

# FluencyBank Timestamped: An Updated Dataset for Disfluency Detection and Automatic Intended Speech Recognition

Amrit Romana, Minxue Niu, Matthew Perez, Emily Mower Provost

January 2024

Corresponding Author: Amrit Romana, 48103, [aromana@umich.edu](mailto:aromana@umich.edu)

Conflict of Interest Statement: The authors declare that they have no conflict of interest regarding the publication of this paper.

## Abstract

**Purpose:** This work introduces updated transcripts, disfluency annotations, and word timings for FluencyBank, which we refer to as FluencyBank Timestamped. This dataset will enable thorough analysis of how speech processing models (such as speech recognition and disfluency detection models) perform when evaluated with typical speech versus speech from people who stutter (PWS).

**Method:** We update the FluencyBank dataset, which includes audio recordings from adults who stutter, to explore the robustness of speech processing models. Our update (semi-automated with manual review) includes new transcripts with timestamps and disfluency labels corresponding to each token in the transcript. Our disfluency labels capture typical disfluencies (filled pauses, repetitions, revisions, and partial words), and we explore how speech model performance compares for Switchboard (typical speech) and FluencyBank Timestamped. We present benchmarks for three speech tasks: intended speech recognition, text-based disfluency detection, and audio-based disfluency detection. For the first task, we evaluate how well Whisper performs for intended speech recognition (i.e., transcribing speech without disfluencies). For the next tasks, we evaluate how well a BERT text-based model and a Whisper audio-based model perform for disfluency detection. We select these models, BERT and Whisper, as they have shown high accuracies on a broad range of tasks in their language and audio domains, respectively.

**Results:** For the transcription task, we calculate an intended speech word error rate (isWER) between the model's output and the speaker's intended speech (i.e., speech without disfluencies). We find isWER is comparable between Switchboard and FluencyBank Timestamped, but within FluencyBank Timestamped, isWER increases with stuttering severity. We find that Whisper transcribes a large portion of revisions in both datasets but that it transcribes filled pauses and partial words at a higher rate in FluencyBank Timestamped. For the disfluency detection tasks, we find the models detect filled pauses, revisions, and partial words relatively well in FluencyBank Timestamped, but performance drops substantially for repetitions because the models are unable to generalize to the different types of repetitions (e.g., multiple repetitions or sound repetitions) from PWS. We hope that the FluencyBank Timestamped will allow researchers to explore closing performance gaps between typical speech and speech from PWS.

**Conclusions:** Our analysis shows that there are gaps in speech recognition and disfluency detection performance between typical speech and speech from PWS. We hope that FluencyBank Timestamped will contribute to more advancements in training robust speech processing models.

# 1 Introduction

Automatic speech recognition (ASR) is a fundamental part of spoken technology such as voice assistants or dictation services. However, research has shown that disruptions in the flow of speech, known as speech disfluencies, negatively impact the accuracy and interpretability of speech recognition output (Goldwater et al., 2010). Addressing this shortcoming is crucial, as all individuals experience disfluencies as they speak. Furthermore, the rates and types of disfluencies may be heightened by various factors, making disfluency detection itself beneficial for downstream applications. For example, for typical speakers, research indicates that disfluency rates tend to rise with planning demands, such as when speaking on an unfamiliar topic (Bortfield et al., 2001). Furthermore, people who stutter (PWS) experience higher rates of disfluencies and in particular higher rates of repetitions, prolongations, or blocks. PWS may feel stuck, as though they are unable to speak the way they would like to speak, and they may experience physical, emotional, and cognitive reactions (Tichenor & Yaruss, 2019). For PWS, disfluency detection may help clinicians detect overt stuttering characteristics and monitor behavioral severity (Riad et al., 2020; Bayerl et al., 2022). In this work, we present an updated dataset: FluencyBank Timestamped, which we can use to explore both tasks: speech recognition and disfluency detection. FluencyBank Timestamped includes speech from PWS, timestamped transcripts, and labels for typical disfluencies (filled pauses, repetitions, revisions, and partial words). We juxtapose disfluency patterns from typical speakers (Switchboard) and PWS (FluencyBank Timestamped), and we answer two fundamental questions related to speech modeling and disfluencies: 1. How does the performance of speech recognition models change for speech that displays different disfluency patterns (such as from PWS)? 2. How does the performance of disfluency detection models change for speech that displays different disfluency patterns? Addressing these questions is one step toward achieving equal access to speech technology.

A main challenge in evaluating the use of these speech processing methods with speech from PWS is the lack of public data. A dataset should include audio, transcripts, disfluency annotations, and word timings. Researchers are beginning to collect these detailed datasets (Mitra et al., 2021; Shonibare et al., 2022; Lea et al., 2023), but they have not been publicly released yet. The few public corpora that have been released are typically missing data components that make it difficult to align the datasets for tasks such as speech recognition or disfluency detection. Howell et al. (2009) released UCLASS which contains audio files from PWS, and a portion of these speech files have been transcribed orthographically allowing researchers to derive disfluency detection targets. Kourkounakis et al. (2021) evaluate segment-level disfluency detection methods with UCLASS, where they focus on classifying entire audio

segments as disfluent rather than locating the disfluency within the segment. However, the main limitation of UCLASS is that just over one hour of the audio has been transcribed. More recently, Lea et al. (2021) released SEP-28k with 3.5 hours of speech from PWS and segment-level disfluency labels such as whether the segment contains a repetition, prolonged sound, or block. Researchers have widely used this dataset for segment-level disfluency detection (Bayerl et al., 2022; Jouaiti & Dautenhahn, 2022), however, the lack of transcripts and the lack of word-level disfluency annotations limits analysis. In their current state, these datasets cannot be used for evaluating speech recognition or disfluency detection at a fine temporal resolution (illustrated in Figure 2). Other work has looked at artificially inserting stuttering-like disfluencies to address the lack of data, but there are open questions as to whether synthetic data accurately reflects the complexity of stuttering (Kourkounakis et al., 2021). To better evaluate tasks such as speech recognition or disfluency detection, a new corpus should include audio from PWS, speech transcripts, word-level disfluency labels, and word timestamps. We introduce FluencyBank Timestamped, an updated version of FluencyBank, to address these dataset needs.

Ratner and MacWhinney (2018) introduced FluencyBank as a resource with speech from PWS. They included audio, transcripts, segment timestamps, and word-level disfluency annotations in the CHAT (Codes for the Human Analysis of Transcripts) format (MacWhinney, 2000). However, these public-facing transcripts and annotations are intentionally incomplete so FluencyBank can be used as a resource in which students can practice fluency assessment among other tasks. For example, filled pauses and sound repetitions that are present in the audio are often not included in these transcripts. Tasks such as ASR or disfluency detection would benefit from complete transcripts that include timestamps and disfluency annotations. As a partial solution, Lea et al. (2021) released updated segments and labels for FluencyBank, but the dataset does not include transcripts of any kind, and their enhanced disfluency labels correspond to entire segments, which limits the granularity of potential analysis. In this paper, we introduce a version of FluencyBank that we refer to as FluencyBank Timestamped, which includes transcripts, word-level timestamps, and word-level disfluency annotations. This updated dataset can be found on the FluencyBank portal but is password protected to ensure that students who are using FluencyBank to practice fluency scoring do not have complete transcripts with disfluency annotations available. We evaluate how performance on two speech modeling tasks (speech recognition and disfluency detection) generalize to speech from PWS.

We first explore an ASR benchmark. In the last few years, ASR accuracy has improved dramatically largely due to increased computational power, new deep learning techniques, and greater data availability. These developments

have contributed to more widespread adoption of ASR technologies including voice assistants and dictation systems. However, the accuracy of these models is significantly hindered by disfluencies (Goldwater et al., 2010). Additionally, a recent survey has found that PWS struggle to use speech-based technology due to ASR underperforming for PWS (Lea et al., 2023). For PWS, there is a gap between what they say verbatim, which may include a high rate of disfluencies, and their intent, which typically does not include disfluencies. Lea et al. find that fine-tuning (i.e., further training) an ASR model with the intended speech from PWS improves transcription performance, yet the results still lag behind those achieved with typical speech. In this paper, we compare how well an open-source ASR model generalizes to intended speech transcription with typical speech (Switchboard) and PWS (FluencyBank Timestamped).

Additionally, we explore disfluency detection benchmarks. Depending on its framing, disfluency detection aims to detect disfluent words from a transcript (text-based) or disfluent frames from audio (audio-based). Disfluency detection has several promising applications for typical speakers as well as for PWS. For typical speakers, disfluency detection can be incorporated into tools to help monitor cognitive load (Müller et al., 2001). For PWS, disfluency detection may help clinicians monitor stuttering severity (Riad et al., 2020; Bayerl et al., 2022). Finally, for PWS, research has shown that incorporating disfluency detection into ASR can improve transcription accuracy (Shonibare et al., 2022). In this paper, we evaluate how well text- and audio-based disfluency detection models generalize to speech from PWS in the FluencyBank Timestamped dataset.

In summary, in this paper we make the following contributions:

- We introduce FluencyBank Timestamped, in which we update the transcripts, timings, and disfluency labels associated with the 5.3 hours of speech in the Adults who Stutter portion of FluencyBank dataset. We make these data public to allow researchers to more thoroughly evaluate how speech technologies perform with speech from PWS.
- We compare how disfluency types and rates vary between typical speakers and PWS. This allows us to gain insight into the flexibility that speech models need to effectively process a diverse range of speech.
- We present zero-shot benchmarks (meaning that the models have not been trained using speech from this domain) for intended speech recognition, text-based disfluency detection, and audio-based disfluency detection with FluencyBank Timestamped. We identify areas in which these models need to improve to process speech

from PWS more accurately.

## **2 Dataset**

### **2.1 FluencyBank Original**

The original FluencyBank dataset includes audio-visual data from children and adults who stutter (Ratner & MacWhinney, 2018). The data were collected with a number of potential activities in mind, including practicing fluency assessment and learning about the behavioral, affective, and cognitive aspects of living with stuttering.

In our work, we focus on the audio from the adults who stutter subset, where adults are recorded as they complete an interview about their experience as a stutterer. This interview includes five open-ended prompts such as “Please talk about the impact of stuttering on your daily life” and “What do you think causes stuttering?” In this dataset, 37 participants are recorded while they complete the interview. The interview length ranges from 4.5 to 25.3 minutes with an average length of 10.6 minutes. For 13 of the 37 participants, the dataset includes labels where clinicians have assessed the severity of the participants’ stuttering using the SSI-4, which is a stuttering assessment commonly used by clinicians and researchers (Riley & Bakker, 2009). According to these assessments, 6 participants are mild, 1 is mild/moderate, 3 are moderate, and 2 are moderate/severe stutterers. In the bulk of our analysis, we work with all 37 participants, but we also include a smaller analysis by stuttering severity. When we use the stuttering severity labels, we group mild and mild/moderate (7 participants) as “mild,” we group moderate and moderate/severe (5 participants) as “moderate,” and we label the other 24 participants as “unknown.”

These data have been transcribed and coded in the CHAT format (MacWhinney, 2000). The CHAT format was introduced to facilitate a wide variety of analysis and is used across TalkBank (a collection of language databases including FluencyBank). The CHAT format includes verbatim transcripts of speech with standard codes to represent linguistic features, such as segment boundaries, parts of speech, and disfluencies. In our work we focus on the CHAT disfluency codes, specifically those for filled pauses (marked by &- such as “&-um”), repetitions (marked by [/] such as “it’s [/] it’s a dog” or “<it’s a> [/] it’s a dog”), revisions (or retracings marked by [/] such as “<I wanted> [/] &-uh I thought I wanted it”), and partial words (fragments marked by &+ such as “to &+b dive” or sound repetitions marked by ⇐ such as “to ⇐d-dive”). However, these transcripts have been left intentionally incomplete so that students can practice fluency assessment.

## 2.2 FluencyBank Timestamped

In our work, we present an updated dataset, FluencyBank Timestamped, which contains new transcripts, disfluency labels, and timestamps to facilitate research on speech processing models. The new dataset has 3,430 annotated segments (5.3 hours of speech) from 37 participants. For each segment, the dataset includes an audio clip and corresponding data CSV file. In the data file, each row corresponds to a word in the transcript and the columns specify the word’s start time, end time, and categorical disfluency labels.

### Disfluency Categories

We label each word in the data file with categorical disfluency labels corresponding to filled pauses, repetitions, revisions, and partial words, with examples in Table 1. We primarily focus on these categories because they are present in typical speech but occur at higher rates in speech from PWS (Ambrose & Yairi, 1999). This allows us to provide direct comparisons of speech recognition models and disfluency detection models across typical and stuttered speech. We also note that previous work has found higher agreement when annotating these disfluency categories, as opposed to other stuttering-like disfluencies such as prolongations or blocks (Lea et al., 2021).

These categories are inspired by the CHAT format (MacWhinney, 2000) to allow easy conversion between the two. However, we condense the CHAT categories in a few ways. First, we do not differentiate between word and phrase repetitions, but given our CSV format where each word in the transcript has its own row, the specific labels for word versus phrase repetitions can be derived based on the number of consecutive words with a repetition label. Similarly, we do not differentiate between word and phrase revisions, but these labels could also be derived. Lastly, we do not differentiate between phonological fragments (“&+sn dog”) and repeated segments (“←r-r-r←rabbit”). Instead, we use a partial word tag in conjunction with the repetition or revision labels, where a partial word that is also a repetition is a repeated segment, while a partial word that is also a revision is a phonological fragment. We reduce the label complexity in these ways to both simplify the annotation process and aid the machine learning model in identifying patterns, but the full CHAT codes can be reconstructed from our labels.

### Step 1: Transcription

We use the RevPro transcription service to derive verbatim transcripts for each participant’s speech. These transcripts are generated by trained human transcribers. Their website outlines that the verbatim transcripts include filled pauses, revisions, and repetitions including sound repetitions, and that they guarantee the transcripts to be 99%

accurate (Rev, 2024). However, given the challenges that exist in generating verbatim transcripts, we validate the RevPro transcripts against the original FluencyBank transcripts.

We first standardize the text in the RevPro and FluencyBank transcripts to address spelling differences, such as “uhm” versus “um”, “etc” versus “et cetera”, “2001” versus “two thousand one.” We then use the Levenshtein distance algorithm (Levenshtein, 1966) to align the original FluencyBank text with the new RevPro text. The Levenshtein distance is a commonly used method to align two strings of different lengths, and it allows us to label tokens in each string as correct (the token is in both strings), inserted (the token is in the RevPro transcript but not the original FluencyBank transcript), deleted (the token is in the original FluencyBank transcript but not the RevPro transcript), or substituted (there are different tokens at a comparable spot in each transcript). Compared to the FluencyBank transcripts, we find the RevPro transcripts for each speaker have on average, 160 inserted tokens, 48 substituted tokens, and 14 deleted tokens. The counts vary considerably by participant, primarily as a function of how long of an interview they provided and how many disfluencies they produced. The first author manually verified deletions and substitutions with the audio to correct a small number of inaccuracies in the RevPro transcript.

## **Step 2: Disfluency Annotation**

Our disfluency annotation process begins with the original FluencyBank labels. During our transcript comparison process, we found that most words were “correct” or present in both the FluencyBank and RevPro transcripts. We preserve the disfluency labels for correct words, but we flag any insertions and substitutions for annotation. For example, if the original FluencyBank text says “I started stuttering” but the new RevPro text says “I started st- stuttering,” then we keep the original labels associated with “I started” and “stuttering,” but identify “st-” as an insertion and flag it for annotation.

We use a team of six annotators to update the labels to reflect transcript changes. The annotation group consisted of computer science PhD students with experience in speech and language research including speech annotation and atypical speech modeling. The annotation team included the first three authors of this paper, along with three others from their research lab.

We created and distributed an annotation guide that included instructions and disfluency descriptions with examples. The instructions asked annotators to review inserted words to determine if the word needed a disfluency label, and review substituted words to determine if the previously assigned disfluency label still applied. In the example text “I started st- stuttering,” an annotator would find that “st-” was inserted, and in this case, they would be



expected to label it as a partial word and a repetition. The instructions also asked annotators to review text near the insertion or substitution, and flag anything if its disfluency label needed to be updated. All annotators met, reviewed the instructions, and completed the initial 10% of the annotations in person where they could discuss any cases needing further clarification. The annotators then completed their set of annotations over the course of two weeks as their schedules allowed.

In the end, each segment was annotated by two different annotators. We calculated interrater agreement for insertions and substitutions using Cohen’s Kappa. Consistent with previous work on disfluency labeling, we found varying results for each disfluency class: for filled pauses  $\kappa=0.79$ , for repetitions  $\kappa=0.87$ , for revisions  $\kappa=0.51$ , and for partial words  $\kappa=0.92$ . This indicates substantial to almost perfect agreement for filled pauses, repetitions, and partial words, and moderate agreement for revisions. After the first annotation pass, three annotators met to review and resolve annotation flags and disagreements.

### **Step 3: Timestamping**

We used Gentle forced alignment to align the new transcripts with the audio and derive timestamps (Povey et al., 2011). Gentle successfully aligned 94% of the words in the text. Upon reviewing these results, we found that the alignment was especially difficult for partial words and repetitions: only 59% of partial words were aligned, and only 85% of repetitions were aligned. We used Audacity with the label functionality to manually review the audio and spectrograms for segments with missing timestamps, and we added timestamps for these when possible. For the remaining words that Gentle did not align and that we did not find in our manual review, we dropped the word from the transcript. In this way, we use Gentle as a second check to verify the RevPro insertions. Using this approach, we dropped 1.5% of non-disfluent words, 4.0% of filled pauses, 11.5% of repetitions, 6.5% of revisions, and 15.4% of partial words which were included in the RevPro transcripts but not aligned with the audio. Finally, we define audio segments’ start and end timings using the original FluencyBank segment-word boundaries but the new word-timestamps.

## **2.3 Switchboard**

Switchboard is a widely used dataset that contains typical speakers discussing assigned topics with strangers over the phone (Godfrey et al., 1992). Their speech naturally contains disfluencies, and the Switchboard dataset includes

transcripts, timestamps, and disfluency labels. In our previous work, we shared an approach with publicly available code to translate the Switchboard error-correction disfluency labels to disfluency categories: filled pauses, repetitions, revision, restarts, and partial words (Romana et al., 2023). In this work, we do not consider restarts because they comprised less than 0.3% of disfluencies in Switchboard. In the end, this dataset includes 189,428 segments (121 hours) of annotated speech. We use Switchboard as a training set for our disfluency detection models to evaluate zero-shot performance with FluencyBank Timestamped.

## 2.4 Comparing disfluencies across datasets

We compare disfluency rates across FluencyBank Timestamped and the Switchboard dataset. We notice that participants in the FluencyBank Timestamped datasets have higher rates of these disfluencies, and this motivates our work in exploring how different disfluency patterns impact intended speech recognition and disfluency detection models. Figure 1 shows the frequency of disfluencies in the original FluencyBank dataset, the updated FluencyBank Timestamped dataset, and the Switchboard dataset. We note that the number of labeled disfluencies in the Switchboard dataset (typical speech) are larger than the number of labeled disfluencies in the original FluencyBank dataset (speech from PWS). Once we retranscribe and update annotations in FluencyBank Timestamped, the rates of disfluencies significantly increase. At the token-level, comparing the updated and original labels, the updated labels have 1.25 times as many filled pauses, 5 times as many repetitions, 2 times as many revisions, and more than 10 times as many partial words. After the update, comparing FluencyBank Timestamped and Switchboard, FluencyBank Timestamped contains 1.5 times as many filled pauses, 3 times as many repetitions, a comparable number of revisions, and 6 times as many partial words. This is consistent with repetitions and sound repetitions being characteristic of stuttering disorders, whereas revisions are a typical disfluency (MacWhinney, 2000). The token timings for FluencyBank Timestamped and Switchboard also allow us to compare the frame-level disfluencies. We find that the token- and frame-level rates are correlated within each dataset. However, across the datasets, the different disfluency patterns highlight the need for robust model adaptation and training strategies to effectively generalize to speech from PWS.

## 3 Method

This study has been determined to be exempt from IRB review under Exemption 4(i) at 45 CFR 46.104(d) as it

involves secondary research for which consent is not required.

### **3.1 Intended Speech Recognition Benchmark**

When an individual stutters, there may be a gap between what they said (including disfluencies) and what they intended to say (Tichenor & Yaruss, 2019). Researchers are working toward intended speech recognition as this would make voice assistants and dictation services more accessible for those with high rates of disfluencies (Lea et al., 2023). However, to the best of our knowledge, there are no publicly available datasets one could use for this specific task, and FluencyBank Timestamped fills this need. For intended speech targets (i.e., ground truth), we combine the transcripts and disfluency labels to remove all disfluencies from the text, although we acknowledge that this may change the underlying message of the segment (Tomanek et al., 2023). We then evaluate how well ASR systems predict these targets. In other words, given an audio segment that contains disfluencies, how well does the ASR system perform in transcribing the speech without disfluencies?

We evaluate Whisper, a widely used open-source ASR model (Radford et al., 2023) that we found in our previous work drops disfluencies in the resulting transcript at a higher rate than other off-the-shelf systems (Romana et al., 2023). Whisper is an encoder-decoder transformer model that has been trained on 680,000 hours of noisy data collected from the web. These training choices have allowed Whisper to generalize well to unseen data. Furthermore, the Whisper implementation provided by Linagora Lab (Louradour, 2023; Giorgino, 2009) includes a flag to detect any disfluencies so they can be removed from the resulting transcript. However, to the best of our knowledge, this flag has not been evaluated with speech from PWS. We compare how well Whisper with the disfluency removal flag performs for intended speech recognition on Switchboard and FluencyBank Timestamped. We specifically use Whisper-small which has 244 million parameters and is available through Huggingface (Wolf et al., 2019).

We first evaluate the model in terms of intended speech word error rate (isWER) or the word error rate between the ASR-generated speech and the speaker’s intended speech target (Mitra et al., 2021). While isWER allows us to evaluate how well Whisper performs for intended speech recognition overall, it does not provide insight into the types of tokens Whisper inserts or deletes when aligned with the intended speech targets. As a solution, we use the Levenshtein distance algorithm to generate an alignment between a segment’s ASR-generated intended speech and ground truth verbatim speech (i.e., all spoken words including disfluencies). Using this alignment, we report what portion of disfluencies Whisper erroneously includes in the generated transcripts.

### **3.2 Text-Based Disfluency Detection Benchmark**

A text-based approach for disfluency detection takes a verbatim transcript (i.e., all spoken words including disfluencies) as input and detects disfluencies related to each token in the transcript. Text-based disfluency detection has the potential to aid speech-language pathologists in annotating speech from assessments. Text-based disfluency detection may also aid ASR systems in removing disfluencies for intended speech transcription. We create the text-based approach by training a BERT model that takes tokens from the manually transcribed text as input and predicts a disfluency label for each token. We illustrate this process in Figure 2.

BERT (Bidirectional Embedding Representations from Transformers) is a transformer-based language model that has been shown to be useful on a range of tasks including question answering, next sentence prediction, and sentence acceptability classification (Devlin et al., 2018). We and other researchers have found that BERT can be fine-tuned to achieve high accuracy in disfluency detection (Rocholl et al., 2021; Romana et al., 2022; Romana et al., 2023). In this paper, we fine-tune BERT on text with disfluencies from typical speakers (Switchboard) and evaluate how it performs in a zero-shot manner on text with disfluencies from PWS (FluencyBank Timestamped).

We fine-tune the full architecture (encoder and output layer) using the Switchboard training set. We use a batch size of 64 and use the Adam optimizer to minimize binary cross entropy loss. We use the development sets to optimize the learning rate (1e-4, 5e-5, 1e-5) and the number of training steps (up to 15 epochs). We evaluate performance on the development set every 500 training steps, and we choose the best model based on unweighted average recall (UAR) and a patience of 5. We repeat this training using 3 random seeds, and report average and standard deviation of recall and F1 score. For the text-based approach, we calculate these metrics at the word-level.

### **3.3 Audio-based Disfluency Detection Benchmark**

One practical limitation of the text-based disfluency detection approach is that it relies heavily on the transcript including all disfluencies. ASR systems may either intentionally or inadvertently remove disfluencies when transcribing speech. Our previous work has explored the impact of using ASR-transcribed text as input for text-based disfluency detection. We evaluated several commonly used ASR architectures, many of which consisted of an encoder which generated speech representations, and a decoder which used the representations to generate a transcript. Our results showed that fine-tuning an ASR system for verbatim transcription could lead to higher accuracies in text-based disfluency detection. However, we found that our best results for disfluency detection from untranscribed speech came from an audio-based approach, where we discarded the ASR decoder and fine-tuned just the encoder for frame-level disfluency detection (Romana et al., 2023). In this work, we use a similar audio-based architecture but we update our

encoder to the Whisper encoder. We illustrate the full approach in Figure 2. In this section, we present our approach for audio-based disfluency detection.

We note that this approach is considerably more challenging than a text-based approach, but it is more scalable and provides additional information because it locates the disfluency within the audio. This may be useful for downstream speech processing applications, such as masking disfluent audio frames in speech recognition (Shonibare et al., 2022). This may also be useful for clinical measures such as separating primary and collateral (disfluent) tracks (Riad et al., 2020) or calculating a speech efficiency score (Amir et al., 2018).

We create the audio-based approach by training a Whisper encoder model that takes raw audio as input and predicts a disfluency label for every 20 ms of audio. We fine-tune the full architecture (encoder and output layer) using the Switchboard training set. We use a batch size of 64 and use the Adam optimizer to minimize binary cross entropy loss. We use the development sets to optimize the learning rate (1e-4, 5e-5, 1e-5) and the number of training steps (up to 15 epochs). We evaluate performance on the development set every 500 training steps, and we choose the best model based on unweighted average recall (UAR) and a patience of 5. We repeat this training using 3 random seeds, and report average and standard deviation of recall and F1 score. For the audio-based approach, we calculate these metrics at the frame-level.

## **4 Results and Discussion**

### **4.1 Intended Speech Recognition**

Whisper attains an isWER of 15.4% on FluencyBank Timestamped. These results show a considerable improvement over the isWER reported for FluencyBank in previous work (Mitra et al. isWER=38.7% (2021)). We suspect that this is in part due to differences between their in-house transcripts versus our RevPro transcripts used for evaluation, but it is primarily due to Whisper outperforming previously used ASR models. We find that overall, Whisper performs comparably with speech from PWS relative to typical speech (Switchboard isWER=15.2%). However, we analyze the results across different levels of stuttering severity for the 13 out of 37 participants that have stuttering severity labels, and we find that stuttering severity negatively impacts isWER. For mild stutterers (7 participants) isWER is 8.9%, but for moderate stutterers (5 participants) isWER increases to 12.3%.

We compare the rates at which Whisper transcribes disfluencies in Switchboard and FluencyBank Timestamped.

For these metrics, a lower score suggests better performance because it indicates that the model does better at skipping disfluencies during transcription. Overall, we find that Whisper transcribes disfluencies in FluencyBank Timestamped at a higher rate than disfluencies in Switchboard, with filled pauses and partial words particularly affected. Whisper transcribes 8.7% and 13.1% of filled pauses from Switchboard and FluencyBank, respectively. Whisper transcribes 9.4% and 10.2% of repetitions from Switchboard and FluencyBank, respectively. Whisper transcribes revisions at higher rates of 43.1% and 47.2% from Switchboard and FluencyBank, respectively. Whisper transcribes partial words 1.4% and 3.0% of partial words from Switchboard and FluencyBank, respectively. We note that these rates, and particularly the partial word transcription rates, are an under-estimation since Whisper may, for example, transcribe a partial word but not spell it the same as is in the verbatim transcript. Table 2 lists these results. While Whisper removed most disfluencies, we still find cases where the model incorrectly transcribed them in the FluencyBank Timestamped dataset. Table 3 showcases some examples of these errors and we encourage researchers to explore methods for improving intended speech recognition for PWS.

## 4.2 Text-Based Disfluency Detection

The text-based disfluency detection model is a BERT-based model which we train on the Switchboard training set and evaluate with the Switchboard testing set as well as with FluencyBank Timestamped. We find that the model’s performance drops by 15% between Switchboard (weighted recall=0.86) and FluencyBank Timestamped (weighted recall=0.73). With Switchboard, BERT achieves a recall score of nearly 0.9 for most classes (filled pause recall=1.00, repetition recall=0.88, partial word recall=0.88). Revisions are harder to detect, which likely stems from the increased context understanding required in detecting revisions (revision recall=0.68). In a zero-shot manner with FluencyBank Timestamped, recall is comparable for filled pauses (filled pause recall=1.00), drops moderately for revisions and partial words (revision recall=0.62, partial word recall=0.80), and drops substantially for repetitions (repetition recall=0.53). We suspect that the drop in performance for repetitions is likely due to the different types of repetitions characteristic of stuttering. For example, FluencyBank Timestamped has a higher rate of multiple repetitions (e.g., “it’s a two two two way street”) as well as sound repetitions (e.g., “I was a co- co- covert stutterer”). See Table 4 for more details on the text-based disfluency detection results.

We explore how the text-based disfluency detection results compare for participants with a mild (7 participants) or moderate (5 participants) stuttering severity label. Filled pause recall is 1.00 for participants regardless of stuttering

severity label. However, when comparing other disfluency classes, we find high levels of variation across participants and that performance drops as stuttering severity increases. Repetition recall drops from 0.68 to 0.54, revision recall drops from 0.64 to 0.62, and partial word recall drops from 0.84 to 0.79, when comparing results across participants with mild versus moderate stuttering severity labels. This suggests that the model did not generalize as well for people with higher stuttering severity. We illustrate the participant-level variation in Figure 3.

### 4.3 Audio-Based Disfluency Detection

The audio-based disfluency detection model is a Whisper encoder-based model that we train on the Switchboard training set and evaluate with the Switchboard testing set as well as with FluencyBank Timestamped. The audio model performance falls behind the text model, but this is expected as the problem is more challenging: the audio-based model aims to detect which frames in the audio contain a disfluency without any transcripts, and background noise and speech intelligibility can impact results more directly than with a text model. With the audio-based model, we find a larger drop in performance (34%) between Switchboard (weighted recall=0.65) and FluencyBank Timestamped (weighted recall=0.43). Recall drops moderately for revisions (Switchboard recall=0.46 compared to FluencyBank Timestamped recall=0.37) and partial words (Switchboard recall=0.34 compared to FluencyBank Timestamped recall=0.25), and recall drops substantially for repetitions (Switchboard recall=0.66 and FluencyBank Timestamped recall=0.29). We hypothesize that repetitions are most impacted again due to different types of repetitions being common among people with typical speech versus PWS. See Table 5 for more details on the audio-based disfluency detection results.

We explore how the audio-based disfluency detection results compare for the participants with a mild (7 participants) or moderate (5 participants) stuttering severity label. We find that filled pause recall drops from 0.89 to 0.79, repetition recall drops from 0.44 to 0.25, revision recall drops from 0.48 to 0.33, and partial word recall drops from 0.26 to 0.20, when comparing results across participants with mild versus moderate stuttering severity labels. We illustrate the participant-level variation in Figure 3 and we encourage future work to evaluate how to improve audio-based disfluency detection performance for individuals with moderate or more severe stuttering severity.

## 5 Conclusion

In this work, we present and benchmark an updated dataset: FluencyBank Timestamped. To the best of our knowledge,

FluencyBank Timestamped is the only publicly-available dataset with transcripts, disfluency annotations, and word-level timestamps corresponding to the audio to facilitate a wide-range of analysis with speech from PWS. We completed this update by first obtaining two independently generated transcript versions (the original FluencyBank and a version by Rev), merging them together, and manually correcting any discrepancies. We then manually updated disfluency annotations corresponding to each word, focusing on filled pauses, repetitions, revisions, and partial words. Finally, we used forced alignment to generate timestamps and manually corrected any missing timestamps.

We present an off-the-shelf benchmark for intended speech recognition on FluencyBank Timestamped using an open-source ASR model that is designed to transcribe intended speech: Whisper. We compare Whisper’s performance on typical speech (Switchboard) and speech from PWS (FluencyBank Timestamped), and we find that it performs comparably on the two datasets overall, but within FluencyBank Timestamped we find that transcription error rates increase with stuttering severity. We find that while Whisper correctly removes disfluencies such as partial word repetitions, it erroneously includes other disfluencies such as filled pauses, full word repetitions, and revisions. We hope this dataset will allow researchers to better understand how speech recognition models perform with speech from PWS and explore methods to reduce the disfluency transcription rate.

We also present benchmarks for zero-shot text- and audio-based disfluency detection on FluencyBank Timestamped. We find that the text-based model generalizes relatively well from typical speech to speech from PWS compared to the audio-based model. However, we find that the performance of both models drops considerably for repetitions, likely due to the different types of repetitions characteristic of stuttering. We also find that the extent to which the models generalize to FluencyBank Timestamped is partly dependent on an individual’s stuttering severity.

This analysis is one step toward achieving equal access to speech technology. While we separate the speech recognition task and disfluency detection tasks for this initial analysis, future work will explore their intersection. For example, we will explore how disfluency detection can aid intended speech transcription and if this detection should be incorporated directly into the ASR model or if it should serve as a postprocessing step. Conversely, we will explore if learning verbatim speech transcription before or alongside disfluency detection can improve detection performance. Ultimately, we hope this dataset will contribute to more advancements to close the performance gap for PWS.

## **6 Limitations**

In our work we compare speech processing performance across a dataset of typical speech (Switchboard) and a dataset



with speech from PWS (FluencyBank Timestamped). Some differences may result from the speaking tasks and audio quality across the two datasets. In terms of speaking tasks, Switchboard consists of audio-only recordings between two strangers discussing an assigned topic from a list of roughly 60 topics. On the other hand, FluencyBank consists of audio-video recordings where a PWS answers questions about their experience as a stutterer. Given the differing task demands across the two datasets (both in terms of recording type and topic), it’s probable that they elicit different disfluency patterns. Moreover, speakers in FluencyBank may use technical terms related to stuttering (e.g., “Broca’s area”) and we suspect that ASR encounters challenges with this dataset partly because of less frequently used terms. In terms of audio quality, Switchboard conversations were recorded over a landline phone at an individual’s home at 8 kHz while FluencyBank Timestamped interviews were recorded in varying environments at a sample rate of 44 kHz. We resample audio in both datasets to 16 kHz, but factors such as background noise associated with varying environments likely impacts model performance. To alleviate these concerns, we include an analysis by stuttering severity within FluencyBank Timestamped.

Another limitation of our study stems from the fact that our team does not include a speech-language pathologist. Research has shown that disfluency annotation is a challenging process and that even experts disagree when labeling stuttered speech from audio without transcripts (Bothe, 2008). To alleviate this concern, we conduct our annotations from transcripts as has been done for the Switchboard dataset (Meteer, 1995); however, an improvement in our process would be to align the audio earlier in the pipeline so that the audio segments could be provided to the annotation team along with the transcripts. Additionally, we limit our analysis to typical disfluencies on which non-clinical annotators demonstrably have higher agreement (Lea et al., 2021). Our labels are focused on typical disfluencies, and while these disfluencies are present in stuttered speech, speech from PWS often contains additional stuttering-specific disfluencies including broken words, prolonged sounds, and blocks. We encourage future researchers who have the expertise to do so to add these stuttering-specific disfluencies to FluencyBank Timestamped. We hope that the transcribed and timestamped nature of our data will assist in this process: for example, the duration of words can be calculated from our provided timestamps, and we expect that broken words or prolonged sounds will be associated with tokens that are longer in duration than they would be otherwise.

Finally, we acknowledge the limitations of the findings we present in Section 4. We derive these results using BERT and Whisper models, but it is important to recognize that different architectures, training data, or training processes may yield different performances. However, our general findings related to the drop in model performance

for speech from PWS are consistent with recent work on ASR (Shonibare et al., 2022; Lea et al., 2023). Fine-tuning these models on speech from PWS will likely improve performance, but we present an analysis of off-the-shelf performance because it reflects the current state of these speech technologies when encountering speech from PWS.

## Acknowledgments

We thank Yara El-Tawil, Aneesha Sampath, and James Tavernor who assisted with the data annotation. This article stems from the 2023 Research Symposium at ASHA Convention, which was supported by the National Institute on Deafness and Other Communication Disorders of the National Institutes of Health under Award R13DC003383 to Margaret Rogers. This work was additionally supported by the National Science Foundation (NSF; RI-2230172). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

## Data Availability Statement

The original FluencyBank dataset is publicly available from the TalkBank portal: <https://fluency.talkbank.org/>. FluencyBank Timestamped can be accessed via the same portal after submitting an email request to Nan Ratner. This controlled access approach ensures that the annotations are accessible to researchers but are kept separate from the publicly-available teaching resource. The Switchboard audio data and timestamped transcripts with typical speech used for model training and comparison purposes can be purchased via LDC: <https://catalog.ldc.upenn.edu/LDC97S62>. In our previous work, we have introduced the processing code to derive filled pause, repetition, revision, and partial word codes for the Switchboard data and this is available on Github: [https://github.com/amritkromana/disfluency\\_detection\\_from\\_audio](https://github.com/amritkromana/disfluency_detection_from_audio).

## References

- Ambrose, N. G., & Yairi, E. (1999). Normative disfluency data for early childhood stuttering. *Journal of Speech, Language, and Hearing Research*, 42(4), 895-909.
- Amir, O., Shapira, Y., Mick, L., & Yaruss, J. S. (2018). The speech efficiency score (SES): A time-domain measure of speech fluency. *Journal of fluency disorders*, 58:61–69.

- Bayerl, S. P., Wagner, D., Nöth, E., & Riedhammer, K. (2022). Detecting dysfluencies in stuttering therapy using wav2vec 2.0. In *Proceedings of Interspeech 2022* (pp. 2868-2872).
- Bothe, A. K. (2008). Identification of children's stuttered and nonstuttered speech by highly experienced judges: Binary judgments and comparisons with disfluency-types definitions. *Journal of Speech, Language, and Hearing Research*.
- Craig, A., Blumgart, E., & Tran, Y. (2009). The impact of stuttering on the quality of life in adults who stutter. *Journal of fluency disorders*, 34(2):61–71.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171-4186).
- Erickson, S., & Block, S. (2013). The social and communication impact of stuttering on adolescents and their families. *Journal of fluency disorders*, 38(4):311–324.
- Giorgino, T. (2009). Computing and visualizing dynamic time warping alignments in R: The dtw package. *Journal of Statistical Software*, 31(7).
- Goberman, A. M., Blomgren, M., & Metzger, E. (2010). Characteristics of speech disfluency in Parkinson disease. *Journal of Neurolinguistics*, 23(5), 470-478.
- Godfrey, J. J., Holliman, E. C., & McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *IEEE 1992 International Conference on Acoustics, Speech and Signal Processing* (Vol. 1, pp. 517–520). IEEE Computer Society.
- Goldwater, S., Jurafsky, D., & Manning, C. D. (2010). Which words are hard to recognize? Prosodic, lexical, and disfluency factors that increase speech recognition error rates. *Speech Communication*, 52(3), 181-200.
- Harmon, T. G., Jacks, A., & Haley, K. L. (2019). Speech fluency in acquired apraxia of speech during narrative discourse: Group comparisons and dual-task effects. *American Journal of Speech-Language Pathology*, 28(2S), 905-914.
- Howell, P., Davis, S., & Bartrip, J. (2009). The university college London archive of stuttered speech (UCLASS).
- Jouaiti, M., & Dautenhahn, K. (2022). Dysfluency classification in stuttered speech using deep learning for real-time applications. In *IEEE 2022 International Conference on Acoustics, Speech and Signal Processing* (pp. 6482–6486). IEEE Signal Processing Society.

- Klein, J. F., & Hood, S. B. (2004). The impact of stuttering on employment opportunities and job performance. *Journal of fluency disorders*, 29(4):255–273.
- Klompas, M., & Ross, E. (2004). Life experiences of people who stutter, and the perceived impact of stuttering on quality of life: Personal accounts of South African individuals. *Journal of fluency disorders*, 29(4):275–305.
- Kourkounakis, T., Hajavi, A., & Etemad, A. (2021). Fluentnet: End-to-end detection of stuttered speech disfluencies with deep learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2986–2999.
- Lea, C., Huang, Z., Narain, J., Tooley, L., Yee, D., Tran, D. T., Georgiou, P., Bigham, J. P., & Findlater, L. (2023). From user perceptions to technical improvement: Enabling people who stutter to better use speech recognition. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1–16).
- Lea, C., Mitra, V., Joshi, A., Kajarekar, S., & Bigham, J. P. (2021). Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter. In *IEEE 2021 International Conference on Acoustics, Speech and Signal Processing* (pp. 6798–6802). IEEE Signal Processing Society.
- Louradour, J. (2023). Whisper-timestamped. <https://github.com/linto-ai/whisper-timestamped>.
- MacWhinney, B. (2000). The CHILDES project: Tools for analyzing talk. 3<sup>rd</sup> edition. Mahwah, NJ: Lawrence Erlbaum Associates.
- Meteer, M. (1995). Dysfluency Annotation Stylebook for the Switchboard Corpus.
- Mitra, V., Huang, Z., Lea, C., Tooley, L., Wu, S., Botten, D., Palekar, A., Thelapurath, S., Georgiou, P., Kajarekar, S., & Bigham, J. (2021). Analysis and tuning of a voice assistant system for dysfluent speech. In *Proceedings of Interspeech 2021* (pp. 4848–4852).
- Müller, C., Großmann-Hutter, B., Jameson, A., Rummer, R., & Wittig, F. (2001). Recognizing time pressure and cognitive load on the basis of speech: An experimental study. In *User Modeling 2001: 8th International Conference, UM 2001 Sonthofen, Germany, July 13–17, 2001 Proceedings 8* (pp. 24–33). Springer Berlin Heidelberg.
- Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., & Vesely, K. (2011). The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society.
- Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust speech recognition via

- large-scale weak supervision. In *International Conference on Machine Learning* (pp. 28492–28518). PMLR.
- Ratner, N. B., & MacWhinney, B. (2018). Fluency bank: A new resource for fluency research and practice. *Journal of fluency disorders*, 56:69–80.
- Rev (Accessed: 2024). Rev transcription services. <https://www.rev.com/>.
- Riad, R., Bachoud-Lévi, A.-C., Rudzicz, F., & Dupoux, E. (2020). Identification of primary and collateral tracks in stuttered speech. In *Proceedings of the Twelfth Language Resources and Evaluation Conference* (pp. 1681-1688).
- Riley, G., & Bakker, K. (2009). SSI-4: Stuttering severity instrument. London, England: PRO-ED, an International Publisher.
- Rocholl, J. C., Zayats, V., Walker, D. D., Murad, N. B., Schneider, A., & Liebling, D. J. (2021). Disfluency detection with unlabeled data and small BERT models. In *Proceedings of Interspeech 2021* (pp. 766-780).
- Romana, A., & Koishida, K. (2023). Toward a multimodal approach for disfluency detection and categorization. In *IEEE 2023 International Conference on Acoustics, Speech and Signal Processing*, (pp. 1–5). IEEE Signal Processing Society.
- Romana, A., Koishida, K., & Provost, E. M. (2023). Automatic disfluency detection from untranscribed speech. *arXiv preprint arXiv:2311.00867*.
- Romana, A., Niu, M., Perez, M., Roberts, A., & Provost, E. M. (2022). Enabling off-the-shelf disfluency detection and categorization for pathological speech. In *Proceedings of Interspeech 2022* (pp. 1916–1920).
- Shonibare, O., Tong, X., & Ravichandran, V. (2022). Enhancing ASR for stuttered speech with limited data using detect and pass. *arXiv preprint arXiv:2202.05396*.
- Tichenor, S. E., & Yaruss, J. S. (2019). Stuttering as defined by adults who stutter. *Journal of Speech, Language, and Hearing Research*, 62(12), 4356-4369.
- Tomanek, K., Tobin, J., Venugopalan, S., Cave, R., Seaver, K., Heywood, R., & Green, J. (2023). Large language models as a proxy for human evaluation in assessing the comprehensibility of disordered speech transcription. *ICML 2023 workshop on Artificial Intelligence Human Computer Interaction*.
- Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Pierric, C., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T., Gugger, S., ... & Rush, A. M. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yaruss, J. S., & Quesal, R. W. (2006). Overall assessment of the speaker’s experience of stuttering (oases): Documenting multiple outcomes in stuttering treatment. *Journal of fluency disorders*, 31(2):90–115.

## Tables and Figures

Disfluency Labels	Examples
Filled Pause	where it was one <b>uh</b> on one but <b>um</b> honestly I don’t know
Repetition	it’s a <b>two two</b> two way street <b>I used to</b> I used to think about... I was a <b>co- co-</b> covert stutterer
Revision	<b>we were</b> uh they were going to... <b>the stuttering is not as</b> um it doesn’t affect me as much I would say <b>m-</b> uh for me now...
Partial Word	I was a <b>co- co-</b> covert stutterer I would say <b>m-</b> uh for me now...

Table 1: Disfluency classes with examples of each type. We note that a token can be in multiple classes, for example, a token can be a partial word + a repetition, or a partial word + a revision. Additionally, a token can be a repetition + a revision, in the case of nested disfluencies.

Figure 1: The frequency of disfluencies (a) at the token-level (per 100 words) and (b) at the frame-level (per 1 second of speech). The Switchboard dataset (SWB) contains typical speech with disfluencies, the original FluencyBank dataset (FB Original) which contains speech from adults who stutter, and the FluencyBank Timestamped dataset (FB Timestamped), has gone through another round of annotation to better reflect the audio. Only Switchboard and FluencyBank Timestamped have token timings necessary for a frame-level comparison.

Table 2: Whisper for intended speech recognition with the Switchboard dataset (SWB), which contains typical speech with disfluencies and the FluencyBank Timestamped dataset (FB TS), which contains speech from people who stutter. Intended speech word error rate (isWER) is the word error rate between the intended speech ground truth text (i.e., without disfluencies) and the ASR-generated text. We also report the percentage of disfluencies that Whisper transcribes for each disfluency type: filled pauses (FP), repetitions (RP), revisions (RV), and partial words (PW).

Dataset	isWER	Disfluencies Transcribed			
		FP	RP	RV	PW
SWB	15.2	8.7	9.4	43.1	1.4
FB TS	15.4	13.1	10.2	47.2	3.0

Table 3: Some examples of the errors Whisper makes in intended speech recognition with speech from people who stutter in the FluencyBank Timestamped dataset. As shown in Table 2, Whisper effectively removes most disfluencies, but these examples show some limitations of the system where Whisper adds a filled pause, repetition, revision, or mispronunciation that are not a part of the intended speech. We put the disfluency error in bold.

Intended	i am friendly towards people
Whisper	i am friendly towards <b>people</b> you know people
Intended	i know part of it is genetic
Whisper	<b>it's it's it's</b> i know product is genetic
Intended	so i went to speech therapy
Whisper	so <b>um</b> i went to speech therapy
Intended	i am doing a master in biology
Whisper	<b>uh</b> i am doing a master in <b>by y y y yology</b>
Intended	it ended when i was seventeen
Whisper	<b>i ended when i was</b> it ended when i was seventeen

Figure 2: Text-based disfluency detection and audio-based disfluency detection. The text-based approach includes a manual transcription step whereas the audio-based approach is fully automated but more challenging.

Table 4: Text-based disfluency detection and categorization performance. The text-based model processes text and predicts disfluencies at the token-level. We train the model using the Switchboard dataset (SWB), which contains typical speech with disfluencies, and we evaluate it in a zero-shot manner using the FluencyBank Timestamped dataset (FB TS), which contains speech from adults who stutter. The target disfluency classes include filled pauses (FP), repetitions (RP), revisions (RV), and partial words (PW). We repeat the training using 3 random seeds, and we report average and standard deviation of F1 score and recall for each class, as well as weighted and unweighted averages. Note: We evaluate these models with manually transcribed text, which may be a practical limitation in certain applications.

(a) F1 score

Dataset	Disfluency Macros		Disfluency Classes				Non-Disfluent
	Unweighted	Weighted	FP	RP	RV	PW	
SWB	0.85 (0.00)	0.85 (0.00)	1.00 (0.00)	0.85 (0.01)	0.70 (0.04)	0.83 (0.01)	0.99 (0.00)
FB TS	0.73 (0.01)	0.76 (0.01)	1.00 (0.00)	0.69 (0.01)	0.40 (0.01)	0.83 (0.02)	0.99 (0.00)

(b) Recall score

Dataset	Disfluency Macros		Disfluency Classes				Non-Disfluent
	Unweighted	Weighted	FP	RP	RV	PW	
SWB	0.86 (0.00)	0.86 (0.00)	1.00 (0.00)	0.88 (0.00)	0.68 (0.01)	0.88 (0.02)	1.00 (0.00)
FB TS	0.74 (0.01)	0.73 (0.01)	1.00 (0.00)	0.54 (0.02)	0.62 (0.01)	0.80 (0.03)	0.99 (0.00)

Table 5: Audio-based disfluency detection and categorization performance. The audio-based model processes raw audio as input and predicts disfluencies at the frame-level. We train the model using the Switchboard dataset (SWB), which contains typical speech with disfluencies, and we evaluate it in a zero-shot manner using the FluencyBank Timestamped dataset (FB TS), which contains speech from adults who stutter. The target disfluency classes include filled pauses (FP), repetitions (RP), revisions (RV), and partial words (PW). We repeat the training using 3 random seeds, and we report average and standard deviation of F1 score and recall for each class, as well as weighted and unweighted averages. Note: The performance of the audio-based approach lags behind the text-based approach, but the audio-based approach is more scalable because it does not rely on a transcription step.

(a) F1 score

Dataset	Disfluency Macros		Disfluency Classes				Non-Disfluent
	Unweighted	Weighted	FP	RP	RV	PW	
SWB	0.62 (0.01)	0.68 (0.01)	0.86 (0.00)	0.70 (0.01)	0.52 (0.02)	0.40 (0.02)	0.98 (0.00)
FB TS	0.46 (0.00)	0.48 (0.00)	0.83 (0.00)	0.41 (0.01)	0.27 (0.02)	0.33 (0.01)	0.95 (0.00)

(b) Recall score

Dataset	Disfluency Macros		Disfluency Classes				Non-Disfluent
	Unweighted	Weighted	FP	RP	RV	PW	
SWB	0.58 (0.01)	0.65 (0.00)	0.86 (0.02)	0.66 (0.03)	0.46 (0.04)	0.34 (0.04)	0.99 (0.00)
FB TS	0.44 (0.01)	0.43 (0.01)	0.85 (0.02)	0.29 (0.01)	0.37 (0.02)	0.25 (0.01)	0.97 (0.00)

Figure 3: (a) Text-based and (b) Audio-based disfluency detection results for each participant in the FluencyBank Timestamped dataset. Both model types result in considerable participant-level variation, although the variation is higher and more correlated with stuttering severity for the audio-based disfluency detection model.