Healey, and Karen A. Hux, "Clinicians' training and confidence associated with treating school-age children who stutter," *Language, Speech, and Hearing Services in Schools*, vol. 28, no. 2, pp. 164–176, 1997.

[5] Rodney M. Gabel, "School speech-language pathologists' experiences with stuttering: an ohio survey," *eHearsay*, vol. 3, no. 1, pp. 5–23, 2013.

[6] Ellen M. Kelly, Jane S. Martin, Kendra E. Baker, Norma I. Rivera, Jane E. Bishop, Cindy B. Krizizke, Deborah S. Stettler, and June M. Stealy, "Academic and clinical preparation and practices of school speech-language pathologists with people who stutter," *Language, Speech, and Hearing Services in Schools*, vol. 28, no. 3, pp. 195–212, 1997.

[7] Kenneth St. Louis and C Durrenberger, "What communication disorders do experienced clinicians prefer to manage?," *ASHA*, vol. 35, pp. 23–31, 35, 01 1994.

[8] Geoffrey A. Coalson, Courtney T. Byrd, and Elizabeth Rives, "Academic, clinical, and educational experiences of self-identified fluency specialists," *Perspectives of the ASHA Special Interest Groups*, vol. 1, no. 4, pp. 16–43, 2016.

[9] Shakeel Ahmad Sheikh, Md Sahidullah, Fabrice Hirsch, and Slim Ouni, "StutterNet: Stuttering Detection Using Time Delay Neural Network," in *EUSIPCO 2021 - 29th European Signal Processing Conference*, Dublin / Virtual, Ireland, 2021.

[10] Tedd Kourkounakis, Amirhossein Hajavi, and Ali Etemad, "Fluentnet: End-to-end detection of stuttered speech disfluencies with deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2986–2999, 2021.

[11] Colin Lea, Vikramjit Mitra, Aparna Joshi, Sachin Kajarekar, and Jeffrey Bigham, "Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter," in *ICASSP*, 2021.

[12] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[13] Shuwen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech 2021*, 2021, pp. 1194–1198.

[14] Alexei Baevski, Wei-Ning Hsu, Alexis Conneau, and Michael Auli, "Unsupervised speech recognition," *NeurIPS*, 2021.

[15] Alexander H. Liu, Wei-Ning Hsu, Michael Auli, and Alexei Baevski, "Towards end-to-end unsupervised speech recognition," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 221–228.

[16] Junyi Peng, Themos Stafylakis, Rongzhi Gu, Oldřich Plchot, Ladislav Mošner, Lukáš Burget, and Jan Černocký, "Parameter-efficient transfer learning of pre-trained transformer models for speaker verification using adapters," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[17] Payal Mohapatra, Bashima Islam, Md Tamzeed Islam, Ruochen Jiao, and Qi Zhu, "Efficient stuttering event detection using siamese networks," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[18] Sebastian Peter Bayerl, Dominik Wagner, Elmar Noeth, and Korbinian Riedhammer, "Detecting Dysfluencies in Stuttering Therapy Using wav2vec 2.0," in *Proc. Interspeech 2022*, 2022, pp. 2868–2872.

[19] Sebastian P. Bayerl, Dominik Wagner, Ilja Baumann, Florian Hönig, Tobias Bocklet, Elmar Nöth, and Korbinian Riedhammer, "A Stutter Seldom Comes Alone – Cross-Corpus Stuttering Detection as a Multi-label Problem," in *Proc. INTERSPEECH 2023*, 2023, pp. 1538–1542.

[20] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: a framework for self-supervised learning of speech representations," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2020, NIPS '20, Curran Associates Inc.

[21] John Harvill, Mark Hasegawa-Johnson, and Chang D. Yoo, "Frame-Level Stutter Detection," in *Proc. Interspeech 2022*, 2022, pp. 2843–2847.

[22] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Micheal Zeng, and Furu Wei, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1505–1518, 2021.

[23] Peter Howell, Stephen Davis, and Jon Bartrip, "The university college london archive of stuttered speech (uclass)," *Journal of speech, language, and hearing research : JSLHR*, vol. 52, pp. 556–69, 05 2009.

[24] Sebastian Bayerl, Alexander Wolff von Gudenberg, Florian Hönig, Elmar Noeth, and Korbinian Riedhammer, "Ksof: The kassel state of fluency dataset – a therapy centered dataset of stuttering," in *Proceedings of the Language Resources and Evaluation Conference*, Marseille, France, June 2022, pp. 1780–1787, European Language Resources Association.

[25] Farhad Javanmardi, Saska Tirronen, Manila Kodali, Sudarsana Reddy Kadiri, and Paavo Alku, "Wav2vec-based detection and severity level classification of dysarthria from speech," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[26] Jiachen Lian, Carly Feng, Naasir Farooqi, Steve Li, Anshul Kashyap, Cheol Jun Cho, Peter Wu, Robin Netzorg, Tingle Li, and Gopala Krishna Anumanchipalli, "Unconstrained dysfluency modeling for dysfluent speech transcription and detection," *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8, 2023.

[27] Jiachen Lian and Gopala Anumanchipalli, "Towards hierarchical spoken language disfluency modeling," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, St. Julian's, Malta, Mar. 2024, pp. 539–551, Association for Computational Linguistics.

[28] Yi-Jen Shih and David Harwath, "Interface Design for Self-Supervised Speech Models," in *Proc. INTERSPEECH 2024*, 2024.

[29] Brian Macwhinney, "The childes project: Tools for analyzing talk: Transcription format and programs (3rd ed.).," *Lawrence Erlbaum Associates Publishers*, vol. 1, 2000.

[30] Nan Bernstein Ratner and Shelley B. Brundage, "A clinician's complete guide to clan and praat," 2020.

[31] Nan Bernstein Ratner and Brian MacWhinney, "Fluency bank: A new resource for fluency research and practice," *Journal of Fluency Disorders*, vol. 56, pp. 69–80, 2018.

[32] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the 40th International Conference on Machine Learning*. 23–29 Jul 2023, vol. 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, PMLR.

[33] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli, "data2vec: A general framework for self-supervised learning in speech, vision and language," in *Proceedings of the 39th International Conference on Machine Learning*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, Eds. 17–23 Jul 2022, vol. 162 of *Proceedings of Machine Learning Research*, pp. 1298–1312, PMLR.